Artificial Intelligence In Medicine 121 (2021) 102189

Contents lists available at ScienceDirect

Artificial Intelligence In Medicine

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

Fusing 2D and 3D convolutional neural networks for the segmentation of aorta and coronary arteries from CT images

Linyan Gu [a,b], Xiao-Chuan Cai [c,*]

[a] Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China
[b] Shenzhen Key Laboratory for Exascale Engineering and Scientific Computing, Shenzhen 518000, China
[c] Faculty of Science and Technology, University of Macau, Avenida da Universidade, Taipa, Macao, China

ARTICLE INFO

Keywords:
Convolutional neural networks
Human aorta and coronary arteries segmentation
2D and 3D network fusion
Medical images

ABSTRACT

Automated segmentation of three-dimensional medical images is of great importance for the detection and quantification of certain diseases such as stenosis in the coronary arteries. Many 2D and 3D deep learning models, especially deep convolutional neural networks (CNNs), have achieved state-of-the-art segmentation performance on 3D medical images. Yet, there is a trade-off between the field of view and the utilization of inter-slice information when using pure 2D or 3D CNNs for 3D segmentation, which compromises the segmentation accuracy. In this paper, we propose a two-stage strategy that retains the advantages of both 2D and 3D CNNs and apply the method for the segmentation of the human aorta and coronary arteries, with stenosis, from computed tomography (CT) images. In the first stage, a 2D CNN, which can extract large-field-of-view information, is used to segment the aorta and coronary arteries simultaneously in a slice-by-slice fashion. Then, in the second stage, a 3D CNN is applied to extract the inter-slice information to refine the segmentation of the coronary arteries in certain subregions not resolved well in the first stage. We show that the 3D network of the second stage can improve the continuity between slices and reduce the missed detection rate of the 2D CNN. Compared with directly using a 3D CNN, the two-stage approach can alleviate the class imbalance problem caused by the large non-coronary artery (aorta and background) and the small coronary artery and reduce the training time because the vast majority of negative voxels are excluded in the first stage. To validate the efficacy of our method, extensive experiments are carried out to compare with other approaches based on pure 2D or 3D CNNs and those based on hybrid 2D-3D CNNs.

1. Introduction

Coronary artery disease (CAD) is the most common type of heart disease and it is one of the leading causes of death worldwide [1]. CAD induces plaque build-up in the coronary arteries, which may cause luminal narrowing, also known as stenosis, and can often be life-threatening when total occlusions of the artery occur. CT coronary angiography is the primary imaging modality for diagnosing CAD due to its superior image resolution [2]. To facilitate the diagnosis, accurate segmentation of the aorta and coronary arteries is a critical step for interpreting CT images for the purpose of stenosis detection and quantification, such as stenosis grading via the fractional flow reserve (FFR) [3]. Due to the large number of pixels in CT images, the process of manual or semi-automatic segmentation is time consuming and tedious, with bias being introduced by clinical experts. Therefore, it is highly desirable to develop an automated and robust system that can efficiently extract the aorta and the coronary artery lumen from the CT images. However, automated segmentation is a challenging task due to the inherent image noise, similar objects in the background, the complicated anatomical system involving the aorta (the largest artery in the human body) and the much smaller coronary arteries, and the large inter-subject variations. Various conventional image segmentation algorithms have been proposed previously to achieve 3D blood vessel segmentation, such as region-based methods [4], edge-based methods [5], tracking-based methods [6], learning-based methods [7], and so on. In the previous few years, medical image segmentation based on deep learning techniques has received vast attention [8].

Deep learning algorithms have rapidly become a methodology of choice for analyzing medical images, such as image registration [9], image segmentation [10,11], image retrieval [12,13], and so on. Among

* Corresponding author.
E-mail addresses: ly.gu@siat.ac.cn (L. Gu), xccai@um.edu.mo (X.-C. Cai).

https://doi.org/10.1016/j.artmed.2021.102189
Received 1 February 2021; Received in revised form 23 September 2021; Accepted 29 September 2021
Available online 7 October 2021

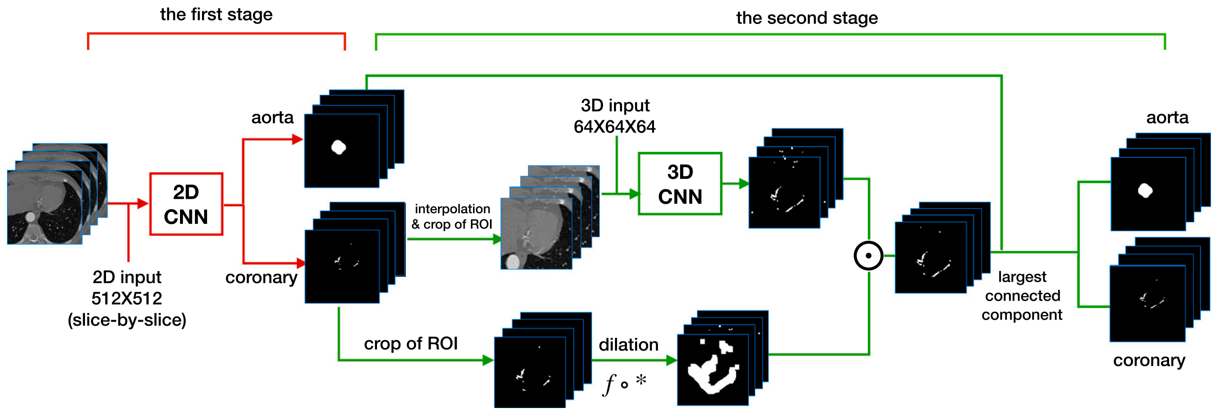


Fig. 1. The general framework of the two-stage method. $\odot$ and * denote the element-wise product and the convolutional operation, respectively, and $f(x)$ equals 1 for $x > 0$ and 0 for $x = 0$. The 2D CNN in the first stage receives the 2D CT slice of size 512×512 from the whole volume, and the input of the 3D CNN in the second stage is the 3D patch of size $64 \times 64 \times 64$ cropped from the candidate regions obtained in the first stage.

