

# Dual-pass Feature Extraction on Human Vessel Images

W. Hernandez · S. Grimm · R. Andriantsimiavona

Received: 15 March 2013 / Revised: 6 September 2013 / Accepted: 25 September 2013 / Published online: 6 November 2013  
© Society for Imaging Informatics in Medicine 2013

**Abstract** We present a novel algorithm for the extraction of cavity features on images of human vessels. Fat deposits in the inner wall of such structure introduce artifacts, and regions in the images captured invalidating the usual assumption of an elliptical model which makes the process of extracting the central passage effectively more difficult. Our approach was designed to cope with these challenges and extract the required image features in a fully automated, accurate, and efficient way using two stages: the first allows to determine a bounding segmentation mask to prevent major leakages from pixels of the cavity area by using a circular region fill that operates as a paint *brush* followed by Principal Component Analysis with auto correction; the second allows to extract a precise cavity enclosure using a *micro-dilation* filter and an *edge-walking* scheme. The accuracy of the algorithm has been tested using 30 computed tomography angiography scans of the lower part of the body containing different degrees of inner wall distortion. The results were compared to manual annotations from a specialist resulting in sensitivity around 98 %, false positive rate around 8 %, and positive predictive value around 93 %. The average execution time was 24 and 18 ms on two types of commodity hardware over sections of 15 cm of

length (approx. 1 ms per contour) which makes it more than suitable for use in interactive software applications. Reproducibility tests were also carried out with synthetic images showing no variation for the computed diameters against the theoretical measure.

**Keywords** Image segmentation · Image brushing · Principal component analysis · Micro-dilation · Edge-walking · Local and nonlocal method · Dual-pass

## Introduction

Vascular diseases continue to be among the leading causes of death worldwide accounting for nearly 18 million deceases by the end of 2008 and with projections of over 20 million by 2030 [29]. Due to the constant improvement of acquisition techniques such as computed tomography and magnetic resonance, it is possible to have a better image detail of multiple body regions such as the vascular system in a noninvasive fashion for the patient.

One important aspect of modeling the vessels for automatic assessment is to assume a cylindrical shape in 3D and its corresponding circular/elliptical equivalent in a transversal cut. Two of the most common causes of internal cavity or lumen deformation of these structures are stenoses and aneurysms. In the case of stenosis, the deformation is manifested as a narrowing produced by the gradual build up of fatty deposits in the walls of the arteries (also known as plaque), which if not treated, might eventually solidify and degenerate in harder tissue accumulations known as calcium. These deposits along with other causes such as high pressure or plain genetics may also undermine the wall thickness to a point where it starts to bulge producing a dilation of the lumen known as aneurysm [27]. Figures

---

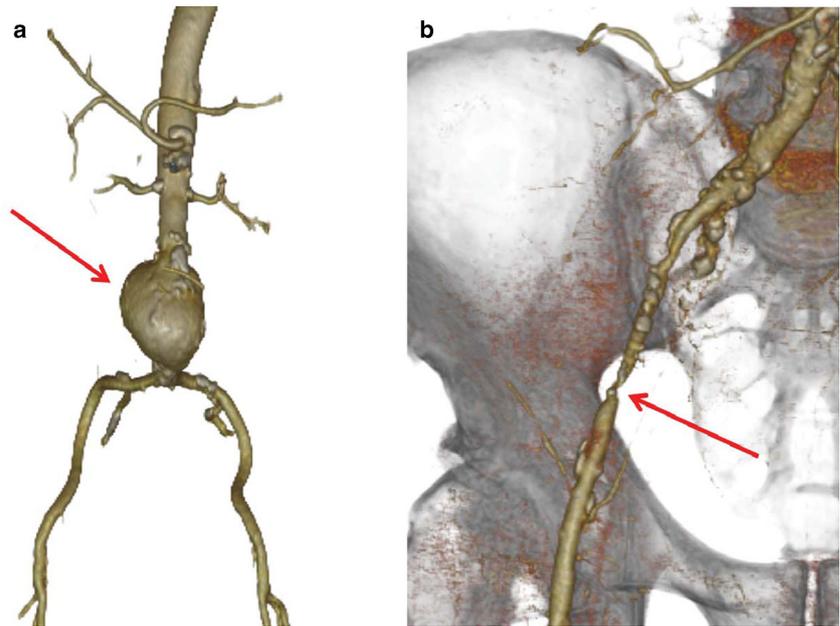
W. Hernandez (✉)  
CCG, School of Computer Sciences, Faculty of Sciences,  
Central University of Venezuela, 47002, Los Chaguaramos,  
1041-A Caracas, Venezuela  
e-mail: walter.hernandez@ciens.ucv.ve

S. Grimm · R. Andriantsimiavona  
Grampian House, Meridian Gate, London E14 9YT, UK

S. Grimm  
e-mail: sgrimm@biotronics3d.com

R. Andriantsimiavona  
e-mail: rado@biotronics3d.com

**Fig. 1** Volume rendering images of a 16-bit image stack with multiple cavity distortions due to fat. **a** Red arrow points to the locations of an aneurysm at the end of the aorta. **b** Red arrow points to the location of a stenosis region



1 and 2 show some typical renderings of these anatomical deformations obtained from a stack of 16-bit images.

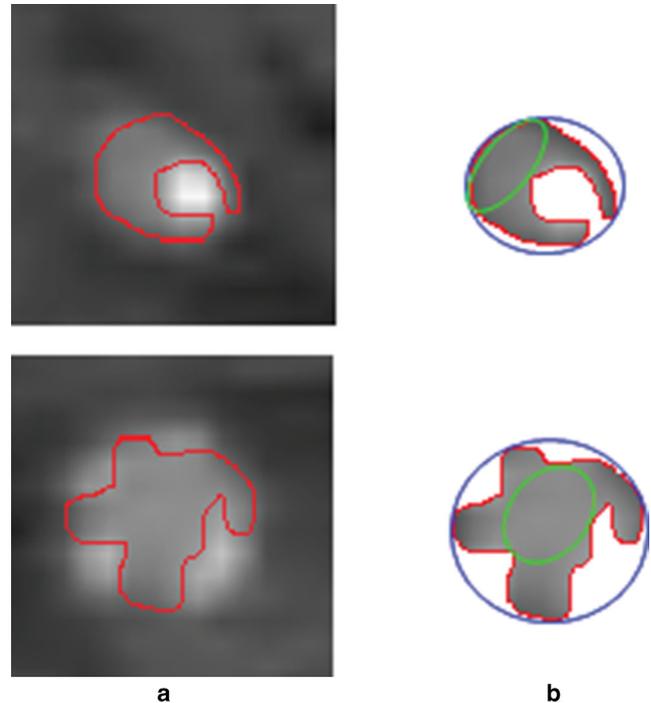
Another important aspect when doing automatic feature extractions in this context is to keep it reasonably fast for interactions from the physician in order to make the whole diagnosis pipeline more efficient. One of the ways to assess the level of cavity distortion is to compute a *stenosis ratio*. These procedure is performed by scrolling through the stack of images and manually marking the diameter of the lumen on each image where a reduced cavity space is found (stenosis point), this is then compared to the diameter (or area) of a neighboring healthy cavity section (reference point).

$$\%stenosis = \frac{1 - \mu_s}{\mu_r} \quad (1)$$

In Eq. 1,  $\mu_s$  refers to the measurement taken on the stenosed point and  $\mu_r$  to the measurement taken on the healthy or reference point. This measurement can either be the maximum diameter of an imaginary ellipsoid enclosing the cavity (expressed in meters) or the area computed as all pixels contained in the cavity (expressed in square meters). The diameter measurement is one of the more straightforward metrics employed by physicians simply because it is easier to manually draw either an ellipsoid or the axis of an imaginary ellipsoid over the cross section to be analyzed. However, the area is an alternative way that offers the chance to include pixels of the lumen that are normally disregarded (see Fig. 2). The work presented in this paper allows for stenosis computation based on both diameters and areas.

Accurate lumen estimations are critical for the selection of the stents and sheets employed on procedures such as aortic aneurysm repairs (e.g., aortic aneurysm repairs) and

also to quantify plaque and/or calcium levels [5]. Despite all the research efforts in this area, lumen segmentation remains a primary challenge for automated processes, particularly when the images acquired to visualize the vascular structure are affected by the conditions previously mentioned.



**Fig. 2** Cavity measurements on a vessel image. **a** Two cut planes of a stack image with high level of cavity distortion and a red overlay of the optimal lumen area. **b** Same areas with two examples of ellipsoids overlaid in blue and green, notice that for both ellipsoids (or any other potential ellipsoid) a good portion of lumen pixels are left out

Literature on the topic of vascular segmentation has been quite prolific partly due to the ubiquity of vessels in the human anatomy and also to the high incidence of vascular diseases worldwide. This has not necessarily translated in a standard way to extract features from the existing imaging modalities since different people in healthcare will have quite diverse opinions about a particular segmentation result.

Broadly speaking, the outcome of a vascular segmentation process involves obtaining a segmentation mask or polygon of the contour and/or a central axis (centerline) from the vessel. The basic assumption is that vessels have tube-like shapes in 3D and elliptical in 2D. An automatic contour computation can be derived from a pre-existing centerline and vice versa. Methods that focus on the extraction of a centerline (centerline tracking) may not necessarily produce a refined representation of the wall contour that is appropriate for quantification purposes; this central axis information though is extremely valuable to exploit a subsequent accurate contour segmentation process. Tracking methods can be grouped as skeleton-based, where the full vessel structure is targeted via low level operations (e.g., thresholding, region growing, erosion, and dilation) [6] and then a centerline is obtained via a skeletonization or centerline-based process where an optimal path is computed by minimizing a cost function which usually incorporates both geometric restrictions and image restrictions [7]; alternatively, the so-called *medialness* operators [2] try to pick the vessel center without relying on a binary segmentation mask.

Contour extraction as such can incorporate different strategies. Snake and active contour approaches such as the dynamic programming scheme from [11] offer a flexible combination of image features and geometry. They can be relatively slow but improvable through the use of multi-scale filters. The principle of the technique is to guide the segmentation via external forces based on image features (e.g., voxel intensity and contrast) while constraining the shape via internal forces (e.g., curvature and orientation). Explicit or parametric active contour models for vascular segmentation were successfully applied to vessel segmentation by [10, 17, 30]. Implicit models are based on the level-set and fast marching methods proposed by [21] and [23] among others; On [9], a new speed-control function was proposed for fast marching in order to handle variable contrasts along the perimeter of the contour in carotid run-offs.

