Dissected aorta segmentation using convolutional neural networks

Tianling Lyu, Guanyu Yang, Xingran Zhao, Huazhong Shu, Limin Luo, Duanduan Chen, Jiang Xiong, Jian Yang, Shuo Li, Jean-Louis Coatrieux, et al.

Highlights:

- An algorithm for dissected aorta segmentation based on 3-D and 2-D convolutional neural networks.
- A 3-D CNN model identifies the proximal and distal regions of the dissected aorta data.
- A 2-D CNN model for aorta segmentation which incorporates boundary information.

Dissected Aorta Segmentation using Convolutional Neural Networks

Tianling Lyu[1], Guanyu Yang[1], Xingran Zhao[1], Yang Chen[1*], Huazhong Shu[1],

Limin Luo[1], Duanduan Chen[2], Jiang Xiong[3], Jian Yang[4], Shuo Li[5],

Jean-Louis Coatrieux[6]

[1] Laboratory of Imaging Science and Technology, Southeast University, Nanjing, China

[2] Department of Biomedical Engineering, Beijing Institute of Technology, Beijing, China

[3] The Chinese PLA Hospital, Beijing, China

[4] School of Optoelectronics, Beijing Institute of Technology, Beijing, China

[5] Digital Imaging Group of London, London, Canada

[6] Université de Rennes 1, Rennes, France

[*] Corresponding author

E-mail: chenyang.list@seu.edu.cn

Abstract

Background and objective: Aortic dissection is a severe cardiovascular pathology in which an injury of the intimal layer of the aorta allows blood flowing into the aortic wall, forcing the wall layers apart. Such situation presents a high mortality rate and requires an in-depth understanding of the 3-D morphology of the dissected aorta to plan the right treatment. An accurate automatic segmentation algorithm is therefore needed.

Method: In this paper, we propose a deep-learning-based algorithm to segment dissected aorta on computed tomography angiography (CTA) images. The algorithm consists of two steps. Firstly, a 3-D convolutional neural network (CNN) is applied to divide the 3-D volume into two anatomical portions. Secondly, two 2-D CNNs based on pyramid scene parsing network (PSPnet) segment each specific portion separately. An edge extraction branch was added to the 2-D model to get higher segmentation accuracy on intimal flap area.

Results: The experiments conducted and the comparisons made show that the proposed solution performs well with an average dice index over 92%. The combination of 3-D and 2-D models improves the aorta segmentation accuracy compared to 3-D only models and the segmentation robustness compared to 2-D only models. The edge extraction branch improves the DICE index near aorta boundaries from 73.41% to 81.39%.

Conclusions: The proposed algorithm has satisfying performance for capturing the aorta structure while avoiding false positives on the intimal flaps.

Keywords: aorta dissection, computed tomography, deep learning, image segmentation

1. Introduction

Aortic dissection (AD) is a severe pathological condition in which an injury to the intimal layer of the aorta allows blood flow between layers of the aortic wall, forcing the layers apart [1]. The incidence of AD is often associated with high blood pressure and low blood vessel wall strength caused by injuries, previous heart surgeries and diseases including Marfan's Syndrome and bicuspid aortic valve [1, 2]. The Stanford system classifies aortic dissections into two groups based on their anatomical locations. Stanford type-A dissection includes the ascending aorta and possibly the aortic arch and the descending aorta, while Stanford type-B concerns the descending aorta or the arch without the involvement of the ascending aorta [3]. Despite the low morbidity, aortic dissection causes quick deaths with high mortality rate. Half of the patients with acute type-A aortic dissection are expected to be dead in 3 days without treatment [4] and at least 10% in 30 days for type-B [5].

Radiological diagnosis of AD is mainly based on computed tomography angiography (CTA). As shown in Fig.1(a), the intimal flap separates the aorta into true and false lumen. The analysis of the morphological features of the dissected aorta, such as the size and location of the primary entry, the diameters of true and false lumen as well as the curvature of the aorta, is of great importance for diagnosis, personalized medical planning and risk evaluation [6]. Accurate measurements in the region near the aortic arch are particularly challenging and a 3-D reconstruction of the aorta based on CTA datasets is needed in order to fulfil these clinical requirements.

Features extraction in dissected aorta relies on a precise 3D delineation of the aorta segments and structures of concern. To avoid manual and time-consuming operations, a fully automatic segmentation must be available. It will not only provide a reliable way to get the relevant lesion characteristics but also, through 3D rendering and manipulation, the basic assistance for treatment planning and advanced simulations. For instance, a few studies investigated the hemodynamic properties of the dissected aorta and provided predictions on post-stenting results [7-9]. Such simulations require not only an accurate segmentation but also a post-processing to get a regular mesh to represent the structures involved.

Traditional aorta segmentation algorithms try to model vascular structures in a mathematical way. One approach commonly used is enhancing vascular structures using a Hessian-matrix-based filter. Frangi et al. [10] proposed Frangi filter for vessel enhancement. Manniesing et al. [11] improved the performance of Frangi filter by following a diffusion step. Another class of algorithms extracts vessel centerlines at first, which define the vascular topology, and is followed by a segmentation based on the extracted centerline. Deschamps and Cohen proposed a minimal cost path method for centerline extraction [12]. Chen et al. introduced a back-tracking operation to the minimal cost path searching in [13]. Researchers use geometric models to segment vascular structures as well. Kayikcioglu and Mitra [14] proposed an algorithm in which cylinders were used to model vessels. Wörz and Rohr [15] proposed a 3-D cylindrical intensity model for human vessel segmentation and quantification. For dissected aorta analysis, Krissian et al proposed a semi-automatic aorta dissection wall segmentation method in [16], which was based on previously extracted centerline features. Cattin et al [17] combined Hough transformation and an arch model to segment dissected aorta arch. Lee et al. [18] proposed a wavelet-based algorithm for true-false lumen segmentation and Fetnaci et al. And deformable models are used in [19] to fulfill the same purpose.

Deep learning (DL) architectures, inherently built on multiple linear and non-linear transformations of data, have achieved remarkable success in wide computer vision (CV) tasks. Convolutional neural network (CNN) is one of the most commonly used deep learning architecture in CV. A recent survey [20] has shown that CNN already attracted a growing attention in the medical imaging field for tasks ranging from segmentation, registration to classification and others. It is widely agreed that deep neural networks are effective in the tasks of feature representation, recognition and classification et, al [21-28]. For image segmentation, deep convolutional neural networks based on Fully Convolutional Network (FCN) [29] show striking improvements over those systems relying on hand-crafted features. Recently, encoder-decoder based networks were developed by using an encoder module that reduces feature map size to capture higher semantic information, and then a decoder module is applied to recover the spatial information [30, 31]. In [32], U-net based networks with added skip connections between encoder and decoder were proposed to enhance segmentation precision. DeepLab models [33-36] use atrous convolution to enlarge the reception field without pooling layers. In [37], a network termed PSPNet was introduced by adding a spatial pyramid pooling at several grid scales for multi-scale object segmentation task .

Though a 3-D CNN appears a natural choice for segmenting volume data [38], the limited GPU memory introduces a dilemma between layer depth and input data volume size. Reducing the 3-D image resolution can be considered an option but at the cost of segmentation accuracy degradation and clinical feature missing. On the other hand, though 2-D CNNs allow segmenting images at high resolution, the Z-axis tissue information tend to be lost. In this work, we propose a CNN-based strategy to give effective dissected aorta segmentation, which consists of the training of three CNN models:

1) Train a 3-D CNN model which identifies the proximal and distal regions of the dissected aorta data. Two portions with distinct topologically features in the aorta can be obtained by applying the trained 3-D CNN model on the dissected aorta data.

2) Train two 2-D CNN models using the data from the above partitioned aorta portions independently. The trained two 2-D CNN models are then applied on each slice within the corresponding portions to fulfill the final segmentation task.