these, medical image segmentation based on deep learning techniques, in particular convolutional neural networks (CNNs), has greatly improved segmentation accuracy for 3D medical images.

1.1. CNNs for medical image segmentation

Generally speaking, a CNN for semantic segmentation consists of a downsampling path and an upsampling path, which can take an input of arbitrary size and produce a correspondingly sized output to classify each pixel with efficient inference and learning. The 2D UNet [10] is one of the most well-known CNN architectures for medical image segmentation, and it combines high-resolution features from the downsampling path and contextual features from the upsampling path with the so-called skip connections. Other 2D CNNs (e.g., [14,15,16]) have also been successfully applied to 3D medical images. Many 3D extensions of the UNet architecture have been introduced; for example, [17] proposed the VNet, which performs 3D segmentation using 3D convolutions with an objective function based on the dice coefficient. There are other 3D CNNs that have demonstrated compelling performance for 3D segmentation; for example, Kamnitsas et al. [18] built a 3D CNN with a multi-stream architecture (called DeepMedic) to capture multi-scale features, and VoxResNet [11] takes advantage of residual learning and integrates multi-modality and multi-level contextual information.

1.2. Trade-off between the field of view and the utilization of inter-slice information

Although many 2D and 3D CNNs have greatly improved segmentation accuracy, there is a trade-off between the field of view and the utilization of inter-slice information when using 2D or 3D CNNs for 3D segmentation [19]. On the one hand, a 2D CNN offers a much larger field of view but is not able to explore the inter-slice connection. On the other hand, a 3D CNN attempts to fully utilize the 3D image information but always has a limited field of view due to the significant memory and computational requirements. Particularly, for the task of the aorta and coronary artery segmentation, a limited field of view increases the difficulty in distinguishing the aorta (or coronary arteries) from other tissues and organs with similar characteristics to those of the aorta (or coronary arteries). In addition, for coronary arteries, a lack of inter-slice information often leads to some level of discontinuity and a high missed detection rate, especially for the luminal narrowing of CAD patients for whom an accurate segmentation is critically important.

1.3. Methods to circumvent the trade-off

There are several methods proposed to circumvent this trade-off by carefully designing the network architecture. For example, [20] proposed a multi-view scheme that utilizes a separate CNN for each orthogonal 2D plane, followed by an adaptive fusion strategy to fuse these three segmentation results. However, this multi-view scheme uses only a small fraction of the 3D image information. [21] extended the 2D UNet to a 2D-3D UNet by retaining the large field of view but reducing the number of feature maps due to memory constraints. However, reducing feature maps may compromise segmentation accuracy since recent evidence [22,23] reveals increasing the number of filters can improve the performance of networks. In [24], a CNN was designed to take volumetric image input as multi-channel vector images (known as a 2.5D representation) that pass through the first 2D multichannel convolutional layer, with the subsequent convolutional operations functioning exactly the same as those in 2D methods. [19] proposed an ensemble learning framework in which a CNN was developed to combine the results from the trained 2D and 3D models, in which three 2D models, one 3D model, and one ensemble network need to be learned.

1.4. CNNs for coronary artery segmentation

Some methods based on CNNs have been developed for coronary artery segmentation. For example, [25] proposed a multi-task CNN with triplanar orthogonal input patches to perform multi-organ segmentation, including the coronary arteries. [26] used two CNNs with the DeepMedic architecture to realize 3D coronary artery segmentation and aorta segmentation, and then further refined the result using the largest connected component method. Both [27] and [2] used the 3D UNet architecture, and [2] used a two-channel strategy, in which the input consists of two channels: one from the original CT image and the other from the vesselness map derived by applying vesselness filters to the original CT image. In [3], the spatial prior knowledge constraint was used together with the CNN to reduce vast majority negative voxels. In addition, [28] used various enhancement methods to pre-process original images and then used the 2D UNet to segment the coronary arteries from these enhanced images. In addition to 3D networks, there are other ways to extract inter-slice information, which can improve the continuity between slices. [29] used the traditional level set method to refine and smooth the boundary of the segmentation results obtained by a 3D network. In [30], a paired multi-scale 3D CNN is used to obtain a larger receptive field and extract 3D contexture information. [31,32] used the tree-structural long short-term memory (LSTM) method and centerlines to model the underlying tree structures of coronary arteries. [33] formulated 2D orthogonal cross-hair filters which make use of 3D context information at a reduced computational burden. Besides, [34] proposed a semi 3D architecture that combines the 3D UNet and 2D