Other works like [1], although applied to ultrasound data, exposed some of the shortcomings of the classical active contour approach and exhibits the benefits of diffusion filters based on Canny detectors [8] and operators such as

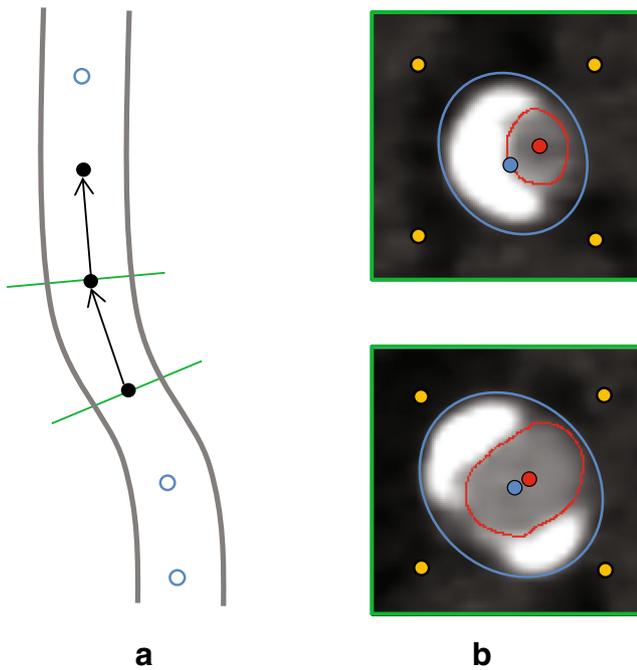
Sobel and Prewitt [19, 25] to improve the accuracy of external energy functions. For a review of the fundamentals of these techniques applied to medical imaging, see [16].

The survey by [13] defines the following grouping criteria for vascular segmentation: extraction pattern recognition techniques, model-based approaches, tracking-based approaches, artificial intelligence-based approaches, neural network-based approaches and tube-like object detection approaches. Later on, the survey by [14], the techniques were updated and grouped based on a high level approach categorized as models, features and extraction techniques. The reader should refer to those two comprehensive surveys to further expand on the methods of this broad segmentation field. In the next section, we present our own approach for lumen extraction.

## Materials and Methods

The input for the algorithm is a list of centerline points that go through the tubular structure. These were automatically obtained from our own centerline tracking implementation based on the principles described by [4]. However, any centerline tracking algorithm or scheme to mark these points can be alternatively employed as this is not the main focus of the work presented. The purpose of the list of centerline points is to create an orientation vector between two consecutive points and then build a perpendicular plane; along this plane, the image containing the vessel to be processed is sampled from the 3D image stack (see Fig. 3a). As for the actual location of the centerline points within the vessel, there are no specific requirements as they might not be the perfect centroids of the tubular structure (see Fig. 3b).

There are two main stages within the algorithm: boundary estimation and inner estimation. The boundary estimation stage was designed with the purpose of including image regions of both medium–low (lumen) and high (calcium) intensity pixels, discarding at the same time image regions or patterns of pixels that may leak and distort the overall tubular shape (such as the ones found in bifurcations and/or adjacent tissue on similar data ranges of the vessel). Following, Principal Component Analysis is applied over the edge pixels of the previously obtained mask and the information used to initialize parameters of subsequent iterations. The inner extraction stage is triggered after the boundary estimation phase; here, the lumen contour is obtained by employing region growing, selection of the biggest component grown, a *micro-dilation* and a loop-free path traversal, or *edge-walking* mechanism that ensures the extraction of a compact and consistent contour. Partial volume effects are considered as well. Following, sections “[Stage I—Boundary Estimation](#)” and “[Stage II—Inner Estimation](#)” explain the details of both stages, section “[Algorithm](#)” showcases the



**Fig. 3** **a** Representation of a vessel cut where the *arrows* depict the directional vectors built using two consecutive centerline points and the resultant perpendicular image plane in *green*. **b** Axial views of two different planes with the lumen depicted in *red* and the outer wall in *blue* (both with its corresponding centroids). Any point inside (*blue* and *red* dots) or close to the vessel (*orange* dots) can be used to center the image plane to be processed

algorithm in pseudo-formal notation, section “[Parameter Adjustment and Measurement Tool](#)” details how to manually adjust the parameters of the algorithm, and section “[Datasets](#)” describes the data used for the tests.

#### Stage I—Boundary Estimation

The objective of this stage is to avoid leakage areas caused by pixels corresponding to branches or other tissues on the same intensity range of the vessel. To illustrate the basic principles of the algorithm, the synthetic image depicted on Fig. 4a was generated to summarize the issues to be approached on a single image. Here, we distinguish between the internal wall of the vessel which includes areas 1, 2, and 3 and the lumen, which can be seen as area 3 not including areas 1 and 2 (calcium artifacts).

The first step in this phase is to extract the pixels of the internal wall of the tubular structure by performing a *circular region fill*; the extracted region will contain both the actual cavity and the calcium (if present). The filling is done using a small circular pixel mask which starts from the centerline point and moves around its neighbors (also in a circular fashion) until all the region is filled. Neighbor and border pixels are found using the tissue intervals defined in Table 1.

The traversal of this circular mask resembles the movement of a paint *brush* as depicted in Fig. 4b where the brushing process maintains a smooth circular border after filtering out any disruptions from the central passage. The radius of the brush is set in two ways:

- When there is no previous fitting information available (e.g., for the first point of the centerline), a fan of pixel rays is casted at equally spaced angles from the centerline point in order to cover a full circumference; the propagation of each one of these rays is terminated when its pixel value exceeds the interval defined by the minimum lumen and maximum calcium threshold values (see Table 1). The radius of the brush is chosen as the third of the maximum ray length from all rays casted;
- If previous fitting information is available, the radius is directly set as a third of the major axis of the ellipse estimated in the previous contour. The brush size will not exceed a predefined maximum size of 5 mm established for the image stacks tested but it can be adjusted to support different sizes (see section “[Parameter Adjustment and Measurement Tool](#)”).

Figure 4b also shows the effects of the brushing, where the red top and bottom areas of the edge patterns which are either not part of the cavity or deviate from the main structure of the vessel are left out and the circular shape of the brush delineates a new circular or convex edge under the red areas of pixels that were filtered out. The segmentation mask obtained acts like a *bounding perimeter* because it effectively prevents that any subsequent segmentation process leaks to undesired areas.

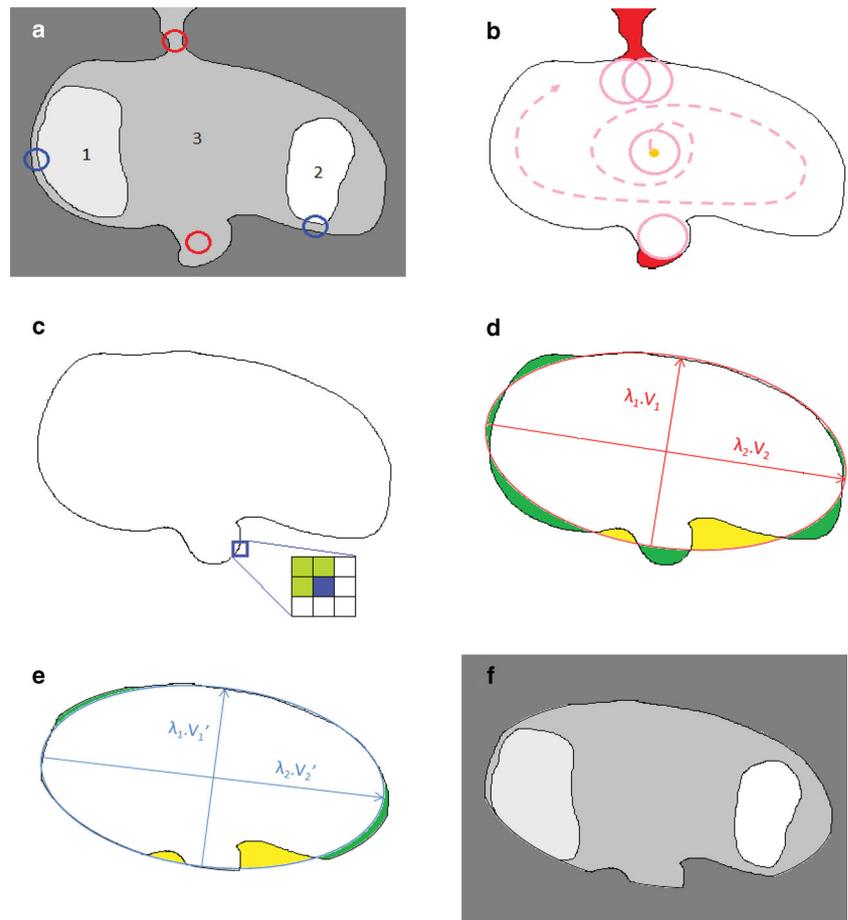
The next step is to use the image obtained after the filtering shown in Fig. 4b as the input for Principal Component Analysis (PCA). This is done with the objective of finding the pixels that best approximate the tubular shape depicted by an ellipse. The first step of the PCA employed in this work consists in the computation of the covariance matrix over the edge positions as:

$$Cov(X, Y) = \frac{\sum_{i=1}^{numEdgePixels} (X_i - \bar{X})(Y_i - \bar{Y})}{numEdgePixels - 1} \quad (2)$$

With  $(X_i, Y_i)$  the center of a  $3 \times 3$  grid where the central pixel must have a minimum of three and a maximum of seven neighbors pixels in the cavity in order to be considered for the statistical mean computation ( $\bar{X}$  and  $\bar{Y}$ ) (see Fig. 4c). Subsequently, the covariance matrix is computed as:

$$C = \begin{pmatrix} Cov(X, X) & Cov(X, Y) \\ Cov(Y, X) & Cov(Y, Y) \end{pmatrix} \quad (3)$$

**Fig. 4** First pass of the algorithm applied to a synthetic characterization of an image plane: **a** Areas 1 and 2 represent calcium; area 3 the cavity; *red circles*, leakages or bifurcations; *blue circles*, areas candidates to partial volume effects. **b** “Brushing” applied to image **(a)** repulses leakage areas (*red*). **c** Example of a  $3 \times 3$  grid moving along the edges which finds three top-left pixels candidates for PCA computation. **d** Eigenvalues ( $\lambda$ ) and eigenvectors ( $v$ ) define the longitude and direction of an ellipse; pixels located in the internal (*yellow*) or external (*green*) areas of the ellipse that are too far away from the approximation (goodness of fit) are filtered out. **e** New ellipse ( $\lambda'$  and  $v'$ ) generated with the pixels excluded from **(c)**. **f** Final image obtained after the first pass



From the covariance matrix, we obtain the systems (4) and (5), {where  $\lambda$  represents the eigenvalues and  $v$  the eigenvectors.

$$\det(C - \lambda I) = 0 \tag{4}$$

$$(C - \lambda I)v = 0 \tag{5}$$

The eigenvectors represent the direction of the axes, and the eigenvalues determine the scale of the major/minor axes of the ellipse (see Fig. 4d and e). To solve this system, the strategies described in [24] were used.