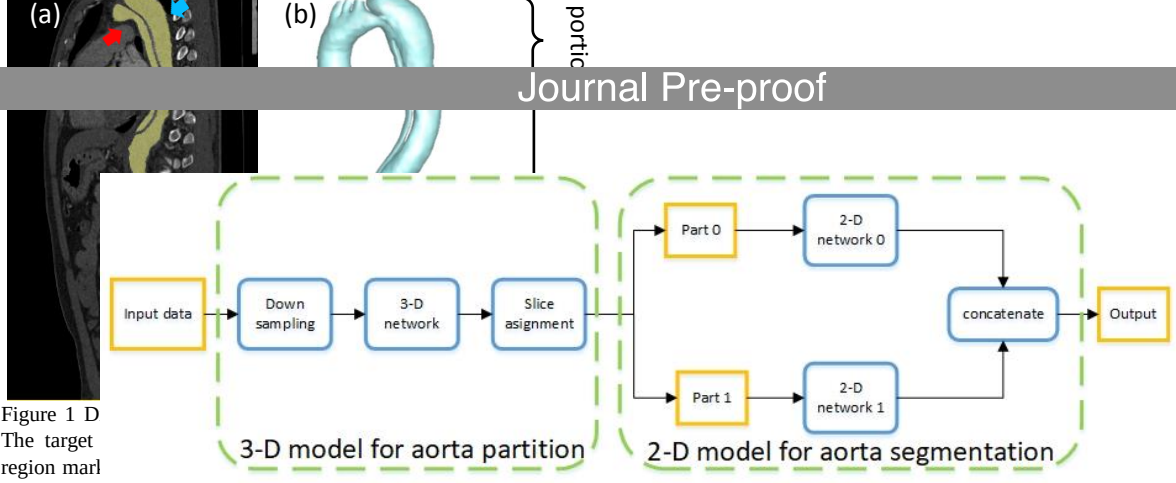

Figure 1 D
The target
region marl
region mark
3-D dissec
different sh

Figure 2 The overall view of the proposed algorithm. Yellow squares represent data while blue squares represent the processing procedures. The input data firstly pass through a 3-D model including down-sampling, 3-D CNN and slice labelling of portion A and portion B. Slices in portion A and portion B separately pass through two 2-D CNNs. The outputs of two 2-D CNNs are concatenated together to provide the final segmentation output.

The main contributions of this work include the following two points. Firstly, we utilized a combination of 3-D and 2-D models to improve aorta segmentation reliability. Secondly, an edge extraction branch was added to the 2-D model to get higher segmentation accuracy on intimal flap area.

The rest of the paper is structured as follows. In Section 2, we describe the proposed algorithm and the detailed implementation. Quantitative performance assessments are provided in Section 3. Conclusion and future work are summarized in Section 4.

2. Material and methods

The compromise we propose in this paper takes into consideration this problem and proposes a solution by making use of the specific shape of the aorta (an aorta arc at top and a straight descending aorta below). Therefore, with a 3-D CNN model, the volume is separated into two portions: (i) the aorta arch including the ascent part (referred as portion B); (ii) the remaining descent aorta portion (referred as portion A). The illustration is given in Fig. 1. The CTA slices in portion B contain two aorta intersections or long ellipses while, in portion A, only one intersection exists. After this, two 2-D CNN models are respectively applied upon the slices of portion A and portion B to get an accurate vessel segmentation. Moreover, we pick out 10 slices around the partition slice in each training volume. These slices are included in the both two training sets for the two 2-D CNNs in case that the 3-D partition is not accurate. The complete processing pipeline is outlined in Fig. 2.

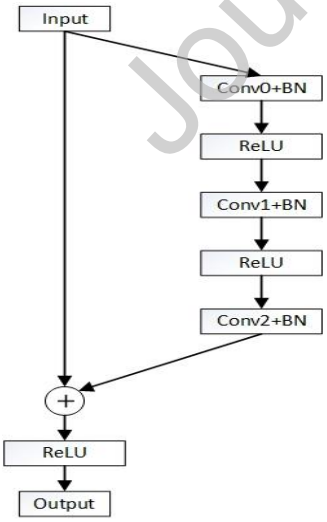

Figure 3 Illustration of a residual block.

2.1 3-D model for aorta partition

The 3-D model consists of three steps. Firstly, the 3-D volumes are down-sampled due to memory limitations. Secondly, a 3-D CNN is utilized to label each portion of the aorta. Finally, a few operations are performed to recover the original volume length in z direction.

2.1.1 data down-sampling

Datasets may have different spatial resolutions according to the CTA imaging system or clinical protocol used. This problem is solved by linear interpolating images into a uniform low resolution. The interpolated volumes has the same voxel size of 2.0x2.0x2.0 mm^3. Thus, any input CTA volume defined with voxel number of $n_x \times n_y \times n_z$ and voxel size $d_x \times d_y \times d_z$ is interpolated into a volume sized $n_x^o \times n_y^o \times n_z^o$ using a linear interpolation defined by (1)-(3):

$$n_x^o = \left\lceil \frac{(n_x - 1) \times d_x}{2.0} \right\rceil + 1 \quad (1)$$

$$n_y^o = \left\lceil \frac{(n_y - 1) \times d_y}{2.0} \right\rceil + 1 \quad (2)$$

$$n_z^o = \left\lceil \frac{(n_z - 1) \times d_z}{2.0} \right\rceil + 1 \quad (3)$$

Generally, the chest CTA images acquired for aorta dissection diagnosis have $n_x = n_y = 512$ and $0.6 < d_x = d_y < 1.0$, which leads to $n_x^o = n_y^o < 256$. The interpolated volume V_o is then put into the center of a uniform volume V_{256} according to the following equation:

$$V_{256}(x, y, z) = \begin{cases} V_o(x', y', z), 0 < x' \le n_x^o, 0 < y' \le n_y^o \\ 0, otherwise \end{cases} \quad (4)$$

where

$$x' = n_x^o/2 + x - 128 \quad (5)$$

$$y' = n_y^o/2 + y - 128 \quad (6)$$

This preliminary step provides coherent volumes sized $256 \times 256 \times n_z^o$ with isotropic pixel resolution of 2.0 mm to to the 3-D CNN.

2.1.2. 3-D CNN

The 3-D CNN is devised based on ResNet proposed in [20] which applies residual convolution blocks to facilitate the training step. As is shown in Fig. 3, the residual block in this 3-D CNN consists of several stacked convolution layers and a skip connection between input and output. The relationship between input x_l and output x_{l+1} is expressed by (7):

$$x_{l+1} = Activate(x_l + F(x_l, \{W_l\})) \quad (7)$$

where l stands for layer number, x_l is the input of layer $l$and x_{l+1} is the output of layer l as well as the input of layer $l + 1$. W_l contains all parameters to be trained in the residual block, and F is the transformation composed of convolution layers, batch normalization layers and activation layers.

Table 1 3-D INVERTED RESIDUAL NETWORK

Layer	Kernel size	Channel number	Output shape
Strided-conv0	$2 \times 2 \times 1$	4	128×128

	Strided-conv1	$2 \times 2 \times 1$	32	64×64
Res_block0	Conv0	$1 \times 1 \times 3$	64	64×64
	Conv1	$3 \times 3 \times 1$	64	64×64
	Conv2	$1 \times 1 \times 3$	32	64×64
	Strided-conv2	$2 \times 2 \times 1$	64	32×32
Res_block1	Conv0	$1 \times 1 \times 3$	128	32×32
	Conv1	$3 \times 3 \times 1$	128	32×32
	Conv2	$1 \times 1 \times 3$	64	32×32
	Strided-conv3	$2 \times 2 \times 1$	96	16×16
Res_block2	Conv0	$1 \times 1 \times 3$	192	16×16
	Conv1	$3 \times 3 \times 1$	192	16×16
	Conv2	$1 \times 1 \times 3$	96	16×16
	Strided-conv4	$2 \times 2 \times 1$	128	8×8
Res_block3	Conv0	$1 \times 1 \times 3$	256	8×8
	Conv1	$3 \times 3 \times 1$	256	8×8
	Conv2	$1 \times 1 \times 3$	128	8×8
	Strided-conv5	$2 \times 2 \times 1$	160	4×4
Res_block4	Conv0	$1 \times 1 \times 3$	320	4×4
	Conv1	$3 \times 3 \times 1$	320	4×4
	Conv2	$1 \times 1 \times 3$	160	4×4
	Strided-conv6	$4 \times 4 \times 1$	192	1×1
	Last_conv	$1 \times 1 \times 1$	1	1×1

Each residual block is composed of three convolution layers followed by batch normalization. The first and third layers are convolutions along the z direction and the second over the x and y directions. Inverted residuals are applied in our implementation in which shortcut connections are added between bottlenecks to allow efficient memory use [39]. We use the Rectified Linear Unit (ReLU) [21] as our activation function in the network.