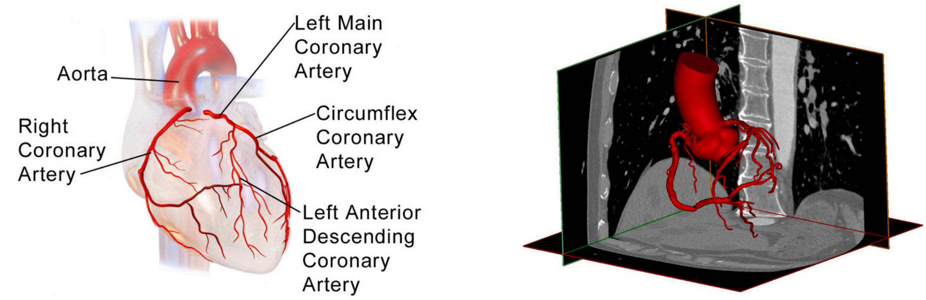
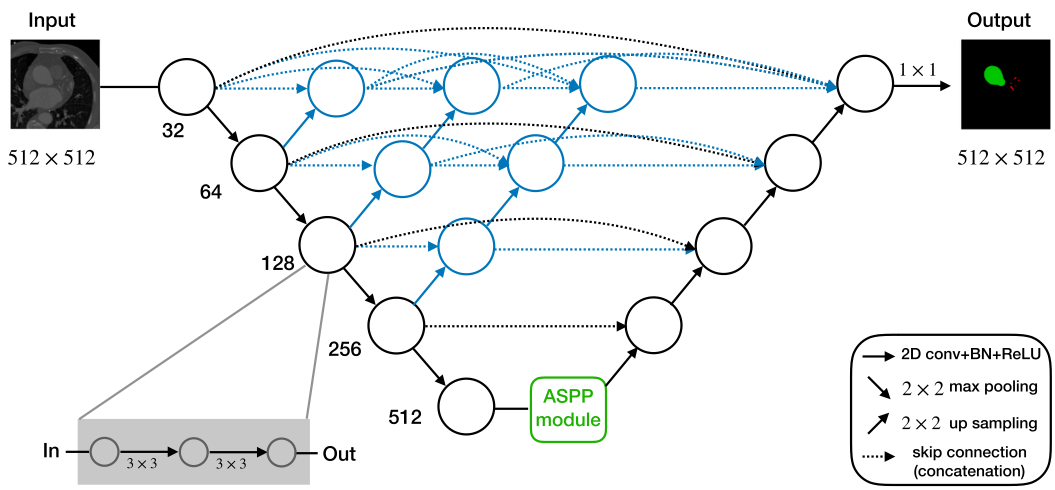
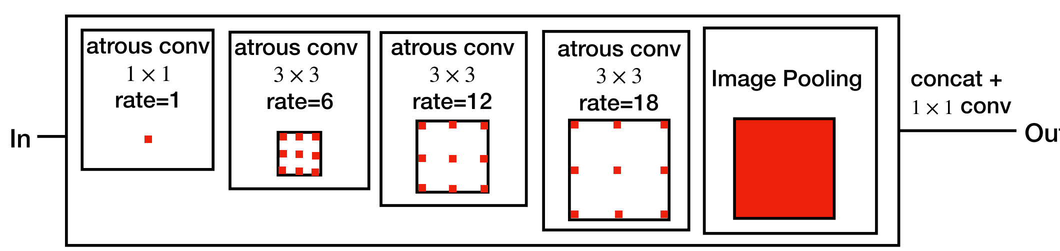


Fig. 2. (Left) Illustration of the aorta and coronary arteries (https://en.wikipedia.org/wiki); (Right) Illustration of CT images of three orthogonal 2D planes and the corresponding aorta and coronary arteries.



(a) UNet++ASPP.



(b) ASPP.

Fig. 3. The architecture of the 2D CNN used in the first stage of our method. (a) The architecture of the UNet++ASPP network. The numbers below the modules represent the number of feature maps at each scale. "Black" indicates the original UNet [10], "blue" shows the nested and dense skip connections in UNet++ [15], and "green" indicates the atrous spatial pyramid pooling (ASPP) module. Components are colored to distinguish between UNet, UNet++, and UNet++ASPP. (b) The architecture of the ASPP module. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

UNet through a dimension conversion layer.

In the present paper, we propose a two-stage strategy to fuse a 2D network with a large field of view and a 3D network that keeps the inter-slice connectivity, in order to obtain accurate segmentation of the aorta and coronary arteries from CT images, the network is shown in Fig. 1. In the first stage, a single 2D UNet is used to segment the aorta and coronary arteries simultaneously in a slice-by-slice fashion. The network receives each slice of the CT image as its input and outputs the category (aorta, coronary arteries, or background) for each pixel. To enhance the multi-scale features, we use the atrous spatial pyramid pooling (ASPP) module [35] to concatenate feature maps generated by atrous convolution with different dilation rates, which helps to resolve ambiguous cases and results in more robust classifications. Following the idea of UNet++ [15], we use the nested and dense skip connections between the downsampling path and the upsampling path, which has been shown to reduce the semantic gap between feature maps and thus can more effectively capture the fine-grained details. In addition, to alleviate the highly unbalanced segmentation problem, we investigate the performances of various objective functions and then utilize a hybrid loss function that combines a generalized dice loss and a cross-entropy loss to train the 2D CNN. In the second stage, a 3D UNet with residual skip connections [36] is applied to further refine the segmentation of the coronary arteries in some candidate regions obtained in the first stage. The 3D CNN receives 3D patches as its input to fully explore the inter-slice connectivity. Only the positive voxels and the false-positive voxels from the first stage are used to train the 3D CNN. Additionally, with the segmentation result of the aorta in the first stage, we can further refine the coronary arteries by using connected component analysis to discard some spurious responses. The main ideas behind the two-stage strategy are as follows: 1) because of the large field of view, the 2D CNN can extract long-range contextual and location information, which can localize the aorta and coronary arteries simultaneously and exclude pseudo-positive voxels for the aorta and coronary arteries; 2) the 3D CNN can extract inter-slice information, which improves the continuity between slices and reduces the missed detection rate of the coronary arteries from the first stage; 3) in the second stage, by focusing on the candidate regions, it can ease the highly unbalanced segmentation problem and reduces the training time compared with directly using a 3D CNN, because the vast majority of negative voxels are excluded in the first stage.