**Table 1** Optimal threshold ranges used for segmentation (in Hounsfield units) estimated experimentally from the image stacks used

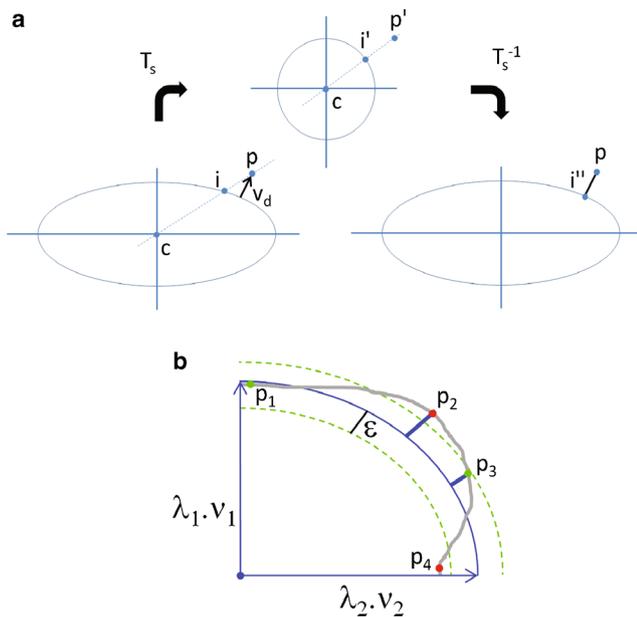
Tissue	HU value range (Min, Max)
Lumen	(150, 500)
Calcium	(500, 1,300)

The ellipse obtained is compared with the bounding perimeter in order to estimate the goodness of the fit ( $E$ ). This was achieved using the normalized sum of squares of Euclidean distances ( $Dist$ ) from the edges as shown in (6).

$$E = \frac{1}{n} \sum_{i=1}^n Dist_i^2; \text{ with } n \text{ the number of vessel edge pixels} \tag{6}$$

These distances can be computed directly from the center of the estimated ellipse assuming it is almost circular [26]. Figure 5a shows that the intersection pixel  $i$  is obtained between the center  $c$  and the pixel  $p$  on the vessel edge. However, if the ellipse elongates beyond a certain degree, then the distance computed is longer than the one described by vector  $V_d$ . For this reason, an approach similar to the one used in [31] was used to make sure all types of ellipses are considered while keeping the computations efficient.

If  $E$  goes beyond a tolerance value ( $\epsilon$ ), then the farthest pixels are filtered out (see Fig. 5b and yellow/green



**Fig. 5** Error computation and goodness of the fit: **a** Intersection pixel  $i$  will not yield the shortest distance (shown as vector  $V_d$ ) to pixel  $p$  when using the center  $c$  of the estimated ellipse directly. Instead, affine transformation  $T_s$  is applied to obtain a circle and pixel  $i'$ , then  $T_s^{-1}$  is used to obtain the shortest path with  $i''$ . **b** First quadrant of a sample ellipse: distances to the estimated ellipse are computed from edge pixels  $p_1$ ,  $p_2$ ,  $p_3$ , and  $p_4$ . A threshold value  $\varepsilon$  is used to establish if a new estimation is necessary. This threshold resembles the *green dotted arcs* where pixels are identified as candidates to be filtered out ( $p_2$  and  $p_4$ ) or to be reconsidered for the next PCA iteration ( $p_1$  and  $p_3$ )

areas on Fig. 4d, e) and the PCA is recomputed. Three values of  $\varepsilon$  (5, 10, and 15 %) were explored to study the goodness of the fit over a number of 16-bit image stacks (see section “[Datasets](#)” for more details on the image sets) resulting in  $\varepsilon = 10\%$  the most adequate tolerance value according to a visual inspection performed by a radiologist.

Here, it is worth noting that due to the initial brushing process which filters the pixels that cause the strongest disruption of the elliptical shape, the remaining edge pixels will tend to group more closely to the ellipse fit. This means that the error correction process does not require a lot of iterations to reach the established goodness  $E$ . Nevertheless, a maximum number of iterations was additionally introduced for a set of cases (less than 1 %) where the recovery will take too long or is simply not possible (e.g., excessive image noise, full obstruction); in these small set of cases, the ellipse from the previous contour is displayed and the users have the option to manually outline the contour themselves (see Fig. 10).

Having refined the bounding perimeter, a more precise boundary estimation is achieved containing both lumen and

calcification areas and discarding leakage areas as shown in Fig. 4f. The final edge candidates of this analysis are stored as a list of pixels to be used on the next stage. Additionally, the geometric information from the ellipse fitting (center, eigenvectors, and eigenvalues) is cached in order to be used in the next contour to set a more precise set of dimensions for the patch, brush, and partial volume analysis (see section “[Stage II—Inner Estimation](#)”).

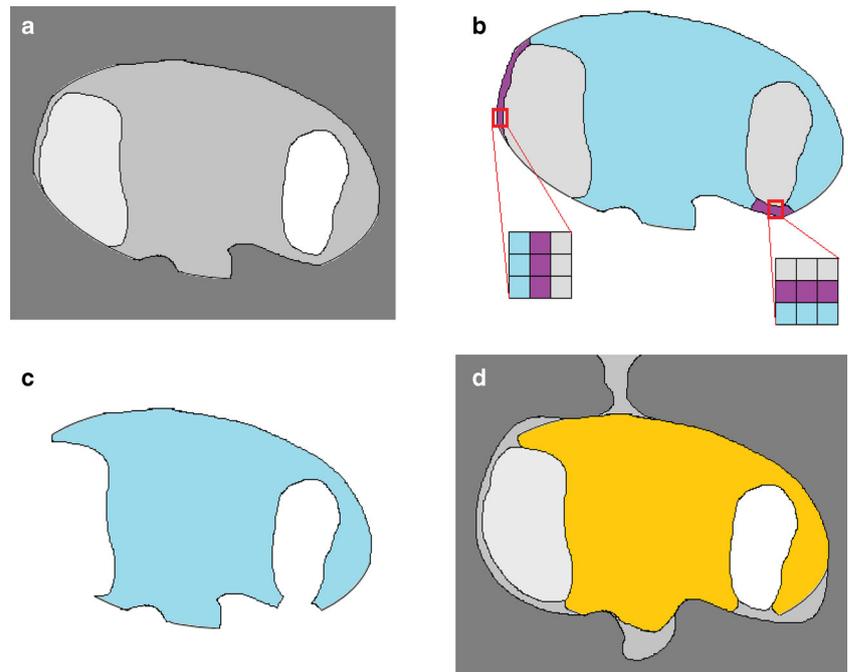
## Stage II—Inner Estimation

The purpose of the inner estimation stage is to identify calcium areas within the boundary estimation previously achieved and to make sure there are no further leakages that affect the overall shape of the internal cavity contour to be extracted.

The first step consists on extracting the pixels corresponding to the lumen via another modified region filling process. Compared to the filling described on section “[Stage I—Boundary Estimation](#)”, this one behaves more similar to a standard region grow (no circular mask/propagation) but only taking into account pixel values corresponding to the lumen (see Table 1) and performed exclusively over the boundary estimation obtained on Stage I. The modifications to the filling allow for filtering of partial volume effects and identification of calcium areas. These pixels are identified by moving along the edge and sampling to the inside of the vessel using half of the minimum ellipse radius (a quarter of the minimum eigenvalue) up to a maximum length of 2 mm (also adjustable, see section “[Parameter Adjustment and Measurement Tool](#)”). If a pixel found within this sampling range holds an intensity value of calcium, then they are considered to be thin enough to be discarded and hence removed from the lumen mask. Figure 6b depicts this principle where the purple pixels of the  $3 \times 3$  grid contains calcium and lumen in both sides and the thin pattern of lumen pixels are considered as part of the calcium producing the shape displayed on Fig. 6c, and d.

The *edge-walking* step was designed to extract the final edge pixels that will conform the contour making sure is properly closed and consistent with the shape obtained in the previous steps. Using the updated edge list from the partial volume filtering a path of neighboring pixels will be followed until the starting pixel is reached again to generate a closed contour checking at the same time that none of them is visited more than once. During this step, a *micro-dilation* is performed using the patterns shown in Fig. 7a. These were carefully selected to identify all possible pixel combinations that will lead to a path with no end; in other words, pixels that are part of additional leakage areas. Thus, removing the central pixel of the  $3 \times 3$  grid centered on any of the 20 patterns ensures that the algorithm does not visit pixels that

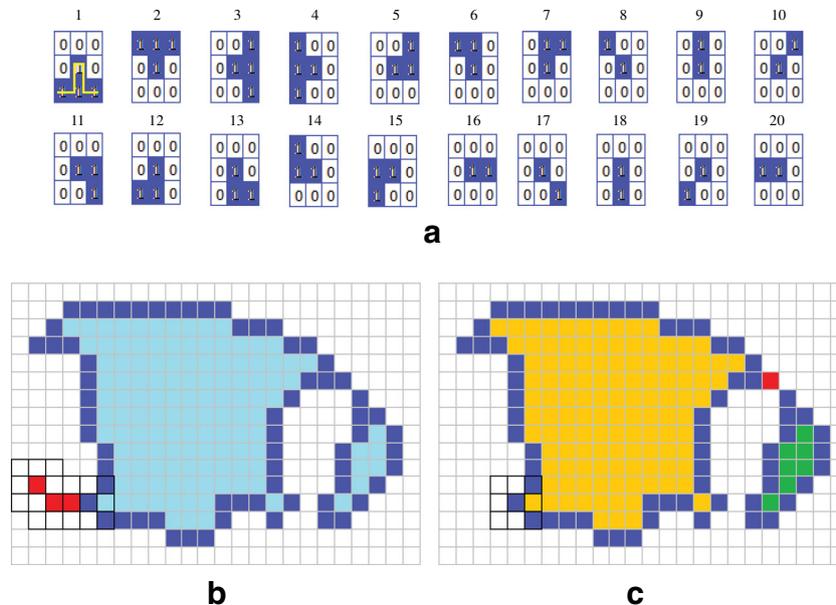
**Fig. 6** Second pass of the algorithm applied to the synthetic characterization obtained in Fig. 4. **a** Image obtained after Stage I used as input for Stage II. **b** Partial volume analysis: violet pixels represent edge pixels located between calcium (gray) and lumen (cyan) pixels. If calcium pixels are found when sampling to the inner side of the violet pixels, then the outer side is considered to be part of the calcium producing the image seen in (c). **d** Overlay of the final lumen mask obtained over the original image



will not really contribute on the overall shape of the contour and/or in the computation of lumen measurements.