The proposed 3-D CNN has 19 convolution layers including 13 normal convolution layers and 6 strided convolution layers. Each convolution layer is followed by a batch normalization layer and an activation layer. Strided convolutions are used in our network with stride=2 or stride=4 in x and y directions. The 3-D network starts with two consecutive strided convolution layers in order to reduce the spatial dimension ahead. Five residual blocks are in the network and each block is followed by a strided convolution layer. From the network information in Table 1, we can see that the data size is reduced from 256×256 to 1×1as the input data go through the strided convolution layers. A sigmoid layer is then applied to get the final output of our 3-D network. The full architecture of this 3-D CNN model is detailed in Table 1.

The portion labelling process can be regarded as a binary classification problem. Thus, we choose cross entropy loss as the objective function of our 3-D model to accelerate the training. The cross-entropy loss function is given by (8):

$$loss_{3D} = -mean(\vec{y} \cdot log(\vec{\hat{y}}) + (1 - \vec{y}) \cdot log(1 - \vec{\hat{y}})) \quad (8)$$

where $\vec{y}$ is the label given and $\vec{\hat{y}}$ is the output of our 3-D network for aorta partition.

2.1.3. Slice assignment

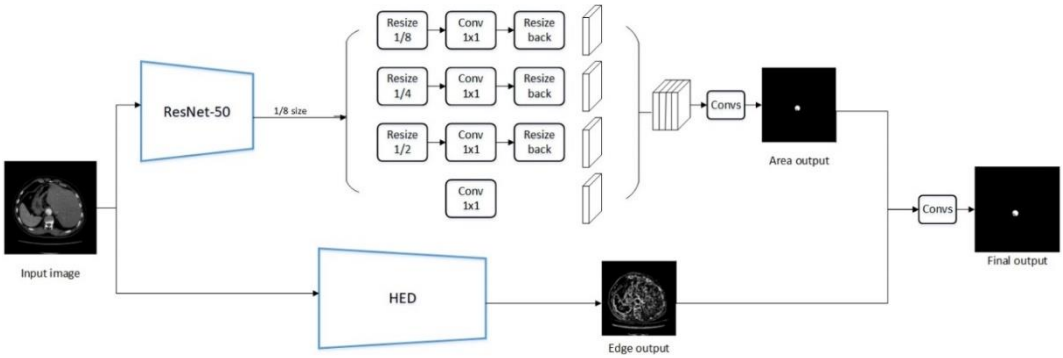

Figure 5 The 2-D network model for aorta segmentation in the proposed method. The model has a two-branch architecture. The upper branch is a PSPnet based on ResNet-50 to acquire a rough segmentation result. The outputs of ResNet-50 are feature maps with 1/8 side length of the input image. The feature maps are then resized into 1/8×, 1/4×, 1/2× and 1× their side lengths for multi-scale segmentation. The branch at bottom is an edge extraction network. After these steps, 6 convolution layers are applied to fuse the two outputs to give the final segmentation result.

The 3-D network outputs a floating number array A_{3D} denoting classification confidence with values between 0 and 1. A_{3D} is a nz_o-sized binary array indicating the slice partition information. In A_{3D}, values lower than 0.5 are assigned to 0 while the other values are assigned to 1. Note slice assignment error might be induced due to the unavoidable errors made by the 3-D CNN. In the aorta volume, we a priori know that the volumes start with slices in portion A and end with slices in portion B and this information is used at first. In addition, considering the slice continuity in each portion, one slice (termed division slice) dividing all the slices into portion A and portion B should be explicitly identified. Some errors in this binary array are corrected using 1-D morphological closing operator with element size set to 5 pixels. A simple search starting from the last slice is applied to find the first slice labeled 0, which is assumed to be the slice dividing the aorta parts belonging to portion A and portion B. The slices below (including) the division slice are labeled as portion A, and the others are labeled as portion B.

A linear interpolation is then applied to restitute the original length nz and another thresholding operation is introduced with threshold=0.5 to get the final division. The overall procedure outlined in this section is depicted in Fig. 5.

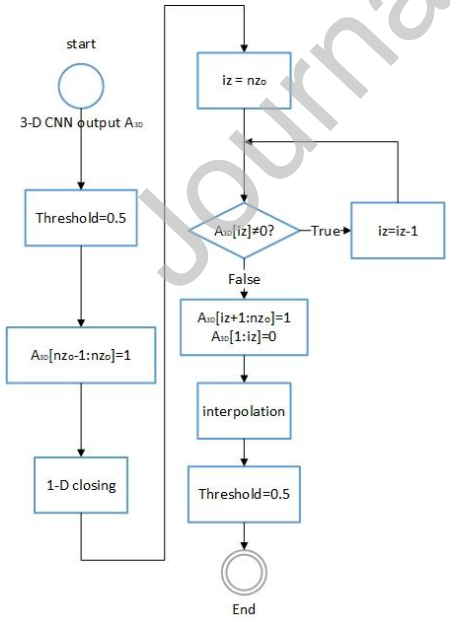

Figure 4 The flow chart of 2-D slice assignment

2.2. 2-D model for aorta segmentation

In order to segment the aorta structure, two identical 2-D CNN models are trained independently using the slices in the two obtained portions. Here the two 2-D CNN models are built based on the PSPnet proposed in [37] which has been shown to achieve state-of-art performance in natural image segmentation. PSPnet applies a spatial pyramid pooling (SPP) module suitable to handle the diameter variations of the aorta over the slices. To further enhance the segmentation accuracy, a boundary extraction branch is added into the model, as depicted in Fig. 4. This multi-branch architecture is inspired by the model proposed in [40]. To obtain a higher instance-segmentation accuracy, the authors split their model into three branches addressing respectively the three tasks of detection, segmentation and boundary extraction after convolution operations. Here, our 2-D PSPnet model only requires two branches for segmentation and boundary extraction whose outputs are later fused. Our whole model is detailed below.

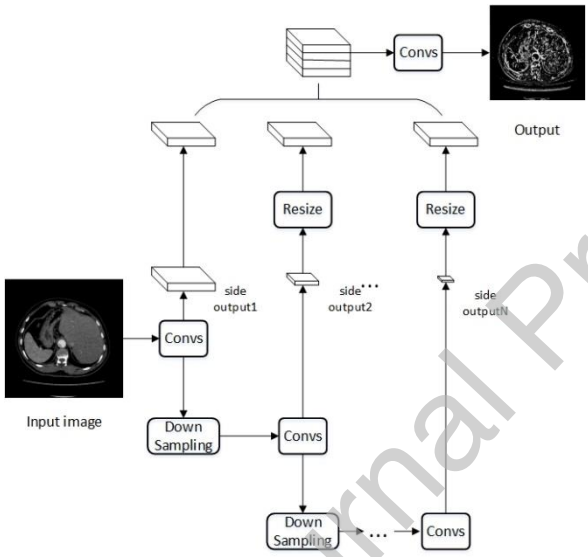

Figure 6 The architecture of HED network.

2.2.1. PSPnet

PSPnet is a CNN architecture aiming at segmenting multi-scale objects. A ResNet is used in PSPnet as a feature extractor followed by a SPP module aimed at segmenting the feature map at different scales. Unlike the method in [37], our solution applies three scaled branches at 1/2, 1/4 and 1/8 of the input resolution, respectively. After the residual blocks and SPP module, images are shrunk into 1/8 of its original resolution because of the strided convolutions in ResNet. Three deconvolution layers are applied to recover the original resolution. A sigmoid cross-entropy loss function $loss_{area}$ is used here as the objective function of this branch:

$$\begin{aligned} loss_{area} &= -mean[W \cdot (L \cdot \log(sigmoid(Y_{area})) \\ &\quad + (1-L) \times \log(1 - sigmoid(Y_{area}))] \\ &= -mean\{W \cdot [L \cdot Y_{area} - \log(e^{Y_{area}} + 1)]\} \end{aligned} \tag{9}$$

where · stands for the Hadamard product operator, L is the ground truth and Y_{area} is the output of the last deconvolution layer. The sigmoid function $sigmoid(x)$ is defined as

$$sigmoid(x) = \frac{1}{1+e^{-x}} \tag{10}$$

and W is a weighting matrix that can be calculated as

$$W = w_t \times L + w_f \times (1 - L) \tag{11}$$

where w_t and w_f correspond to the weighting parameters for vascular and non-vascular areas, respectively.