The rest of the paper is organized as follows. In Section 2, the dataset and the detailed design of the framework are presented and discussed. We report the experiments and results in Section 3. Finally, conclusions are drawn in Section 4.

2. 2D and 3D context information fusion by two-stage convolutional neural networks

In this section, we propose a two-stage strategy to achieve accurate segmentation of the aorta and coronary arteries from CT images. In the first stage, a 2D CNN, referred to as UNet++ASPP, is proposed to extract large-field-of-view information to segment the aorta and coronary arteries simultaneously. The UNet++ASPP borrows the spirit of ASPP and the nested and dense skip connections of the UNet++ to fully explore multi-scale features for accurate segmentation of the aorta and coronary arteries. Additionally, to alleviate the highly unbalanced segmentation problem for the 2D CNN, we study some loss functions and their hybrid loss functions. In the second stage, a 3D UNet with residual skip connections, referred to as 3D ResUNet, is applied to extract inter-slice information to further refine the segmentation of the coronary arteries obtained in the first stage. The inter-slice information is expected to improve the continuity between slices and reduces the missed detection rate of the coronary arteries from the first stage.

2.1. Dataset

In this paper, we aim to segment the aorta and coronary arteries from CT images. As shown in Fig. 2, the coronary arteries consist of the right coronary artery (RCA) and left coronary artery (LCA), that originate from the aorta just above where it exits the left ventricular chamber of the heart. The dataset used in this paper comes from 59 patients (i.e., 59 sets of CT images) with CAD. Each set of images consists of 275 slices of 2D images of size 512×512. The dataset is separated into a training set (with 34 sets), a validation set (with 5 sets), and a testing set (with 20 sets), all of whose voxels are manually labelled by clinical experts into three classes: the aorta, the coronary arteries and the background. Another challenge is that highly unbalanced problem occurs in coronary artery segmentation; for example, for all 59 sets of CT images in the experiments of this paper, the average proportion of the coronary artery voxels to the whole volume is only approximately 0.816%.

2.2. 2D convolutional networks for large-field-of-view information extraction

In the first stage, we create a 2D CNN to process each slice of the images to extract large-field-of-view information. The network is constructed by incorporating the atrous spatial pyramid pooling (ASPP) module into a modified UNet architecture, named UNet++ [15]. Such an approach has a built-in mechanism for multi-scale feature learning, and will be referred to as the UNet++ASPP network in this paper.

The UNet model (shown in black in Fig. 3 (a), [10]) consists of a downsampling path and an upsampling path, in which the feature maps in the upsampling path concatenate with those from the downsampling path. As illustrated in Fig. 3, every step in the downsampling path consists of two 3×3 convolutional layers (with each consisting of a convolution, a batch normalization (BN), and a rectified linear unit (ReLU)), followed by a downsampling layer with stride 2. In the upsampling path, every step consists of an upsampling layer with stride 2, a concatenation with the corresponding feature maps from the downsampling path, and two 3×3 convolutional layers. Moreover, the number of feature channels is doubled in the downsampling layer, and it is halved in the upsampling layer. The skip connections between the downsampling and upsampling paths enable the extraction of the precise localization information and long-range context [10]. Furthermore, UNet++ [15], as shown in black and blue in Fig. 3 (a), re-designs the skip pathways in UNet by using the nested and dense skip connections. Instead of directly receiving the feature maps from the downsampling path, in UNet++, the feature maps pass through a dense convolution block whose number of convolutional layers depends on the pyramid level. These re-designed skip pathways can reduce the semantic gap between the feature maps of the downsampling path and those of the upsampling path [15].

Additionally, we exploit a multi-scale feature learning mechanism, i.e., the ASPP module [35] (as shown in Fig. 3 (b)), in the bottom layer of UNet++, which is called UNet++ASPP and is shown in black, blue and green in Fig. 3 (a). Besides, the network architecture that combines the UNet and the ASPP module is called the UNetASPP. ASPP is proposed to concatenate feature maps generated by atrous convolutions with different dilation rates. The output $\boldsymbol{y}$ of the atrous convolution of an input $\boldsymbol{x}$ with a convolutional kernel $\boldsymbol{w}$ is defined by

$$\boldsymbol{y}[\boldsymbol{i}] = \sum_{k} \boldsymbol{x}[\boldsymbol{i} + r \cdot \boldsymbol{k}]\boldsymbol{w}[\boldsymbol{k}], \tag{1}$$

where $\boldsymbol{r}$ denotes the rate parameter corresponding to the stride with which the input signal $\boldsymbol{x}$ is sampled. Standard convolution is a special case corresponding to $\boldsymbol{r} = 1$. Atrous convolution allows us to adaptively modify the filter's field of view by changing the rate value so that the neurons in the output feature map of the ASPP module contain multiple receptive field sizes, which encode multi-scale information and eventually boost the performance. Inspired by [35], in the ASPP module, four 3×3 atrous convolutions with dilated rates $\boldsymbol{r} = 1$, 6, 12, 18 and one global average pooling are carried out in parallel, which are then concatenated and passed through another 1×1 convolution.