Figure 7b shows an example of the *micro-dilation* process where the grid moves recursively along the edge and

identifies patterns 17 and 16 from Fig. 7a; the pixels are consequently removed and the new resulting area can be seen in the highlighted grid of Fig. 7c. After the micro-dilation is applied, and for cases when a previously visited



**Fig. 7** *Micro-dilation* and *Edge-walking* processes. **a** Pixel patterns for the edge-walking step. The central pixel of the  $3 \times 3$  grid will be removed from the edge list whenever a pattern is found. The yellow line on pattern 1 also shows that the central pixel would cause the bottom center pixel to be visited twice during the traversal if not filtered out. **b** The highlighted  $3 \times 3$  grid moves through the edge list and identifies three patterns (red pixels) from the list shown in (a) (from left

to right: pattern 17, pattern 16, and again pattern 16). The resulting area is shown in (c). **c** Highlighted  $3 \times 3$  grid after micro-dilation. This image also shows that if a pixel is visited twice (red pixel) after the filtering, two new segmentation components will be generated (green and yellow areas) during the edge-walking and the biggest one chosen as the definite lumen area (yellow area)

pixel is hit (different from the starting pixel), then more than one contour will be generated and the biggest closed contour will be chosen as the final outcome of the algorithm (see Fig. 7c). As previously mentioned on section “[Stage I—Boundary Estimation](#)”, if the contour is still not adequate for the users, they can simply outline the contour themselves (see Fig. 10).

#### Algorithm

Algorithm 1 shows in pseudo-formal notation a summary of the steps previously described to extract the contour from a stack of images. A more detailed version can also be seen in Algorithm 2 where lines from 5 to 15 correspond to Stage I and lines 16 to 31 correspond to Stage II.

It is worth mentioning that the new center of the ellipse obtained at the end of Stage I fit may not necessarily be the same (or even close) to the original centerline point used as input; however, this new center could be useful to produce curved planar reformation images (CPR) [12] where the center of the outer walls of the tubular structure is required instead of the lumen center. Also worth to note is that line 13 checks the maximum number of iterations (*max\_value*) is not hit before moving on to Stage II as explained at the end of section “[Stage I—Boundary Estimation](#)”. Figure 8 shows the algorithm applied to a single image from the stack.

---

#### Algorithm 1 General outline of the contour extraction algorithm

---

```

1:  for all centerlinePoints do
2:    // Stage I
3:    LoadPreviousEstimation();
4:    FilterOuterLeakages();
5:    repeat
6:      ApplyPCA();
7:      Estimate_Error();
8:    until errorMinimized
9:    // Stage II
10:   StoreCurrentEstimation();
11:   Filter_Parcial_Volume_Effect();
12:   while badPatternAlongPath do
13:     Micro-dilation();
14:     EdgeWalking();

```

---



---

#### Algorithm 2 Detailed outline of the contour extraction algorithm

---

```

1:  // Initialization stage
2:  for all centerlinePoints do
3:    Build_Patch_Plane();
4:    // Stage I
5:    Set_Brush_Radii(prevEllipseInfo);
6:    Brush_Edges();
7:    repeat
8:      EdgeList ← PCA();
9:      E ← Estimate_Error();
10:     if (E > ε) then
11:       Filter_Edge_Pixels(EdgeList);
12:     until (E ≤ ε) ∨ (max_iterations)
13:     if max_iterations = max_value then
14:       continue;
15:     // Stage II
16:     prevEllipseInfo ← Store_Current_EllipseInfo();
17:     Threshold_and_Eval_Components();
18:     Filter_Parcial_Volume_Effect();
19:     // Micro-dilation and Edge-Walking();
20:     startPixel ← EdgeList.StartPixel();
21:     endPixel ← EdgeList.EndPixel();
22:     currentPixel ← EdgeList.StartPixel();
23:     while currentPixel ≠ endPixel do
24:       while IsBadPattern(currentPixel) do
25:         FilterPattern(EdgeList);
26:       if (currentPixel = startPixel) then
27:         ContourList.Add(NewContour(startPixel,
28:           currentPixel));
29:       if (currentPixel ≠ endPixel) then
30:         startPixel() ← currentPixel;
31:       else
32:         ← MaxContour(ContourList);

```

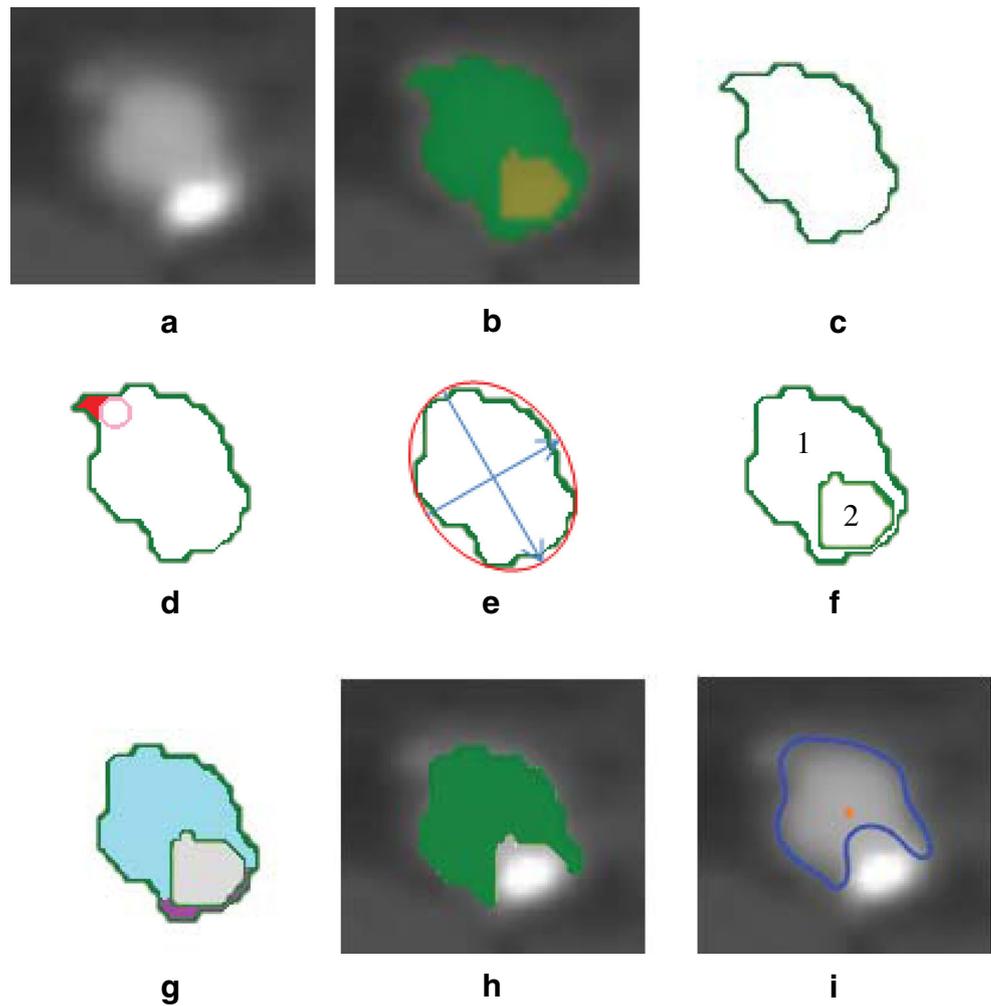
---

#### Parameter Adjustment and Measurement Tool

The algorithm was integrated as part of a visualization and measurement tool implemented in DirectX® and C++ (Visual Studio 2010®, 64-bit compiler). This tool allows for the adjustment of the most relevant parameters of the algorithm, manual delineation of contours across a vessel centerline, estimation of the stenosis ratio (as previously shown in Eq. 1), and contour correction.

The area, maximum, and minimum diameters are automatically computed using Algorithm 1 on a chosen section of the vessel (points A and B in Fig. 9a). The overall maximum and minimum diameters are also automatically located and displayed over our own implementation of CPR images along with the stenosis computation for the diameter and the area. Points A and B can be manually adjusted by the user

**Fig. 8** Algorithm applied to single image from the stack. **a** Original image. **b** Overlay with thresholds for lumen in *green* and *yellow* for calcium. **c** Initial edge mask used for the brushing. **d** *Pink circle* represents the brushing process applied on a small leakage at the top left that will be filtered out (*red area*). **e** Ellipse fit to the new edge mask obtained from the previous step (no leakages); no error correction was required in this case. **f** The partial area obtained is reanalyzed for partial volume effects, this time including both thresholds ranges for lumen (*area 1*) and calcium (*area 2*). **g** *Violet pixels* were marked as part of the calcium as they fall below the maximum tolerance defined in the partial volume effect step. **(h)** Final lumen mask obtained. **i** Overlay with final contour extracted after the micro-dilation and edge-walking process



and the ratio is automatically recomputed and displayed in real time.