2.2.2. Edge extraction branch

We apply an edge extraction branch based on the Holistically-nested Edge Detection (HED) network proposed in [41] to extract the aorta boundaries as well as the intimal flaps. HED is a network for edge detection under the deep supervision proposed in [42] to "guide" segmentation results on early layers. The main branch of HED is a standard CNN with convolution layers and strided-conv layers. A side output layer is applied to acquire the edge features at each resolution after the convolution block. Each side output layer l_k corresponds to a sigmoid cross-entropy loss function $loss_{edge}^{k}$. The sketch map of the HED network is reported in Fig. 6 with implementation details in Table 2. Each convolution layer is followed with a batch normalization layer and an activation layer. It should be noted that not all activation layers and batch normalization layers are listed in Table 2. Instead of using multi-task learning, we implemented the edge extraction branch as an individual network. This is because the edge extracted this branch is used to correct the segmentation result from the PSPnet and any influence from the segmentation branch will destroy the meaning of the edge extraction branch.

All side outputs are resized to the original resolution, concatenated together and then fed into convolution layers to get the output E_{out}. A sigmoid cross-entropy loss function $loss_{out}$ is used as the objective function on E_{out}.The loss function for the edge extraction branch $loss_{edge}$ is expressed as

$$loss_{edge} = loss_{out} + \sum_{k=1}^{N} \omega_k loss_{edge}^{k} \tag{12}$$

Here, ω_k is the weighting parameter corresponding to the loss function $loss_{edge}^{k}$ of side output layer l_k. N is the number of side output layers.

The edge label at the original resolution is calculated using the aorta label. The operation is defined by

$$label_{edge} = L \oplus SE - L \tag{13}$$

where L is the aorta label in a given slice, $\oplus$is the morphological dilation operator and SE is a 3×3 structural element. The edge labels for lower resolutions are built by down-sampling $label_{edge}$ using max-pooling operations in each 2×2 neighborhood.

Table 2 2-D EDGE EXTRACTION NETWORK

Layer	Kernel size	Channel number	Output shape
Conv_block0	$\{3 \times 3\} \times 3$	64	512×512
Side_output0	1×1	1	512×512

Max_pool0	2×2	64	256×256
Conv_block1	$\{3 \times 3\} \times 2$	128	256×256
Side_output1	1×1	1	256×256
Max_pool1	2×2	128	128×128
Conv_block2	$\{3 \times 3\} \times 3$	128	128×128
Side_output2	1×1	1	128×128
Max_pool2	2×2	128	64×64
Conv_block3	$\{3 \times 3\} \times 2$	256	64×64
Side_output3	1×1	1	64×64
Max_pool3	2×2	256	32×32
Conv_block4	$\{3 \times 3\} \times 3$	256	32×32
Side_output4	1×1	1	32×32
Concatenate	-	5	512×512
Last_conv	1×1	1	512×512

The loss function $loss_{edge}$ is calculated around the tagged aorta area to learn specified knowledge. The aorta boundaries and the intimal flaps should be predicted as 1 while the aorta voxels should be predicted as 0. The prediction in other regions are inessential in this branch. Therefore, the boundary area $\Omega$is defined using the following expression:

$$\Omega = \{(x, y) | L(x, y) = 1 \vee label_{edge}(x, y) = 1\} \quad (14)$$

where $\vee$stands for logical OR operation.

2.2.3. Label fusion

After the above steps, a label fusion network is used to output the final segmentation by combing the outputs of the PSPnet and the edge extraction branch (as illustrated in Fig. 7). Another cross-entropy loss function $loss_{fusion}$ is used as the objective function of our 2-D model. It is defined deep supervision and end-to-end learning, as the weighted sum of the above three loss functions

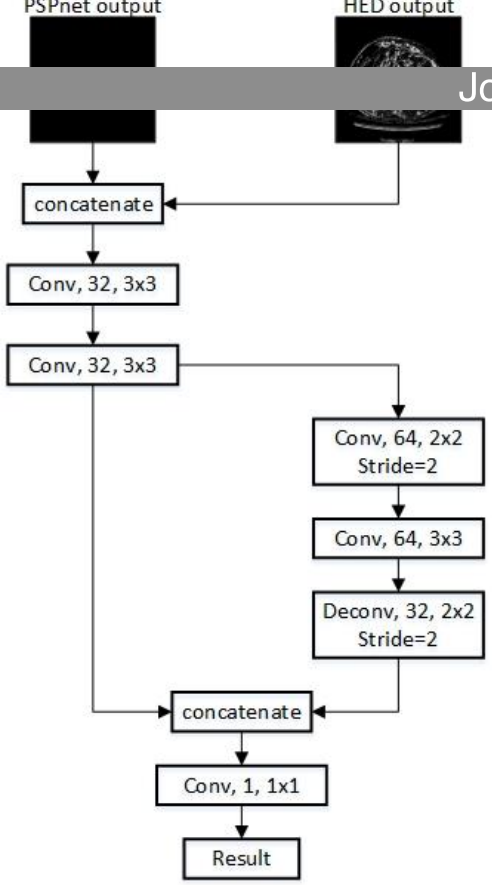

Figure 7 The architecture of the fusion network.

$$loss_{2D} = \mu_1 \cdot loss_{area} + \mu_2 \cdot loss_{edge} + \mu_3 \cdot loss_{fusion} \quad (15)$$

where μ_1, μ_2 and μ_3 are the inter-weights between the three loss functions.

2.3. Implementation Details

In this section, we briefly describe the implementation of the proposed algorithm and the training process. The computer platform was configured as follows: CPU was Intel(R) Core(TM) i7-5930K 3.50GHz; GPU was NVIDIA TITAN X with 12G memory. All codes were written under Python 2.7 and we used Tensorflow r1.4 as the deep learning library [43]. The CUDA edition used here was 8.0. Note that the full source code and the trained models of our approach will be made publicly available.

Unlike natural color images, CTA images only have one channel for CT values. Fine-tuning models trained on natural image datasets like ImageNet [44] may not be a good choice. Therefore, the all convolution kernels in our networks were initialized randomly under Gaussian distribution and all bias parameters were set to zero initially.

During the training phase, the loss function was minimized using the adaptive moment estimation (ADAM) algorithm in [45]. Parameters of each network were optimized end-to-end using the back-propagation calculation. All models were trained using 15,000 iterations, and the learning rate was set to 10^{-3} for the first 10k iterations and 10^{-4} for the following 5k iterations. The decay rates β_1 and β_2 in ADAM were set to 0.9 and 0.999, respectively, according to the default setting in Tensorflow framework. For the 3-D model used in the first stage, due to the varying size of the 3-D CTA data, the batch size was set to 1 for the 3-D CNN model (1 volume data used for each training iteration). To fully exploit the GPU memory, the batch size was set to 4 for the 2-D CNN model (4 slices used for each training iteration). Parameter w_t and w_f in the weighting matrix were set to 3 and 1 by considering that the vascular pixels only occupy a small portion of the images.

Parameter μ_1, μ_2, μ_3 and allω_k appearing in the calculation of $loss_{2D}$ and $loss_{edge}$were set to 1 as it was done in [40] and [41].

It should be noted that no data augmentation was applied as the aorta remain in similar positions for all the CTA images. In fact, we tested some augmentation operations like image normalization without observing any obvious improvement in segmentation accuracy.