As mentioned in the introduction, the large receptive field of 2D CNNs can help learn long-range contextual information. Thus, we are interested in the receptive field of a CNN, which is defined as the size of the region in the input that produces the feature [37]. For simplicity, similar as [37], it is assumed that there is a single path from input to output and the input and feature maps are 1D signals. For higher-dimensional signals, the derivations can be applied to each dimension independently. Additionally, when regarding the combination of features from different scales or even the concatenations (e.g., through skip connections), the receptive field size refers to the largest size among all the paths from input to output. As in [37], r_l is denoted as the receptive field size of the final output feature map with respect to feature map in the l-th layer, and the general recurrence equation can be written by [37]:

$$r_{l-1} = s_l \cdot r_l + (k_l - s_l), \tag{2}$$

where k_l and s_l denote the kernel size and the stride of the convolutional kernel in the l-th layer. Then, for a CNN with L layer, the receptive field size of the final output feature map with respect to the input is defined as [37]:

$$r_0 = \sum_{l=1}^{L} \left((k_l - 1) \prod_{i=1}^{l-1} s_i \right) + 1. \tag{3}$$

If not specified otherwise, by the receptive field size of a CNN, we will always mean r_0, that is, the size of the region in the input that

Table 1
The number of learnable parameters and receptive field size of different 2D networks.

Network structure	UNet	UNet++	UNetASPP	UNet++ASPP
Number of parameters	7.85 M	9.16 M	12.31 M	13.62 M
Receptive field size	205 × 205	205 × 205	512 × 512	512 × 512

produces the output features. Additionally, if the receptive field size is beyond the input size, the input size is regarded as the receptive field size. Table 1 shows the receptive field size of the 2D networks,[1] which indicates that: 1) the nested and dense skip connections in UNet++ have no effect on the receptive field size compared with UNet; 2) the ASPP module significantly enlarges the receptive field size of the network due to its atrous convolutions.

Medical image segmentation for regions that represent a very small fraction of the full image is always a challenge. When the segmentation process targets rare observations, image semantic segmentation requires pixel-wise labelling and small-volumed organs contribute less to the loss; then, a severe class unbalance occurs between candidate labels, thus resulting in sub-optimal performance. As stated in Section 2.1, the average proportion of the coronary artery voxels to the whole volume is only approximately 0.816%. Several loss functions have been proposed to alleviate the highly unbalanced segmentation problem, such as the generalized dice loss, the weighted cross-entropy loss, and the dice loss. In this paper, we investigate six loss functions for training the 2D CNN: the generalized dice loss (GDL) [38], the cross-entropy loss (CEL), the focal loss (FL) [39], and the hybrid loss functions that combine two of the previous three. The generalized dice loss uses the class re-balancing properties of the generalized dice overlap, a known metric for segmentation evaluation, as a loss function for the class unbalance problem. Let $\{r_{ln}\} \in \{0,1\}^{L\times N}$ represent the ground truth over N voxels for an L-class problem and $\{p_{ln}\} \in [0,1]^{L\times N}$ represent the predicted probabilistic map. Then, the GDL can be expressed as:

$$\mathscr{L}_{\mathrm{GDL}} = 1 - 2\frac{\sum_{\ell}^{L} w_\ell \sum_n^N r_{ln}p_{ln} + \varepsilon}{\sum_{\ell}^{L} w_\ell \sum_n^N r_{ln} + p_{ln} + \varepsilon}, \tag{4}$$

where the weight of each class is inversely proportional to the squared volume of the label of this class, i.e., $w_\ell = 1/(\sum_{n=1}{}^{N} r_{\ell n})^2$, and $\varepsilon = 10^{-5}$ is used to ensure the loss function stability by avoiding the numerical issue of dividing by 0. Another loss function, the cross-entropy loss, is commonly used in the pixel-wise semantic segmentation task, which is defined by

$$\mathscr{L}_{\mathrm{CEL}} = -\frac{1}{N}\sum_n^N \sum_\ell^L r_{ln} log p_{ln}. \tag{5}$$

The cross-entropy loss treats each voxel equally, without considering the class unbalance problem. As a variant of the cross-entropy loss, the focal loss focuses on training on a sparse set of poorly classified voxels and preventing the vast number of easily classified voxels from overwhelming the model during training, which can be represented as:

$$\mathscr{L}_{\mathrm{FL}} = -\frac{1}{N}\sum_n^N \sum_\ell^L r_{ln}(1-p_{ln})^2 log p_{ln}. \tag{6}$$

In addition, we consider the hybrid loss functions consisting of contributions from two of these three loss functions. Formally, the hybrid loss function can be expressed as:

$$\mathscr{L} = \mathscr{L}_1 + \alpha \mathscr{L}_2, \tag{7}$$

where $\mathscr{L}_1, \mathscr{L}_2 \in \{\mathscr{L}_{\mathrm{GDL}}, \mathscr{L}_{\mathrm{CEL}}, \mathscr{L}_{\mathrm{FL}}\}$ and α is the trade-off between two losses. We discuss choice of the loss function in Section 3.