To help alleviate the inherent variability between studies and/or readings, a simple interface parameter configuration tool was included (see Fig. 9b). The basic guidelines to adjust the parameters of the algorithm are the following:

- ★ **Fit group:** altering the values of the fit will help to improve segmentation in cases where the outer wall present a high level of distortion from an ellipse (e.g., noisy data).
- ★ **Dimensions group:** increasing or decreasing the maximum patch size defines the segmentation action ratio of all steps of the algorithm. Decreasing the stamp size will make the region filling to go further on potential leakage areas or around calcium areas; increasing the stamp size will filter out small areas stronger. Reducing the max partial volume effect will have the effect of marking a more aggressive set of pixels towards the

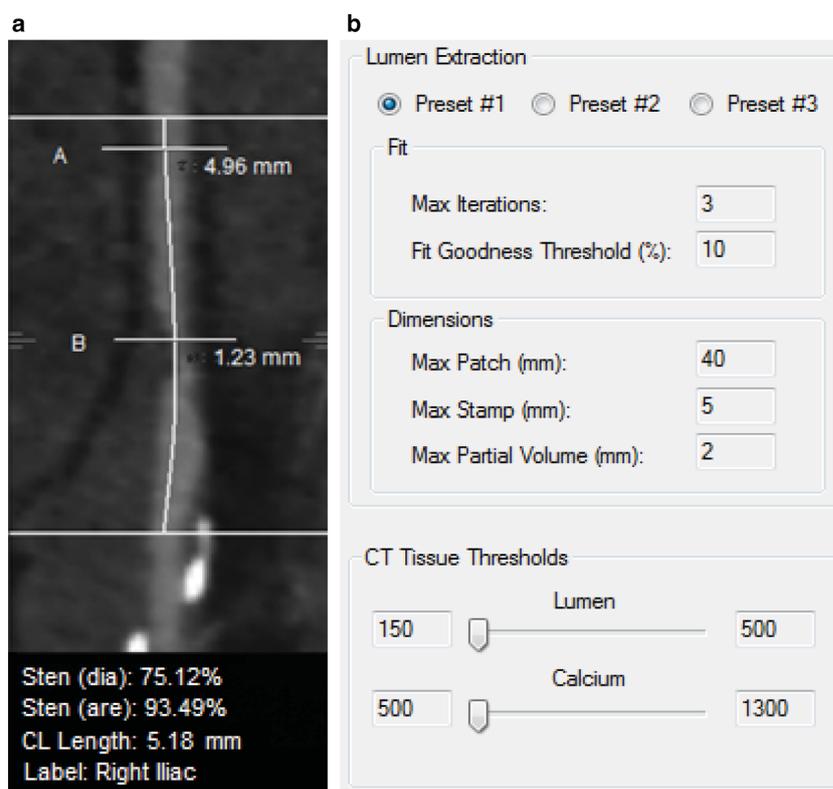
lumen (see section “[Hypothesis Tests](#)”); increasing it will also filter out more portions of potential lumen.

- ★ **Threshold group:** increasing the minimum lumen threshold will have the effect of decreasing the overall lumen area where the algorithm will operate (and vice versa). The minimum calcium threshold defines simultaneously the maximum lumen threshold and hence can be used to define the borderline between these two regions.

Three sets of default values or presets for the algorithm were also included. Figure 9b shows the values used for the tests made in this work wrapped under the one called “Preset# 1.” Two extra presets were provided to roughly adapt the algorithm to vessels of smaller size; however, more presets can be included to group vessels not only by size but also by level of fatty deposits, age, or gender among others [18].

Finally, for cases when the segmentation result is ultimately not appropriate for the user, a contour correction

**Fig. 9** Parameter adjustment and measurement tool. **a** Curved planar reformation of a tracked section of 5.18 mm on stack  $S_{03}$  with overlay of stenosis measurement: point *A* is the reference or healthy point and point *B* is the obstructed or stenosed point. Points *A* and *B* are automatically highlighted for a chosen section of the vessel or can also be marked interactively. The stenosis ratio using both the diameters and the area is shown in the bottom as well the length of the chosen vessel section and a user selected label **b** User interface element allows for adjustment of lumen extraction parameters and tissue thresholds



tool was incorporated to manually correct any section of the contour found over a particular slice (see Fig. 10).

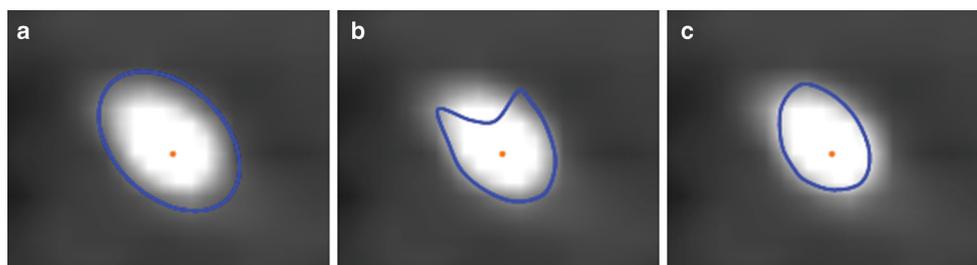
#### Datasets

To validate different aspects of the algorithm, a total of 30 anonymized CT angiography runoff scans were employed (Toshiba Aquilion™) with 120 kVp, 250 mA, 5–15 ms scan time, 1.0 mm slice thickness,  $512 \times 512$  rows  $\times$  columns,  $0.781 \times 0.781$  pixel spacing, field-of-view of 400 mm, and 16 bits samples per pixel. The average number of slices per stack was 1,440. All the studies contained stenosed sections ranging from subtle to obvious from the base of the aorta throughout the iliacs and its corresponding minor branches.

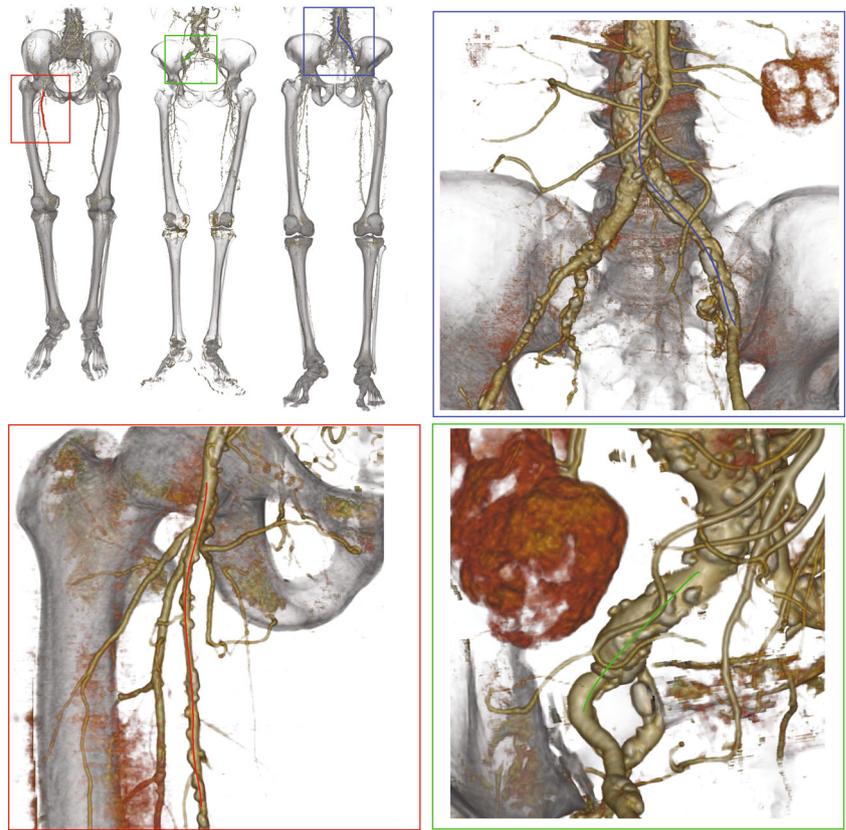
The studies were roughly grouped according to its visible level of cavity distortion from low to high (low:  $S_{01}$  to  $S_{10}$ , medium:  $S_{11}$  to  $S_{20}$ , and high:  $S_{21}$  to  $S_{30}$ ). Figure 11 shows a representative case for each group.

A radiologist with experience on abdominal aortic aneurysm repairs assisted in selecting vessel sections of 15 cm of length on each study. These sections contained 160 planes or slices to analyze and 4,800 in total ( $160 \times 30$  studies). For each section, he manually delineated both the outer wall and the lumen across each of the planes using the standard electronic measurement tools (i.e., caliper, ruler, and polyline) provided by the built-in software tool provided by the scanner manufacturer. The outer wall maximum diameters ranged between 2 and 40 mm which offered a wide

**Fig. 10** Contour correction. **a** A previous ellipse is taken as the contour of a failed fit due to full calcium obstruction. **b** A user can modify the ellipse by simply dragging over the line contour with the mouse to obtain the desired contour shown in (c)



**Fig. 11** Volume rendering of image sets  $S_{03}$  (red),  $S_{15}$  (green), and  $S_{28}$  (blue). Each colored section corresponds to the section of the vessel chosen to run the algorithm. Notice that all samples taken have a certain degree of cavity distortion along the path



range of measurements to test. For each vessel segment defined above, our lumen detection algorithm was tested for performance, stability, and accuracy as detailed in the section “Results”.

**Results**

Performance Tests

For this test, each of the 30 segments of vessel was evaluated using two types of commodity hardware: Type I was a 2.10-GHz dual-core CPU with 4 GB of RAM and Type II was a 2.40-GHz quad-core CPU with 4 GB RAM under 64-bit Windows operating systems. Execution times are shown in Table 2 for both platforms.

These times approximate slightly less than 2 ms/cm along the centerline in the Type I hardware and slightly more

than 1 ms/cm along the centerline in the Type II hardware which makes it more than adequate for real-time changes in tissue threshold and/or in the parameters of the algorithm.

Table 3 shows the execution times for referential cases  $S_{03}$ ,  $S_{28}$ , and  $S_{28}$ . Stack  $S_{15}$  is a typical example of a complicated case due to the area being swollen and the sheer nature of the calcifications present (as seen in Fig. 11) hence showing the slowest performance; however, an execution time of 34 ms is still more than adequate to be used on interactive applications. Cases like  $S_{28}$  are similar to  $S_{15}$ ; only the vascular structure was thinner; cases like  $S_{03}$  showed the fastest performance even though they had a high presence of calcium artifacts since they are also thinner.

Hypothesis Tests

For this part of the evaluation, our tracking algorithm was first executed along the image slices that were included on the 15-cm segments previously defined by the radiologist in order to extract a centerline. Following, each of these points were automatically fetched to the lumen detection algorithm (see Algorithm 1).

Using the radiologists annotations as the gold standard, image intersections were computed between the lumen areas automatically extracted by our algorithm and the ones manually annotated. Quantitative indicators such as

**Table 2** Execution times in milliseconds for the 4,800 contours evaluated

Hardware type	Min	Max	Avg	Std
Type I	3	150	24	7.67
Type II	2	65	18	4.89

**Table 3** Execution times in milliseconds for stacks  $S_{03}$ ,  $S_{15}$ , and  $S_{28}$

Study <sub>id</sub>	Type I				Type II			
	Min	Max	Avg	Std	Min	Max	Avg	Std
$S_{03}$	2	39	4	0.53	1	11	3	0.19
$S_{15}$	3	121	40	20.10	1	104	34	9.85
$S_{28}$	5	73	22	11.24	3	59	20	5.33

true positives rate (TPR or sensitivity), false positive rate (FPR), and positive predictive value (PPV) were computed as shown in Fig. 12.