2.4. Dataset

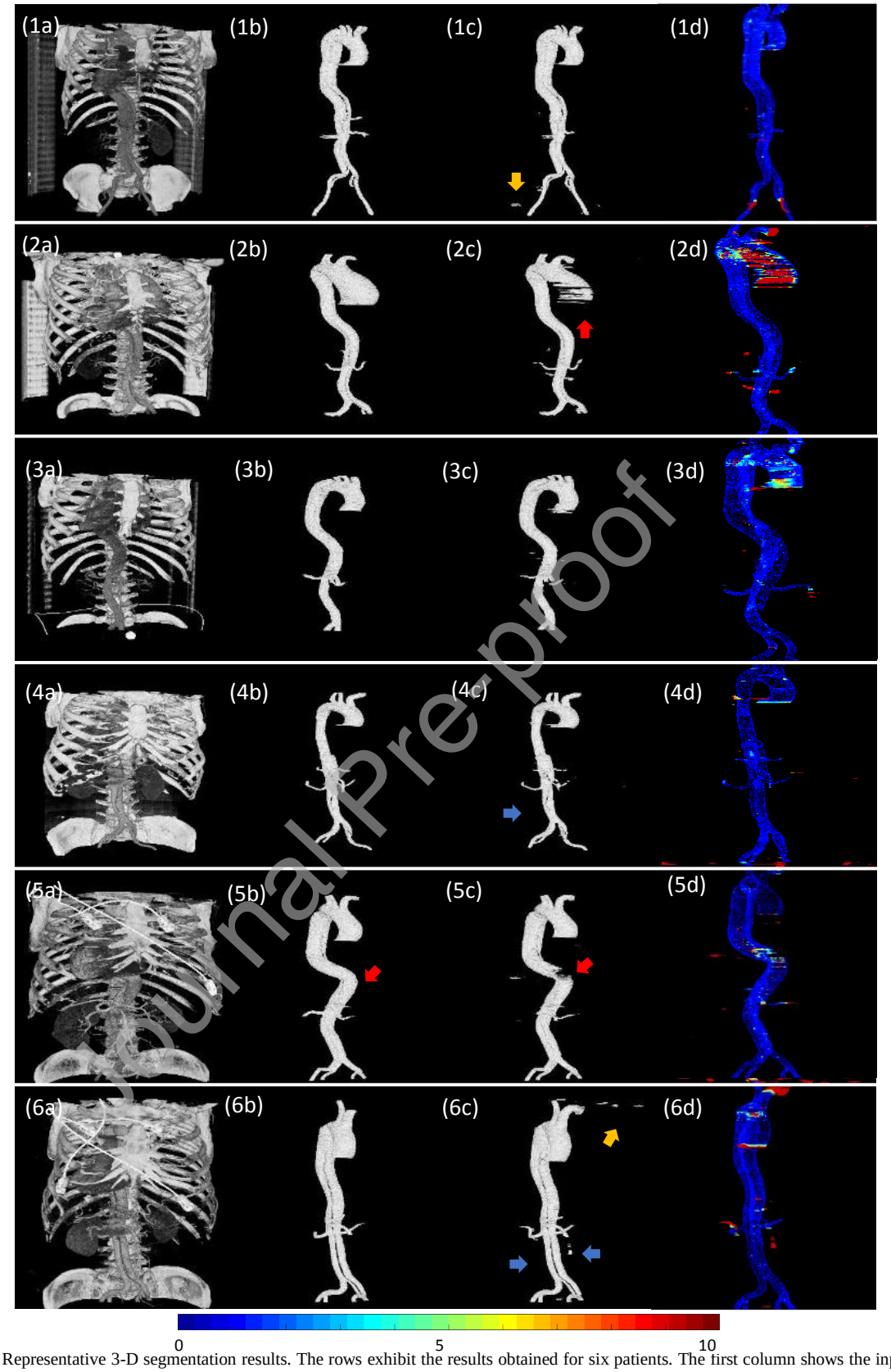

Figure 8 Representative 3-D segmentation results. The rows exhibit the results obtained for six patients. The first column shows the input volumes. The second column represents the manually delineated results. The third row shows the results of the proposed algorithm. The fourth column provides Maximum Intensity Projection (MIP) image of Hausdorff distance between results' boundaries and manually delineated boundaries. The color bar is shown at the bottom. The proposed algorithm works well in most areas. Blue arrows indicate some vessel branches well segmented by our algorithm. Yellow arrows mark some falsely segmented areas while red arrows mark one missing area.

The CTA volumes for training and evaluation were collected from 42 AD patients including 37 males and 5 females in

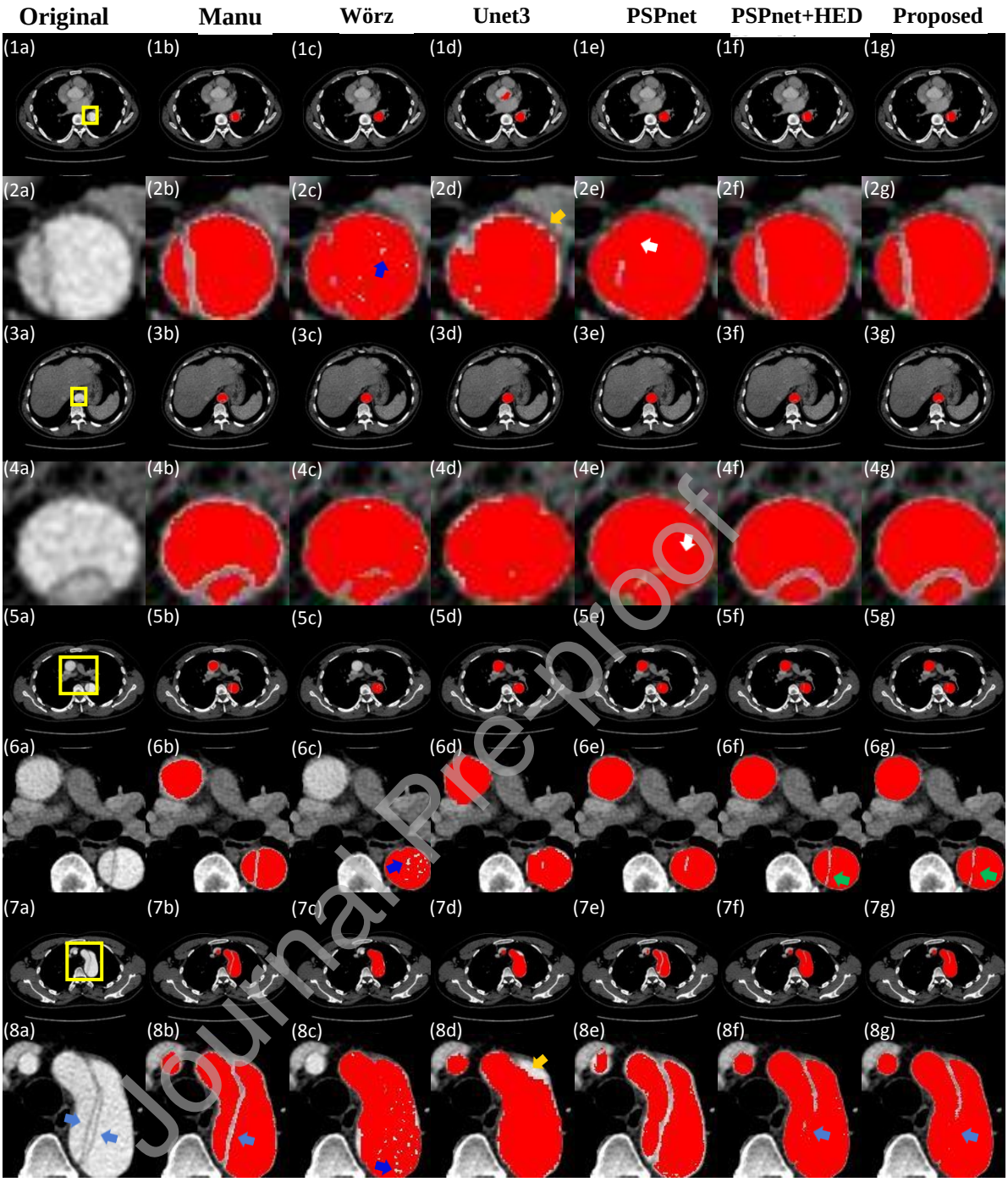

Figure 9 Typical illustrations of segmentation results on several slices from one data volume. The first, third, fifth and seventh rows provide results on the whole 512×512 images. The second, fourth, sixth and eighth rows focus on the aorta indicated by yellow rectangles. The first column shows the original CTA images. The second column displays the ground truth. The third column to the seventh columns correspond respectively to the segmentation results of Wörz method., Unet3d, PSPnet, PSPnet+HED and the proposed algorithm. Dark blue arrows mark some holes appeared in the results of Wörz method. Yellow arrows mark two obvious zig-zagging errors in the results of the Unet3d method. White arrows indicate the areas where PSPnet being unable to divide the true and false lumen. Blue and green arrows give examples with large errors even with the edge extraction branch.

the PLA General Hospital. These patients were aged between 30 to 68. Each volume has from 300 to 1200 slices in Z-axis with 512×512 pixels in each slice. The pixel resolution varies from 0.6825 to 0.9850 millimeter as different CT machine was used for scanning and the slice thickness is between 0.625 and 1.5 millimeter. Aorta contours were extracted manually by cardiovascular experts from Beijing Institute of Technology and Southeast University. Labels indicating portion A and B

in aorta were manually specified under the guidance of an experienced radiologist. In total, we got 42 3-D volumes with 23946 slices. Since we do not have much data for training and validating, the dataset was divided into 6 groups for six-fold cross validation, each group including 7 volumes. Five groups were used for training and the remaining group was used for testing.