2.3. 3D convolutional networks for inter-slice information extraction

In the first stage of the algorithm, the region of interest of the coronary arteries is obtained. Then, in the second stage, as shown in Fig. 1, a 3D CNN is applied to further refine the segmentation of the coronary arteries by extracting the inter-slice information that is ignored in the first stage.

By focusing on the region of interest, the number of 3D regions fed into the 3D CNN during both training and testing can be reduced. Moreover, in the training phase, only the positive voxels and the false-positive voxels from the first stage are used. This training strategy can ease the highly unbalanced segmentation problem since the vast majority of negative voxels are excluded in the first stage. In the testing phase, only the neighbourhood of the voxels classified as coronary arteries in the first stage is refined by the trained 3D CNN. More formally, denote the region of interest of the coronary arteries obtained in the first stage as Ω_{ROI}, and denote the coronary artery segmentation result for the region Ω_{ROI} in the first stage and in the second stage as S_{2D} and S_{3D}, respectively. S_{2D} and S_{3D} are 3D tensors with elements equal to 0 or 1, where 1 denotes the coronary artery and 0 the non-coronary artery. Then, the segmentation refined by the 3D CNN is given by

$$S_{\mathrm{refine}} = S_{3D} \odot f\left(S_{2D} * K_{\mathrm{refine}}\right), \tag{8}$$

where $\odot$ denotes the element-wise product and the 1-value voxels in $f(S_{2D} * K_{\mathrm{refine}})$ are called the candidate regions for the coronary arteries. These candidate regions are obtained by applying a binary dilation to the regions classified as coronary arteries in the first stage, which can be performed by a 3D convolutional operation. In $f(S_{2D} * K_{\mathrm{refine}})$, $*$ and K_{refine} denote the convolutional operator and the kernel of the binary dilation operator, respectively, and $f(x)$ equals 1 for $x > 0$ and 0 for $x = 0$. In this paper, we set K_{refine} as an all-ones 3D tensor of size 15 × 15 × 15. That is, all patches of size 15 × 15 × 15 centered on the voxels classified as coronary arteries are regarded as the candidate regions.

After the segmentation, we post-process the result by finding the largest connected component and discarding the other responses. This is done by first apply a binary dilation to the segmentation and then find the largest connected component. Note that the post-processing only discards some spurious responses and that the result of the dilation operation is not included in the segmentation result. More formally, with the binary segmentation result of the aorta and coronary arteries denoted by S_{aorta} and S_{coronary}, respectively, the final segmentations of the aorta and coronary arteries are given by

$$\begin{aligned} \widetilde{S}_{\mathrm{aorta}} &= S_{\mathrm{aorta}} \odot LCC\left(f\left(\left(S_{\mathrm{coronary}} \vee S_{\mathrm{aorta}}\right) * K_{\mathrm{final}}\right)\right), \\ \widetilde{S}_{\mathrm{coronary}} &= S_{\mathrm{coronary}} \odot LCC\left(f\left(\left(S_{\mathrm{coronary}} \vee S_{\mathrm{aorta}}\right) * K_{\mathrm{final}}\right)\right) \odot \left(\overline{\widetilde{S}_{\mathrm{aorta}}}\right), \end{aligned} \tag{9}$$

where K_{final} denotes the kernel of the binary dilation operator, f is the same as that in Eq. (8), $\overline{\widetilde{S}_{\mathrm{aorta}}}$ denotes the binary negation of $\widetilde{S}_{\mathrm{aorta}}$, $\vee$ denotes the element-wise logical OR operator, and LCC denotes the largest connected component operation, which outputs 1 if the voxel belongs to the largest connected component and 0 otherwise. In this paper, we set K_{final} as an all-ones 3D tensor of size 7 × 7 × 7. Note that, for the sizes of K_{refine} and K_{final}, there is a trade-off between reducing missed detection rate and increasing false positive samples. More specifically, as the sizes of K_{refine} and K_{final} increases, the missed detection rate of the coronary arteries is reduced, while the number of false positive samples increases. The sizes of K_{refine} and K_{final} are empirically chosen based on the performance of the validation data.

For the second stage, the network is constructed using the popular UNet shown in Fig. 3, in which the 2D convolutions are replaced with 3D convolutions. Inspired by [16], two modifications are made: (1) replacing concatenation joining with summation joining between the

[1] The receptive field size is computed using the software: https://github.com/fornaxai/receptivefield.

Table 2
Comparisons of the performances of different 2D models on the testing data.

Network structure	Coronary artery				Aorta			
	DCS (%)	HD (mm)	ASSD (mm)	Precision (%)	DCS (%)	HD (mm)	ASSD (mm)	Precision (%)
UNet	81.15 ± 4.90	15.86 ± 17.25	1.47 ± 1.27	83.92 ± 6.54	96.57 ± 1.96	3.86 ± 5.75	0.57 ± 0.35	97.57 ± 1.24
UNet++	81.82 ± 4.65	20.98 ± 20.45	1.97 ± 2.36	83.13 ± 7.92	97.38 ± 1.40	5.75 ± 16.84	0.56 ± 0.65	97.50 ± 1.06
UNetASPP	82.89 ± 4.24	9.57 ± 7.47	0.93 ± 0.50	82.64 ± 6.49	96.72 ± 2.37	**2.01 ± 1.82**	**0.40 ± 0.32**	**98.29 ± 0.84**
UNet++ASPP	**84.11 ± 4.57**	**7.40 ± 5.96**	**0.82 ± 0.55**	**85.55 ± 7.25**	**97.51 ± 0.75**	2.31 ± 2.48	0.51 ± 0.39	97.54 ± 1.26

The bold indicates the best metrics among all the four networks.