Table 4 shows the average and standard deviation for the sensitivity, FPR, and PPV (Global column); studies  $S_{03}$ ,  $S_{15}$ , and  $S_{28}$  are also included. Figure 12 shows that the blue areas marked by the specialist approximated an ellipsoid which does not necessarily include all the lumen areas or may also contain certain bits of calcium. Despite this fact, the high level of sensitivity seen in the global column of Table 4 is a natural consequence of the fact that most of the pixel contours manually annotated were contained within the area of the ones produced by the automatic method; this means that only a small proportion of blue areas were left out in the intersection image.

Figure 13a shows the relationship between sensitivity and PPV, with the PPV showing a high dispersion. However, Fig. 13b showed a much higher variability of the FPR against sensitivity. Further analysis of these estimators against the average diameter per study (Fig. 13c, d and e) confirmed the highest rate of variability around the 5-mm mark. This is especially noticeable in slices similar to the one depicted in Fig. 14a where the algorithm extracted a contour more similar to the wall than the lumen (more false positives and almost no true negatives) yielding a high FPR.

**Table 4** Sensitivity, false positive rate, and positive predictive values for all image sets compared to the representative cases  $S_{03}$ ,  $S_{15}$ , and  $S_{28}$

	Global	$S_{03}$	$S_{15}$	$S_{28}$
Sensitivity				
Avg	0.9809	0.9643	0.9864	0.9981
Std	0.0085	0.0065	0.0052	0.0064
FPR				
Avg	0.2483	0.9510	0.2100	0.1520
Std	0.2541	0.2929	0.0523	0.0617
PPV				
Avg	0.9170	0.8847	0.8926	0.9270
Std	0.0362	0.0346	0.0260	0.0188

To improve segmentation results for these cases, the algorithm was re-executed changing the default values of two parameters as shown in Fig. 13b: maximum partial volume width (max partial volume) and lumen threshold. These parameters are the most relevant to avoid over segmentation due to high incidence of partial volume effects, especially within small areas. Figure 13e shows the effects of increasing the minimum lumen threshold value to 200 HU (without altering the max partial volume) and Fig. 13f, g show the effects of reducing the max partial volume size to 1.5 and 1 mm respectively using the original threshold values for the lumen. Figure 13g showed the best agreement according to the radiologist and Fig. 13h depicts the new intersection image where the gray areas represent a much higher number of true negatives and the red areas a smaller number of FPs. This is also reflected in Fig. 13i where all FPR values below the 12-mm mark were reduced and also yielded less dispersion. Table 5 summarizes these results numerically.

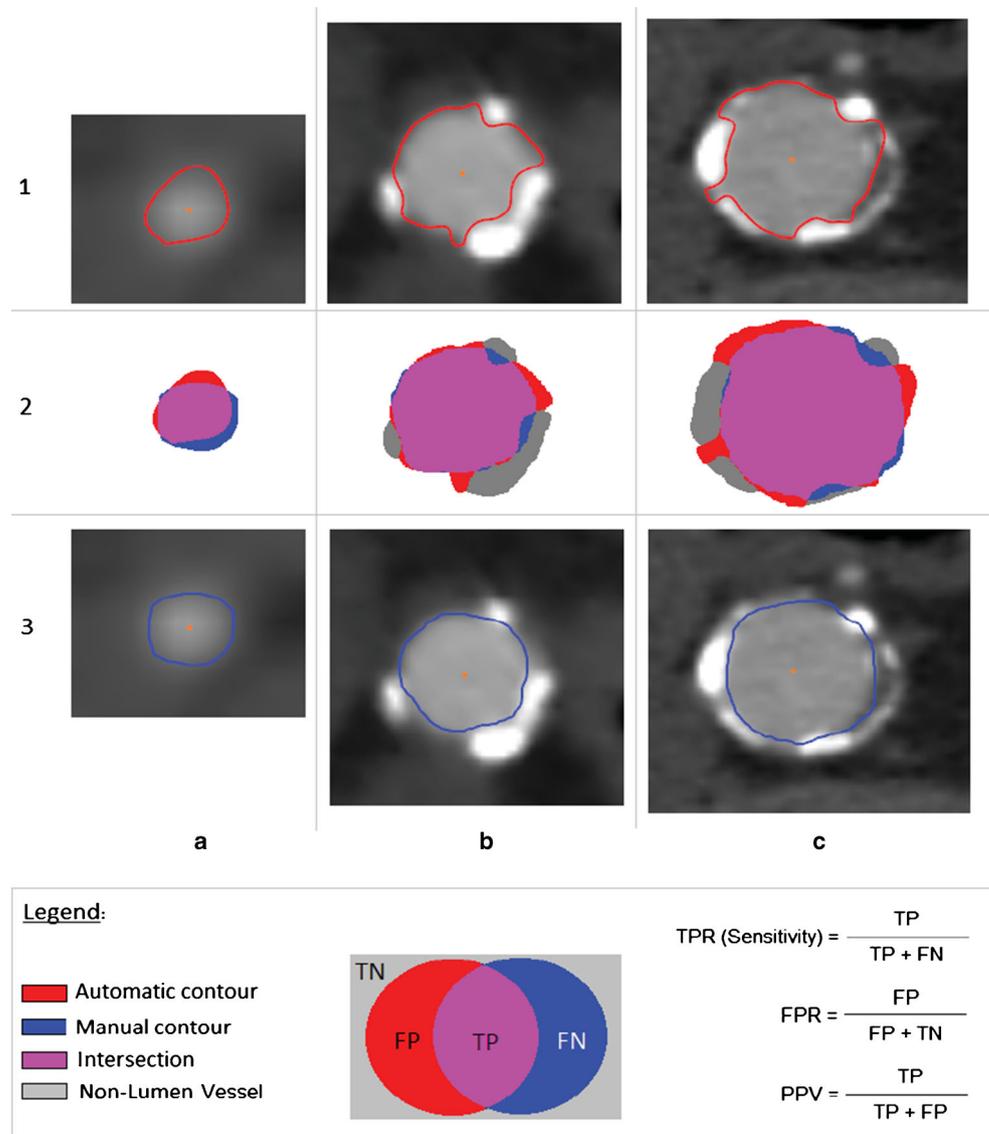
Figure 13f, j showed a natural improvement for the PPV values due to the reduction of false positives. It is also worth mentioning here is that the rest of the estimators did not show significant changes.

Reproducibility Tests

The last part of the evaluation was designed to ensure the stability of the algorithm. For this purpose, a synthetic image stack of  $512 \times 512 \times 150$  pixels with a pixel size of  $1 \text{ mm}^3$  was used. The object represented is a cylinder of 15 cm of length with a radius of 18 mm as depicted in Fig. 15 where eight different centerline axes described by the yellow spots were taken.

The algorithm was run for each of the eight axial axes defined by the yellow points across the same cut planes and the diameters obtained were compared against the theoretical value. Also a visual inspection of the diameters produced was done to ensure that the contours were generated in the appropriate location. The diameters obtained for all cases were also 18 mm showing no variations among the eight different runs. The average execution time for the 18 contours in total is shown on Table 6.

**Fig. 12** Image intersections computed for hypothesis testing. Areas for stack  $S_{03}$  correspond to column  $a$ ,  $S_{15}$  to column  $b$ , and  $S_{28}$  to column  $c$ . Row 1 represents the contours obtained by the algorithm, row 3 the contours manually delineated by the specialist, and row 2 the intersection between both. As indicated by the legend, TPR, FPR, and PPV values were computed assuming red represents the false positives (FP), blue the false negatives (FN), magenta the true positives (TP), and gray the true negatives (TN)



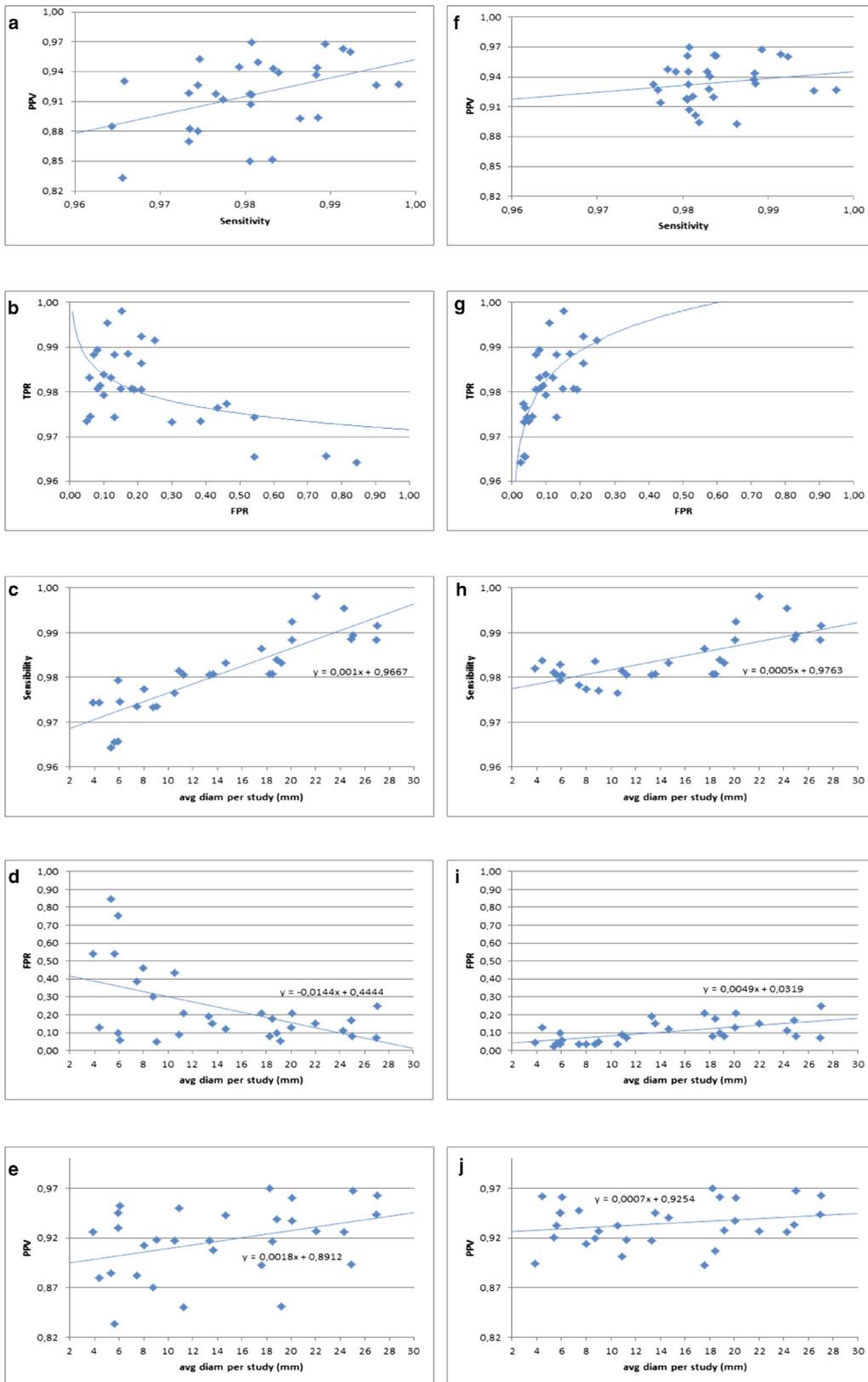
**Discussion and Conclusions**

We presented and evaluated a novel and fully automated algorithm for the extraction of cavity features on images of human vessels. Two phases were designed and efficiently implemented in order to: (1) avoid major leakages to pixels of nearby tissue and (2) obtain a precise and consistent contour even in the presence of the typical image artifacts generated by calcium.