2.5 Performance Evaluation

Four metrics Precision, Recall, Intersection of Union (IoU) and Dice were calculated to evaluate the algorithm performance. Precision and Recall are calculated according to the following equations:

$$Precision = \frac{1}{M}\sum_{i=1}^{M}\frac{TP_i}{TP_i+FP_i} \tag{16}$$

$$Recall = \frac{1}{M}\sum_{i=1}^{M}\frac{TP_i}{TP_i+FN_i} \tag{17}$$

Here, M is the number of data volumes. True Positive (TP_i) specifies the number of pixels labeled positive in both manual dilineation and segmentation result in the i -th volume. False Positive (FP_i) is the number of pixels outside the groundtruth contour but positive in the segmentation result in the i -th volume. Conversely, False Negative (FN_i) counts the negative pixels in the i -th volume inside the groundtruth contour. The other two metrics IoU and Dice are defined using the set theory :

$$IoU = \frac{1}{M}\sum_{i=1}^{M}\frac{|G_i\cap S_i|}{|G_i\cup S_i|} \tag{18}$$

$$Dice = \frac{1}{M}\sum_{i=1}^{M}\frac{2(|G_i\cap S_i|)}{|G_i|+|S_i|} \tag{19}$$

Here,G_i represents the manually segmented aorta which we regard as ground truth and S_i is the segmented result in the i -th volume. Operator ∩and ∪stands for intersection operation and union operation. These two metrics can also be calculated using TP_i, FP_i and FN_i as follows：

$$IoU = \frac{1}{M}\sum_{i=1}^{M}\frac{TP_i}{TP_i+FP_i+FN_i} \tag{20}$$

$$Dice = \frac{1}{M}\sum_{i=1}^{M}\frac{2TP}{2TP+FP+FN} \tag{21}$$

The minimal dice index (minDice) among all testing data in corss-validation is also compared to show the performance on the worse case.

In addition to an evaluation on the whole image, the performance was also estimated by restricting the computation of the four metrics to the boundary area Ω_{edge} defined as

$$\Omega_{edge} = G \oplus SE - G \circ SE \tag{22}$$

where G is the manual delineation as the ground truth, SE is the $3 \times 3 \times 1$ structural element with all ones for the morphological operations used: dilation represented by $\oplus$and erosion by $\circ$.

3. Results

Table 3 lists the results obtained when using different algorithms on our dataset. All models here were trained under the same protocol and the training iteration numbers and parameters in ADAM algorithm were kept the same.

Table 3 COMPARISON WITH OTHER METHODS

Method	Precision	Recall	IoU	Dice	minDice
Unet3d	88.55	86.33	77.50	87.26	**81.90**
ERFNet	90.14	87.60	80.34	88.40	74.15
PSPnet	92.81	87.03	81.15	89.34	65.03
PSPnet+HED	**93.66**	90.82	85.22	91.81	69.14
Proposed	93.54	**91.44**	**85.66**	**92.13**	74.50

All models are evaluated using the whole volumes. Unet3d in the table refers to a 3-D encoder-decoder architecture based on the idea proposed in [32], which adds skip connections between the encoder path and the decoder path. The 2-D 3×3 convolutions used in the original U-net model in [32] were replaced by 3-D $3 \times 3 \times 3$ convolutions in the Unet3d method. The 2-D pooling layers and up-sampling layers were substituted by 3-D strided convolution and deconvolution layers. Considering the limited GPU memory, the down-sampling operation mentioned in section 2.1 was introduced to reduce the original volume to size $256 \times 256 \times nz_o$. An up-sampling operation was applied on the output of the network to restore the original volume dimensions of the final segmentations. The metrics, including Precision, Recall, IoU and Dice, were calculated between the up-sampled results and labels in their original resolution. The Unet 3d model is regarded as the baseline model. To further illustrate the advancement of the proposed method, the method is also compared with ERFNet [46], which is an 2-D segmentation specific CNN.

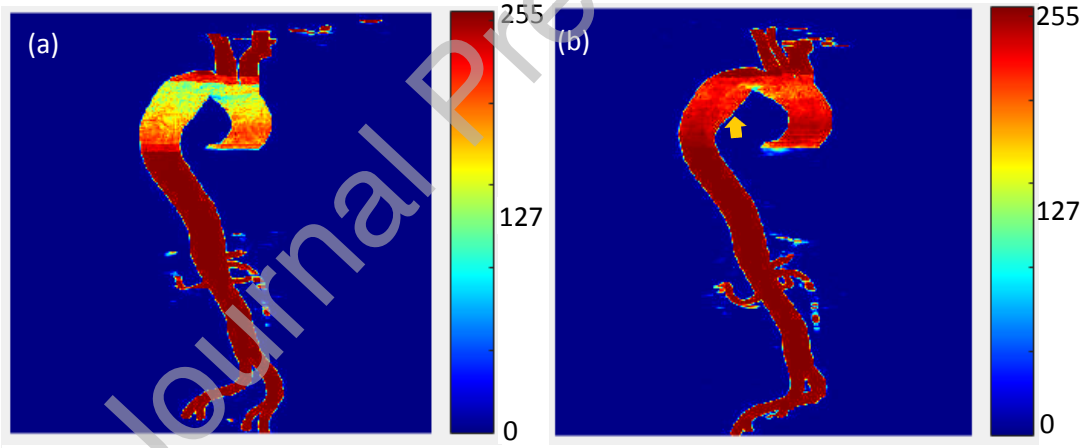

Figure 10 Network outputs for one data volume before applying the final 0.5 threshold. The values are multiplied with 255 and exhibited using pseudo-color. The left image shows the output of PSPnet+HED while the right image shows the output of the proposed algorithm

In Table 3, PSPnet refers to a single 2-D network model without the edge extraction branch, and PSPnet+HED stands for the 2-D network for aorta without the 3-D model for aorta portion partition. So, these two networks were both trained using the whole data volume without portion partition. It can be seen in Table 3 that the proposed algorithm has an improved performance when compared to the other models. The proposed model leads to the best results in Recall, IoU and Dice metrics with values equal to 91.44%, 85.66% and 92.13%, respectively. The Precision index for the proposed algorithm is only slightly lower than the best result produced by PSPnet+HED, the difference being 0.12%. For minDice, the performance of our model takes the second place behind Unet3d. ERFNet performs slightly worse than PSPnet on our dataset, but its

performance is more stable as it achieves higher minDice value. The proposed method performs better on every metric compared to ERFNet.

Table 4 lists the quantitative results with calculation limited to the boundary area Ω_{edge}. The models with an edge extraction branch (PSPnet+HED and the proposed method) perform better than those without (Unet3d, ERFNet and PSPnet). The Unet3d method leads to the smallest values for all the four metrics. The results from the methods of PSPnet and Wörz method are slightly better than the Unet3d method. The PSPnet+HED and the proposed method show quite similar performance.

Fig. 8 illustrates the 3-D views of the results for six representative data volumes. In most cases, visual comparisons with the manually delineations show that the proposed algorithm performs well in capturing aorta features. Besides, the connections between vessel branches and aorta are also clearly extracted using the proposed method (see the blue arrows in Fig. 8(4c) and Fig. 8(6c)). Several non-aorta voxels (i.e. yellow arrows in Fig. 8(1c) and Fig. 8(6c)) can be easily removed by keeping the largest connected region. However, in fig. 8(2c), some voxels of the ascending aorta are missing (see the red arrow). This phenomenon results from the abnormally large radius of the ascending aorta. A similar situation (indicated by a red arrow) can be observed in Fig. 8(5c).