Table 3
Comparisons of the performances of UNet++ASPP trained with different loss functions on the testing data.

Loss functions			Coronary artery				Aorta			
GDL	CEL	FL	DCS (%)	HD (mm)	ASSD (mm)	Precision (%)	DCS (%)	HD (mm)	ASSD (mm)	Precision (%)
√			81.59 ± 6.04	10.57 ± 8.94	1.02 ± 0.47	85.31 ± 5.50	93.10 ± 6.00	7.00 ± 7.33	1.31 ± 1.64	95.60 ± 6.71
	√		78.22 ± 7.23	19.57 ± 23.67	2.81 ± 5.40	84.95 ± 9.40	95.06 ± 5.67	3.45 ± 3.72	0.68 ± 0.73	**97.74 ± 2.46**
		√	79.68 ± 5.84	11.50 ± 9.03	1.15 ± 0.72	**86.72 ± 4.57**	96.00 ± 3.46	4.03 ± 6.55	0.69 ± 0.80	97.69 ± 2.53
√	√		**84.11 ± 4.57**	**7.40 ± 5.96**	**0.82 ± 0.55**	85.55 ± 7.25	**97.51 ± 0.75**	**2.31 ± 2.48**	**0.51 ± 0.39**	97.54 ± 1.26
√		√	83.37 ± 4.97	12.36 ± 10.84	1.10 ± 0.85	82.13 ± 7.50	96.89 ± 2.54	3.11 ± 6.99	0.55 ± 0.87	96.55 ± 3.65
	√	√	79.00 ± 6.88	10.64 ± 9.14	1.13 ± 0.71	85.57 ± 5.50	95.36 ± 5.75	5.91 ± 10.13	0.94 ± 1.70	95.18 ± 5.64

The bold indicates the best metrics among all the loss functions.

Table 4
Comparisons of the coronary artery segmentation performances on the testing data between the first stage and the second stage of our method. UNet, UNet++, and UNet++ASPP are used in the first stage, respectively.

	DCS (%)	HD (mm)	ASSD (mm)	Sensitivity (%)	Specificity (%)	Precision (%)
(a) UNet						
The first stage	81.15 ± 4.90	15.86 ± 17.25	1.47 ± 1.27	79.39 ± 9.04	99.972	83.92 ± 6.54
The second stage	**85.85 ± 4.60**	**8.91 ± 8.96**	**0.91 ± 0.95**	**86.68 ± 7.58**	**99.989**	**85.83 ± 7.58**
(b) UNet++						
The first stage	81.82 ± 4.65	20.98 ± 20.45	1.97 ± 2.36	81.64 ± 8.57	99.970	83.13 ± 7.92
The second stage	**85.52 ± 5.41**	**11.47 ± 14.72**	**1.24 ± 1.90**	**87.49 ± 6.61**	**99.972**	**84.84 ± 10.49**
(c) UNet++ASPP						
The first stage	84.11 ± 4.57	7.40 ± 5.96	0.82 ± 0.55	83.48 ± 7.82	99.971	**85.55 ± 7.25**
The second stage	**86.62 ± 3.96**	**5.57 ± 5.51**	**0.61 ± 0.43**	**89.52 ± 5.39**	**99.987**	84.54 ± 7.73

The bold indicates the best metrics among the first stage and the second stage.

downsampling path and the upsampling path, and (2) adding a residual skip connection to each module. Due to the use of residual skip connections in the network, we call the network 3D ResUNet. Deep residual networks have shown compelling accuracy and nice convergence properties, as there are fewer layers to propagate through, which reduces the impact of vanishing gradients. Furthermore, replacing a concatenation with an addition can be regarded as a residual learning mechanism, which directly employs residual properties. A residual unit can be expressed as:

$$\boldsymbol{y} = \mathscr{F}(\boldsymbol{x}, \{W_i\}) + \boldsymbol{x}, \tag{10}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are the input and output vectors of this unit, $\{W_i\}$ is a set of weights associated with the residual unit, and $\mathscr{F}$ denotes the residual function, i.e., a stack of three $3 \times 3 \times 3$ convolutional layers in our implementation.

3. Experiments

3.1. Evaluation metrics

In this paper, the evaluation metrics consist of three types of measures: the dice score coefficient (DSC), the 95th-percentile of the Hausdorff distance (HD), and the average symmetric surface distance (ASSD). The DSC is a measure of the spatial overlap between the segmentation result S and the ground truth G, defined by

$$DSC(G,S) = \frac{2|G \cap S|}{|G| + |S|} \cdot 100\%, \tag{11}$$

where $|\cdot|$ denotes the cardinality of a set. A larger value of DSC indicates a higher proximity between the ground truth and the segmentation result. The Hausdorff distance measures the maximal distance between the segmentation results and the ground truth, with a smaller value showing a higher segmentation accuracy. To improve the robustness of the conventional HD, we use the 95th percentile of the distances to suppress the outliers [40], which is defined as