Hypothesis tests were performed between the contours generated by the algorithm and the ones manually delineated by a specialist (golden standard) obtaining a sensitivity of 98.39 %, false positive rate of 8.99 %, and PPV of 93.67 % after doing minor adjustments on the parameters of the algorithm. Sensitivity showed that the areas marked by the specialist that were not identified by the algorithm fell under less than 2 % of the cases and showed small variations

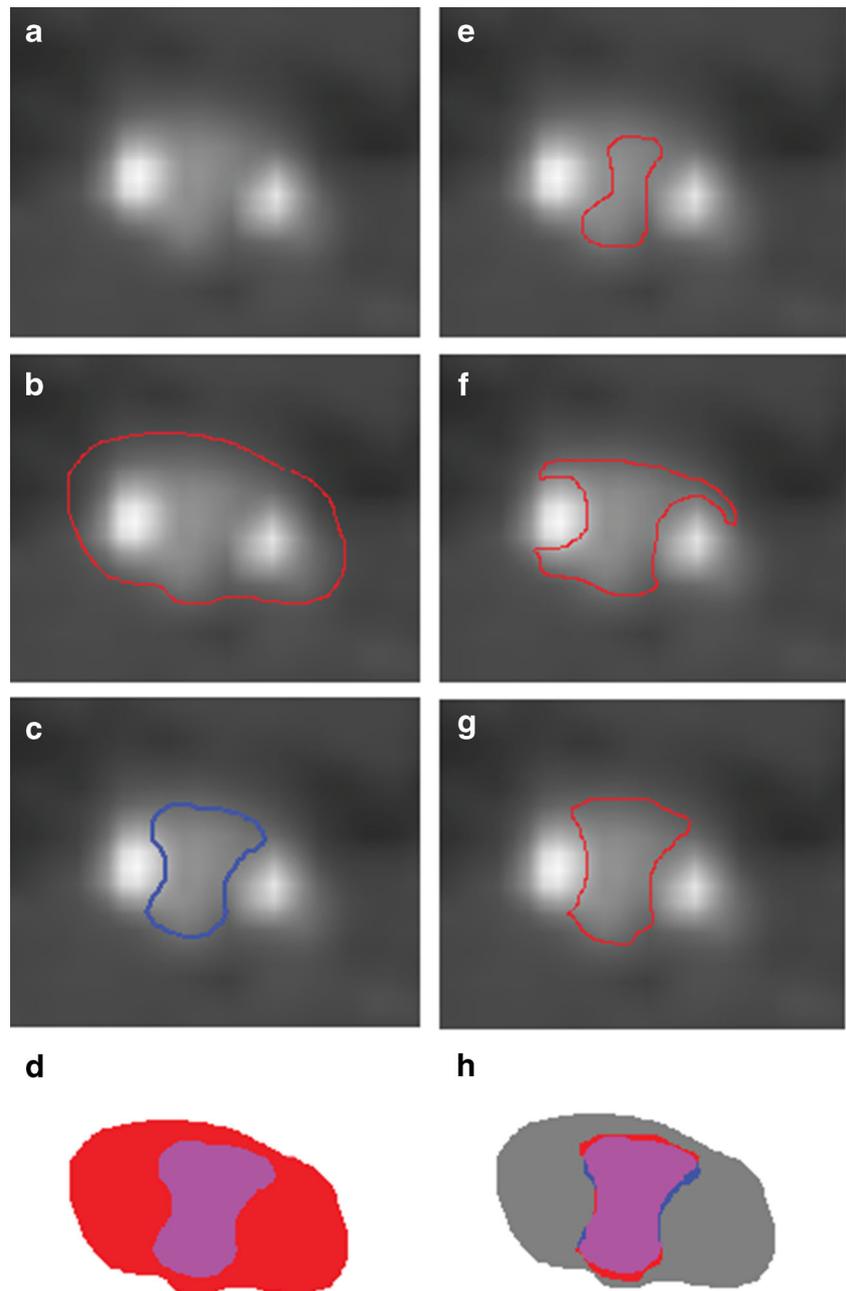
among the test data overall. FPR and PPV showed a higher rate of variability due to the specific effect of partial volume on smaller diameters. The algorithm showed no variations in diameter computations against well-established dimensions of a synthetic image set when selecting eight different centerline axes, proving its stability.

Execution time did not exceed 2 ms/cm of centerline length for the worst case which makes the algorithm more than suitable for use on interactive applications. This was demonstrated as part of a parameter adjustment and measurement tool where minimum and maximum diameters are automatically marked over a CPR rendering of the section of the vessel. The length of the section and the stenosis ratio are also automatically computed. Both the reference and the stenosis marks can be manually adjusted along the selected vessel section and the ratio is automatically recomputed on the fly. These visual tools integrated with the parameter



**Fig. 13** Sensitivity (TPR), FPR, and PPV values using default and modified parameters for the algorithm. *Left column (a, b, c, d, e):* hypothesis estimators computed using default algorithm values; *Right column (f, g, h, i, j):* hypothesis estimators computed after setting max partial volume to 1 mm

**Fig. 14** Effects of partial volume for small diameters: **a** Original image slice. **b** Initial algorithm estimation. **c** Manual annotation. **d** Intersection of **(b)** and **(c)**. **e** Estimation after increasing the minimum lumen value. **f** and **g** Estimation after reducing the maximum partial volume size. **h** Intersection between **(c)** and **(g)** shows the best agreement



options allowed for quick and effective changes among the different types of acquisitions studied.

Our algorithm presents a unique combination of some of the principles of classic techniques while trying to avoid its main drawbacks. For example, methods that employ active contours rely on the definition of a constraint function which is normally difficult to establish and may lead to instabilities of convergence and tend to be very sensitive to initialization [1, 3, 20]. Region filling methods on the other hand are less sensitive to initialization but very few are adapted to geometrical constraints efficiently [22].

Our region filling approach acts as a brush that repulses shape distortions while maintaining elliptical edges and at the same time is not susceptible to initialization issues. Additionally, since there are no convergence side effects adjusting any of the parameters of the algorithm does not significantly affect the speed of any recomputation.

Methods that rely on the application of multi-scale filters or the use of edge-based filters are applied globally and can be time consuming. We developed our own local filter which is applied only over potential leakage areas using an

**Table 5** Sensitivity, FPR, and PPV values after adjusting algorithm parameters

	Global	S <sub>03</sub>	S <sub>15</sub>	S <sub>28</sub>
Sensitivity <sub>adjusted</sub>				
Avg	0.9839	0.9806	0.9864	0.9984
Std	0.0054	0.0025	0.0033	0.0064
FPR <sub>adjusted</sub>				
Avg	0.0899	0.0239	0.2100	0.1521
Std	0.0565	0.0094	0.0523	0.1372
PPV <sub>adjusted</sub>				
Avg	0.9367	0.9206	0.8926	0.9270
Std	0.0202	0.0157	0.0257	0.0188

**Table 6** Execution times in milliseconds for the synthetic image set

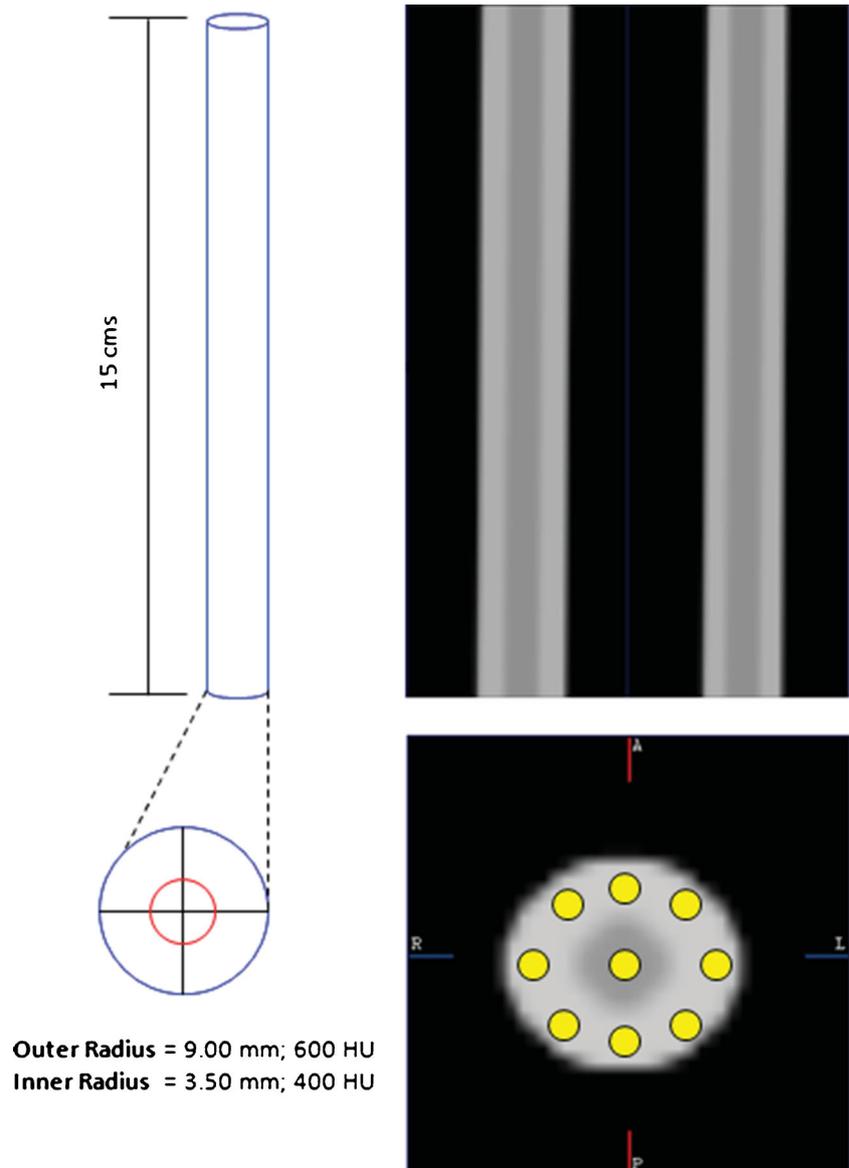
Hardware type	Avg	Max	Std
Type I	4	0.10	0.000012
Type II	3	0.09	0.000011

effective pattern recognition approach that avoids the usual time consuming effects of a large scale filter [15].