Table 4 METRICS ON BOUNDARY

Method	Precision	Recall	IoU	Dice
Unet3d	65.20	62.73	46.56	63.45
ERFNet	78.85	69.02	57.52	73.83
PSPnet	80.74	68.66	58.10	73.41
PSPnet+HED	**81.62**	83.11	68.89	**81.39**
Proposed	81.32	**83.45**	**68.92**	**81.39**

Fig. 9 provides 2-D axial views corresponding to the volume depicted in Fig. 8(3a). The visualized results are consistent with the quantitative values reported in Table 3 and Table 4.

The computational cost for the proposed algorithm is analysed in Table 5. The data volume in Fig. 8(1a) with the size $512 \times 512 \times 433$ was used in computation cost evaluation. It is found in Table 5 that the Wörz method has the highest computation cost, and the proposed algorithm requires more computation time than other neural network-based methods including Unet3d, PSPnet and PSPnet+HED. This is because three different models are included in the proposed algorithm.

Table 5 COMPUTATIONAL TIME ON ONE SELECTED VOLUME IN SIZE 512x512x433

Method	Time(s)
Unet3d	17.10
ERFNet	20.80
PSPnet	28.96
PSPnet+HED	37.98
Proposed	48.74

4. Discussion

In this paper, we proposed an algorithm for dissected aorta extraction/segmentation on CTA images. The proposed algorithm firstly divided the 3-D volume into two different portions with distinct topologies using a 3-D CNN. Two 2-D CNNs based on PSPnet were then applied to fulfill the segmentation task upon the slices within the two portions. An edge branch based on HED network was incorporated to improve the extraction accuracy of intimal flaps and boundaries.

The performance of the proposed method was compared with several different deep-learning-based methods include Unet3d, PSPnet, PSPnet+HED and ERFnet. Among all the methods, Unet3d performs the worst, and the extracted aorta boundaries present Zig-zagging errors (yellow arrows in Fig. 9(2d) and Fig. 9(8d)), which might be the consequence of down-sampling and up-sampling operation in this method. However, even with a GPU with enough physical memory, the interpolation is still needed due to the anisotropy of CT images. The performance of PSPnet is better than that of Unet3d method. With the proposed method, the outer boundaries of aorta are better extracted but much less at intimal flaps than other methods (white arrows in Fig. 9(2e) and Fig. 9(4e)). This is attributed to the use of the edge extraction branch. The models with edge extraction branch (PSPnet+HED and proposed method) perform much better in lumen extraction than the other methods (Unet3d and PSPnet) in most cases (see the illustrations given in Fig. 9(2f), Fig. 9(2g), Fig. 9(4f) and Fig. 9(4g)). Although better than Unet3d and PSPnet, the proposed method leads to reduced adhesion errors (pointed out by the green arrows in Fig. 9(6f) and Fig. 9(6g)). Specifically, we can note that the performance is relatively poor around the area with multiple intimal flaps (the blue arrows in Fig. 9(8a)), which were caused by patient breathing or cardiac motions. But in this work only one intimal flap is routinely delineated (see the blue arrow in Fig. 9(8b)) in manual labelling even for these multiple intimal flaps. Such contradiction might lead to lowered performance of the algorithm.

With respect to the metrics minDice in Table 3, PSPnet, PSPnet+HED and the proposed method perform not as well as Unet3d. To highlight this, Fig. 10 depicts the direct output of 2-D network for the data volume displayed in Fig. 8(6a). A scalar value termed vesselness (between 0 and 1) was assigned to each voxel representing the confidence we get for this voxel to be inside the aorta. Voxels with vesselness larger than 0.5 were labeled as aorta points in the final segmentation result. In Fig. 10, the results are presented in pseudo-color. It can be observed in Fig.10 that for the PSPnet+HED method the vesselness values in portion B are lower than those in portion A. However, the vesselness values might also drop below 0.5 and some voxels in aorta region are missing in the segmentation result. Fig. 10(b) depicts the output vesselness values of the proposed method. We find that the vesselness values in portion B are much higher than the vesselness values in the same area in Fig. 10(a) for the PSPnet+HED method, which implies that the portion partition operation of 3-D CNN contributes to the improvement of the final segmentation. Nevertheless, compared to the overall vesselness values in aorta structure, the resulted vesselness values are slightly lower in the arch region with sharp bending (see the region pointed by yellow arrow in Fig. 10(b)). Further improvement can be expected by using finer portion classification in the first stage.

We also compared the the proposed method with a non-learning-based method proposed in [15] by Wörz et al. All parameters are set according to the paper. As the algorithm is unable to segment aorta in all slices, only chest slices (including ascending aorta, aortic arch and a small part of descending aorta) are used to evaluate the performance of this algorithm. Wörz method performs badly compared to other learning-based methods. It only achieves an average Dice value of 81.01 while it takes more than 9 minutes to process an input volume sized 512x512x433. Deep-learning-based methods achieve superior performance with much reduced computation time in testing compared with those conventional methods.

Experimental results in this study have shown that the proposed algorithm has satisfying performance for capturing the aorta structure while avoiding false positives on the intimal flaps. However, some improvements still need to be considered.

The dataset is small for training a 3-D segmentation network and that is the reason why we only use the 3-D network for aorta partition. The inputs to the 3-D network are also down-sampled to weaken the influence of voxel sizes and image details. An additional step called slice assignment is applied after the 3-D network to further reduce the impact of errors. The 2-D network does not seem to easily get overfitted as we have more than 23000 slice to train this network, but we will try adding data augmentation like flipping or affine transforms to build a better training dataset.

The compromise done between computational resources and segmentation efficiency leads us to separate the process into two stages. As seen in Table 5, the proposed method nearly doubled the computation time compared to 2-D only method like PSPnet. The second stage is operations in 2-D slices and thus the continuity along the z direction is not fully used.

The vesselness notion, which is introduced for estimating the confidence of a voxel belonging to the aorta structure, still presents variations in the segmented arch parts. These variations will lead to notable errors on some CTA volumes. The "minDice" values in Table 3 give a rough glancing of this kind of errors. In the results from 2-D method (PSPnet and PSPnet+HED), 3 volume of 42 observed severe aorta structure missing in the segmentation results. Most of which appeared on the aortic arch. The proposed method alleviated this problem to a certain extent by adding a 3-D network. However, aortic structure missing still happens sometime (see Fig.8(5c)). A larger training dataset would be necessary to perfectly solve this kind of problems.

Specially, the lower accuracy due to multiple local intimal flaps need to be overcome by increasing the number of patient cases and avoiding the contradicting manual delineations. Further work will also be devoted to processing clinical datasets with more pathological situations such as aorta aneurysms.

Acknowledgements

The authors would like to thank Radiology Department of the PLA General Hospital for providing clinical image data.

Funding: This study was funded in part by the State's Key Project of Research and Development Plan [Grant 2017YFC0109202, 2017YFA0104302 and 2017YFC0107900], in part by the National Natural Science Foundation [Grant 81530060 and 81471752].

Conflict of Interest: The authors declare that they have no conflict of interest.

Ethical approval: All procedures performed in studies involving human participants were in accordance with the ethical standards of the institutional and/or national research committee and with the 1964 Helsinki declaration and its later amendments or comparable ethical standards. Informed consent was obtained from all individual participants included in the study.

References

[1] Criado FJ. Aortic dissection: a 250-year perspective. *Texas Heart Institute Journal.* 2011;**38(6)**:694.

[2] White A, Broder J, Mando-Vandrick J, Wendell J, Crowe J. Acute Aortic Emergencies—Part 2 Aortic Dissections. *Advanced emergency nursing journal.* 2013 Jan 1;**35(1)**:28-52.

[3] Daily PO. Management of acute aortic dissections. Ann Thorac Surg. 1970; 10:237-47.

[4] Coady MA, Rizzo JA, Goldstein LJ, Elefteriades JA. Natural history, pathogenesis, and etiology of thoracic aortic aneurysms and dissections. *Cardiology clinics.* 1999 Nov 1;**17(4)**:615-35.

[5] Hagan PG, Nienaber CA, Isselbacher EM, Bruckman D, Karavite DJ, Russman PL, Evangelista A, Fattori R, Suzuki T, Oh JK, Moore AG. The International Registry of Acute Aortic Dissection (IRAD): new insights into an old disease. *Jama.* 2000 Feb 16;**283(7)**:897-903.

[6] Xu H, Li Z, Dong H, Zhang Y, Wei J, Watton PN, Guo W, Chen D, Xiong J. Hemodynamic parameters that may predict false-lumen growth in type-B aortic dissection after endovascular repair: A preliminary study on long-term multiple follow-ups. *Medical Engineering & Physics*. 2017 Dec 1;**50**:12-21.

[7] Chen D, Müller-Eschner M, von Tengg-Kobligk H, Barber D, Böckler D, Hose R, Ventikos Y. A patient-specific study of type-B aortic dissection: evaluation of true-false lumen blood exchange. *Biomedical engineering online*. 2013 Dec 1;**12(1)**:65.

[8] Cheng SW, Lam ES, Fung GS, Ho P, Ting AC, Chow KW. A computational fluid dynamic study of stent graft remodeling after endovascular repair of thoracic aortic dissections. *Journal of vascular surgery.* 2008 Aug 1;**48(2)**:303-10.

[9] Chen D, Müller-Eschner M, Kotelis D, Böckler D, Ventikos Y, von Tengg-Kobligk H. A longitudinal study of Type-B aortic dissection and endovascular repair scenarios: computational analyses. *Medical engineering & physics.* 2013 Sep 1;**35(9)**:1321-30.

[10] Frangi AF, Niessen WJ, Vincken KL, Viergever MA. Multiscale vessel enhancement filtering. *In nternational conference on medical image computing and computer-assisted intervention* 1998 Oct 11 (pp. 130-137).

[11] Manniesing R, Viergever MA, Niessen WJ. Vessel enhancing diffusion: A scale space representation of vessel structures. *Medical image analysis.* 2006 Dec 1;**10(6)**:815-25.

[12] Deschamps T, Cohen LD. Fast extraction of minimal paths in 3D images and applications to virtual endoscopy. Medical image analysis. 2001 Dec 1;5(4):281-99.

[13] Chen Y, Zhang Y, Yang J, Cao Q, Yang G, Chen J, Shu H, Luo L, Coatrieux JL, Feng Q. Curve-like structure extraction using minimal path propagation with backtracking. *IEEE Transactions on image processing.* 2015 Nov 2;**25(2)**:988-1003.

[14] Kayikcioglu T, Mitra S. A new method for estimating dimensions and 3-d reconstruction of coronary arterial trees from biplane angiograms. *In [1993] Computer-Based Medical Systems-Proceedings of the Sixth Annual IEEE Symposium* 1993 Jun 13 (pp. 153-158).

[15] Worz S, Rohr K. Segmentation and quantification of human vessels using a 3-D cylindrical intensity model. *IEEE transactions on Image Processing.* 2007 Jul 16;**16(8)**:1994-2004.

[16] Krissian K, Carreira JM, Esclarin J, Maynar M. Semi-automatic segmentation and detection of aorta dissection wall in MDCT angiography. *Medical image analysis.* 2014 Jan 1;**18(1)**:83-102.

[17] Kovács T, Cattin P, Alkadhi H, Wildermuth S, Székely G. Automatic segmentation of the vessel lumen from 3D CTA images of aortic dissection. *In ildverarbeitung für die Medizin 2006* 2006 (pp. 161-165).

[18] Lee N, Tek H, Laine AF. True-false lumen segmentation of aortic dissection using multi-scale wavelet analysis and generative-discriminative model matching. InMedical Imaging 2008: *Computer-Aided Diagnosis* 2008 Mar 17，**6915**, p. 69152V).

[19] Fetnaci N, Łubniewski P, Miguel B, Lohou C. 3D segmentation of the true and false lumens on CT aortic dissection images. *In hree-Dimensional Image Processing (3DIP) and Applications 2013* 2013 Mar 12，**8650**： 86500M).

[20] Litjens G, Kooi T, Bejnordi BE, Setio AA, Ciompi F, Ghafoorian M, Van Der Laak JA, Van Ginneken B, Sánchez CI. A survey on deep learning in medical image analysis. *Medical image analysis.* 2017 Dec 1;**42**:60-88.

[21] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. *Communications of the ACM.* 2017 May 24;**60(6)**:84-90.

[22] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556.* 2014 Sep 4.

[23] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. *InProceedings of the IEEE conference on computer vision and pattern recognition* 2016 (pp. 770-778).

[24] Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. *InProceedings of the IEEE conference on computer vision and pattern recognition* 2017 (pp. 4700-4708).

[25] Yin XR, Zhao QL, Liu J, Chen Y et al. Domain Progressive 3D Residual Convolution Network to Improve Low Dose CT Imaging. *IEEE, Transactions on Medical Imaging*. 2019; **38(12)**, 2903-2913.

[26] Ge RJ, Yang GY, Chen Y, et al. PV-LVNet: Direct Left Ventricle Multitype Indices Estimation from 2D Echocardiograms of Paired Apical Views with Deep Neural Networks, *Medical Image Analysis*. 2019; **58**, 101554..

[27] Ge RJ, Yang GY, Chen Y, et al. K-Net: Integrate Left Ventricle Segmentation and Direct Quantification of Paired Echo Sequence, *IEEE, Transactions on Medical Imaging*. 2020; **39(5)**, 1690-1702.

[28] Lyu TL, Chen Y et al. Estimating dual-energy CT imaging from single-energy CT data with material decomposition convolutional neural network. *Medical Image Analysis*. 2021; **70(5)**:102001.

[29] Shelhamer E, Long J, Darrell T. Fully convolutional networks for semantic segmentation. *IEEE transactions on pattern analysis and machine intelligence.* 2017 Apr;**39(4)**:640-51.

[30] Badrinarayanan V, Kendall A, Cipolla R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence.* 2017 Jan 2;**39(12)**:2481-95.

[31] Sermanet P, Eigen D, Zhang X, Mathieu M, Fergus R, LeCun Y. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229.* 2013 Dec 21.

[32] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. *InInternational Conference on Medical image computing and computer-assisted intervention* 2015 Oct 5 (pp. 234-241).

[33] Chen LC, Papandreou G, Kokkinos I, Murphy K, Yuille AL. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*. 2014 Dec 22.

[34] Chen LC, Papandreou G, Kokkinos I, Murphy K, Yuille AL. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence.* 2017 Apr 27;**40(4)**:834-48.

[35] Chen LC, Papandreou G, Schroff F, Adam H. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*. 2017 Jun 17.

[36] Chen LC, Zhu Y, Papandreou G, Schroff F, Adam H. Encoder-decoder with atrous separable convolution for semantic image segmentation. *InProceedings of the European conference on computer vision (ECCV)* 2018 (pp. 801-818).

[37] Zhao H, Shi J, Qi X, Wang X, Jia J. Pyramid scene parsing network. *InProceedings of the IEEE conference on computer vision and pattern recognition* 2017 (pp. 2881-2890).

[38] Milletari F, Navab N, Ahmadi SA. V-net: Fully convolutional neural networks for volumetric medical image segmentation. *In2016 fourth international conference on 3D vision (3DV)* 2016 Oct 25 (pp. 565-571). IEEE.

[39] Sandler M, Howard A, Zhu M, Zhmoginov A, Chen LC. Mobilenetv2: Inverted residuals and linear bottlenecks. *InProceedings of the IEEE conference on computer vision and pattern recognition* 2018 (pp. 4510-4520).

[40] Xu Y, Li Y, Wang Y, Liu M, Fan Y, Lai M, Eric I, Chang C. Gland instance segmentation using deep multichannel neural networks. *IEEE Transactions on Biomedical Engineering.* 2017 Mar 23;**64(12)**:2901-12.

[41] Xie S, Tu Z. Holistically-nested edge detection. *InProceedings of the IEEE international conference on computer vision 2015* (pp. 1395-1403).

[42] Lee CY, Xie S, Gallagher P, Zhang Z, Tu Z. Deeply-supervised nets. *InArtificial intelligence and statistics* 2015 Feb 21 (pp. 562-570).

[43] Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado GS, Davis A, Dean J, Devin M, Ghemawat S. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467.* 2016 Mar 14.

[44] Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, Berg AC. Imagenet large scale visual recognition challenge. *International journal of computer vision.* 2015 Dec 1;115(3):211-52.

[45] Kingma DP, Ba J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980.* 2014 Dec 22.

[46] Romera E, Alvarez JM, Bergasa LM, Arroyo R. Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*. 2017 Oct 9;**19(1)**:263-72.