Most methods also require the initial point to be inside the vessel; in our case, we require the point to be either inside or in a neighborhood which means that our final segmentation can be used to re-center a failed tracked centerline point with respect to the lumen automatically.

For the particular set of studies tested, we demonstrated that by doing minor adjustments on the parameters of the

**Fig. 15** Synthetic image set used for stability tests. *Left side*, scheme of the theoretical measures (*length, outer, and inner ellipse*). *Right side*, Multi-planar reconstruction rendering of the cylinder from sagittal, coronal, and axial angles. The zoomed axial cut shows the eight yellow points traversed by the parallel centerlines used for the test



algorithm, we were able to extract the lumen properly covering a wide extent of diameters. However, for other types of studies, a few runs of the algorithm may be required in order to establish the optimal parameters. It was also shown with the default set of parameters and presets defined that the algorithm required some fine tuning with vessels between 2 and 6 mm. More tests with these type of vessels will be done as the ellipse fit may be even skipped directly to Stage II. Also tests with noisy images will be done to establish performance impact on the fitting stage.

Currently, the algorithm takes a complete set of centerline points as input, this was designed specifically to focus on extracting a precise lumen contour regardless of the centerline tracking implementation. An advantage of this approach is that lumen extraction parameters can be fine tuned without recomputing the centerline. However, overall performance of the segmentation process can drastically drop from a poor tracking implementation. Bringing together a closer integration between these two task could help to alleviate this situation.

Other tests that might be incorporated to further establish the precision of our lumen estimation include: a comparison between the stenosis ratios included in the parameter adjustment and measurement tool versus a manual estimation of this ratio by a specialist, the effects of tortuosity and the addition of more observers. The algorithm can also benefit from an automatic estimation of optimal default threshold values for the threshold group shown in Fig. 9.

Although the edge list obtained on Stage I was only used as the input for the specific set of steps performed on Stage II, the micro-dilation and edge-walking concepts could also be directly applied to the boundary mask in order to extract and display an additional contour which approximates the outer wall. Following this idea, a modified version of the algorithm could also be implemented to target a more precise outer wall contour and integrate a more broad set of tools as done in [28].

Also worth noting is that although the algorithm was tested with specific image stacks that offered a steady range of thresholds for segmentation, it can be easily adapted to vessels of other areas of the anatomy as well as other types of acquisitions protocols and modalities also popular in vascular diagnosis such as magnetic resonance angiography (MRA). This concept was demonstrated by providing three groups of values called presets which can be expanded to include vessel types using more thorough classifications such as the one shown in [18] and could also be presented to the user prior analyzing a vessel or even during the loading of the image set.

**Acknowledgments** This work was partially funded by the Scientific and Humanistic Development Council (CDCH) of the Central University of Venezuela (UCV) and Biotronics3D Ltd., UK. The image sets employed were kindly provided by Biotronics3D Ltd. All

images were generated with the rendering engine for advanced medical visualization courtesy of Biotronics3D Ltd.

## References

1. Abdel-Dayem AR: Detection of arterial lumen in sonographic images based on active contours and diffusion filters. In: Proceedings of the 7th International Conference on Image Analysis and Recognition - Volume Part II, ICIAR'10. Springer-Verlag, Berlin, 2010, pp 120–130
2. Aylward S, Pizer S, Eberly D, Bullitt E: Intensity ridge and widths for tubular object segmentation and description. In: Proceedings of the 1996 Workshop on Mathematical Methods in Biomedical Image Analysis (MMBIA '96). IEEE Computer Society, Washington, DC, USA, 1996, pp 131–138
3. Baltaxe Milwer M, Flórez Valencia L, Hernández Hoyos M, Magnin I, Orkisz M: Fast marching contours for the segmentation of vessel lumen in CTA cross-sections. In: Conference Proceedings IEEE Engineering Medicine Biology and Society. IEEE, IEEE, Lyon, France, 2007, pp 791–794
4. Bitter I, Kaufman AE, Sato M: Penalized-distance volumetric skeleton algorithm. *IEEE Trans Vis Comput Graph* 7(3):195–206, 2001
5. Bladin C, Alexandrov A, Murphy J, Maggiano R: Carotid stenosis index. a new method of measuring internal carotid artery stenosis. In: 19th International Conference on Stroke and the Cerebral Circulation. American Heart Association, San Diego, USA, 1995, pp 230–234
6. Boskamp T, Rinck D, Link F, Kümmerlen B, Stamm G, Mildenerger P: New vessel analysis tool for morphometric quantification and visualization of vessels in ct and mr imaging data sets. *Radiographics* 24(1):287–297, 2004
7. Buhler K, Felkel P, Cruz A: Geometric methods for vessel visualization and quantification - a survey. Vienna: Tech. Rep. 035, VRVis Research Center, 2002
8. Canny J: A computational approach to edge detection. *IEEE Trans Pattern Anal Mach Intell* 8(6):679–698, 1986
9. Flórez L, Azencot J, Maciej O: Algorithm for blood-vessel segmentation in 3d images based on a right generalized cylinder model: application to carotid arteries. In: Proceedings of the 2010 International Conference on Computer Vision and Graphics: Part I, ICCVG'10. Springer-Verlag, Berlin, 2010, pp 27–34
10. Frangi A, Niessen W, Hoogeveen R, van Walsum T, Viergever M: Model-based quantitation of 3-d magnetic resonance angiographic images. *IEEE Trans Med Imaging* 18(10):946–956, 1999
11. Geiger D, Gupta A, Costa LA, Vlontzos J: Dynamic programming for detecting, tracking, and matching deformable contours. *IEEE Trans Pattern Anal Mach Intell* 17:294–302, 1995
12. Kanitsar A, Fleischmann D, Wegenkittl R, Felkel P, Gröller E: CPR—curved planar reformation. In: IEEE Visualization, 2002, pp 37–44
13. Kirbas C, Quek F: A review of vessel extraction techniques and algorithms. *ACM Comput Surv* 36(2):81–121, 2004
14. Lesage D, Angelini E, Bloch I, Funka-Lea G: A review of 3d vessel lumen segmentation techniques: models, features and extraction schemes. *Med Image Anal* 13(6):819–845, 2009
15. Manniesing R, Viergever MA, Niessen WJ: Vessel enhancing diffusion: a scale space representation of vessel structures. *Med Image Anal* 10(6):815–825, 2006
16. McInerney T, Terzopoulos D: Deformable models in medical image analysis: a survey. *Med Image Anal* 1:91–108, 1996
17. Mille J, Cohen LD: Deformable tree models for 2d and 3d branching structures extraction. In: IEEE Computer Society Conference

- on Computer Vision and Pattern Recognition Workshops, 2009, pp 149–156
18. Mortensen J, Talbot S, Burkart J: Cross-sectional internal diameters of human cervical and femoral blood vessels: relationship to subject's sex, age, body size. *Anat Rec* 226:115–124, 1990
  19. Prewitt JMS: Object enhancement and extraction. In: *Picture Processing and Psychopictorics*, 1970, pp 75–149
  20. Rivest-Henault D, Cheriet M, Deschenes S, Lapierre C: Length increasing active contour for the segmentation of small blood vessels. In: *Proceedings of the 2010 20th International Conference on Pattern Recognition, ICPR '10*. IEEE Computer Society, Washington, DC, USA, 2010, pp 2796–2799
  21. Savchenko V, Pasko EA, Okunev OG, Kunii TL: Function representation of solids reconstructed from scattered surface points and contours. *Comput Graph Forum* 14:181–188, 1995
  22. Sekiguchi H, Sugimoto N, Eiho S, Hanakawa T, Urayama S: Blood vessel segmentation for head MRA using branch-based region growing. *Syst Comput Japan* 36(5):80–88, 2005
  23. Sethian JA: Theory, algorithms, and applications of level set methods for propagating interfaces. *Acta Numerica* 5:309–395, 1996
  24. Smith B, Boyle J, Dongarra J, Garbow B, Ikebe Y, Klema V, Moler C: Matrix eigensystem routines—EISPACK guide. *Lect Notes Comput Sci* 6:26–27, 1976
  25. Sobel I, Feldman G: A  $3 \times 3$  isotropic gradient operator for image processing (1973). In: Duda R, Hart P Eds. Presented at a talk at the Stanford Artificial Project in 1968, unpublished but often cited, orig. in *Pattern Classification and Scene Analysis*. Wiley, 1973, pp 271–272
  26. Stricker M: A new approach for robust ellipse fitting. In: *International Conference on Automation, Robotics and Computer Vision–ICARCV 1994*, vol 2, 1994, pp 940–945
  27. Sutton D: *Textbook of Radiology and Imaging*, 7th edition Churchill Livingstone, 2002, pp 265–509
  28. Vukadinovic D, van Walsum T, Manniesing R, Rozie S, Hameeteman R, de Weert T, van der Lugt A, Niessen WJ: Segmentation of the outer vessel wall of the common carotid artery in CTA. *IEEE Trans Med Imaging* 29(1):65–76, 2010
  29. World Health Organization: Fact sheet no.317, 2013. Website <http://www.who.int/mediacentre/factsheets/fs317/en/index.html>
  30. Xavier RT, Orriols X, Radeva P, Binefa X, Vitrià J, Cañero, C, Villanueva JJ: Eigensnakes for vessel segmentation in angiography. In: *International Conference on Pattern Recognition*. IEEE, 2000, pp 340–343
  31. Zimmermann K, Svoboda T: Approximation of euclidean distance between point from ellipse. Research Report CTU–CMP–2005–23. Center for Machine Perception, vol K13133. FEE Czech Technical University, Prague, Czech Republic, 2005