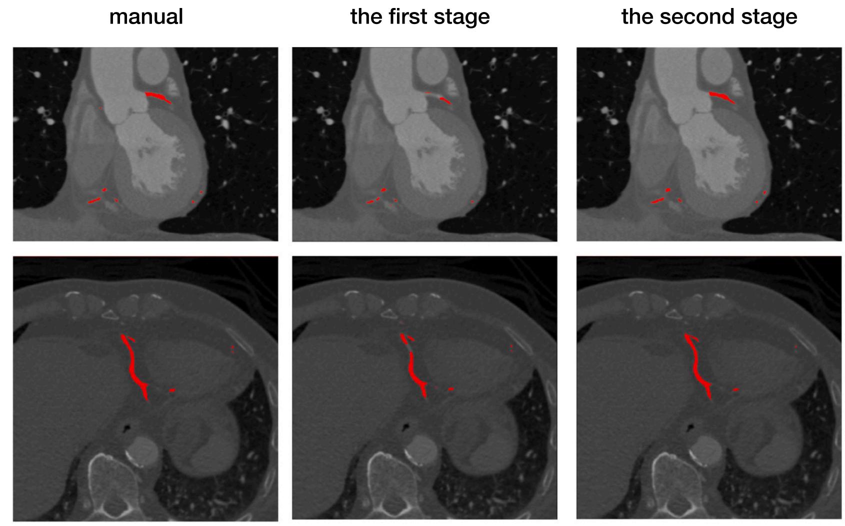
$$HD(G,S) = max\{h_{95}(G,S)\,, h_{95}(S,G)\,\}, \tag{12}$$

where $h_{95}(G,S) = {}^{95}K_{s\in S}{}^{th} d(s,G)$ is the 95th percentile of the distances from all $s \in S$ to G. ASSD is the average of all the distances from points on the boundary of the segmentation result (denoted by B_S) to the boundary of the ground truth (denoted by B_G) and from B_G to B_S, which is defined by [41]:

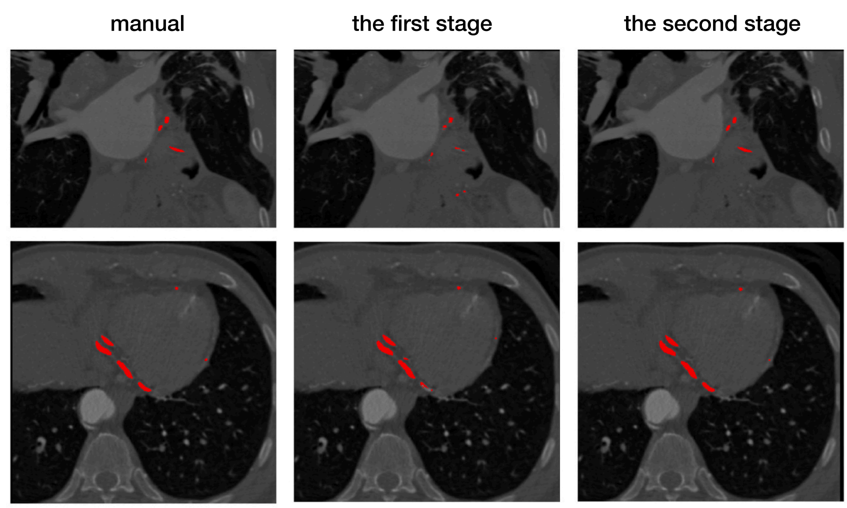
$$ASSD(B_G, B_S) = \frac{1}{|B_G| + |B_S|} \times \left(\sum_{x\in B_G} d(x, B_S) + \sum_{x\in B_S} d(x, B_G) \right) \tag{13}$$

A smaller value of $ASSD(B_G, B_S)$ indicates a better segmentation accuracy.

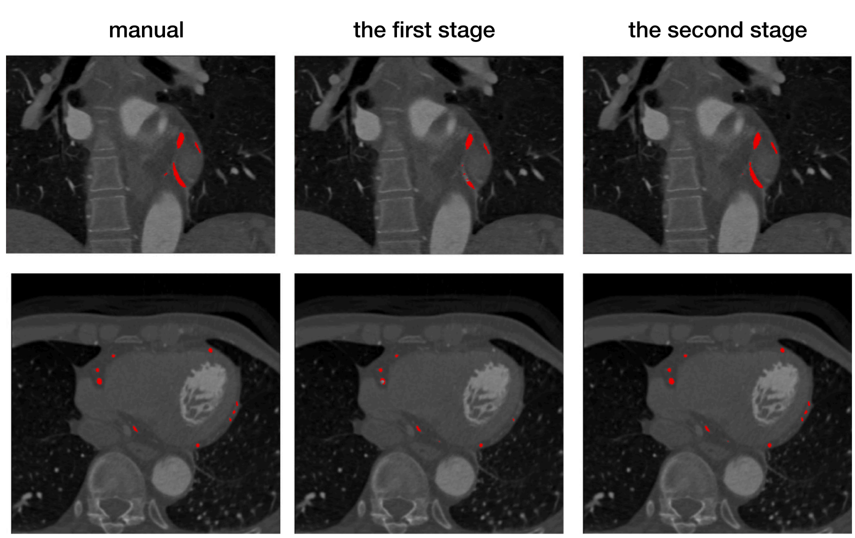
Additionally, to validate that the second stage reduces the missed detection rate of the coronary arteries from the first stage, we use the sensitivity (also known as recall) metric, which measures the proportion



(a) UNet

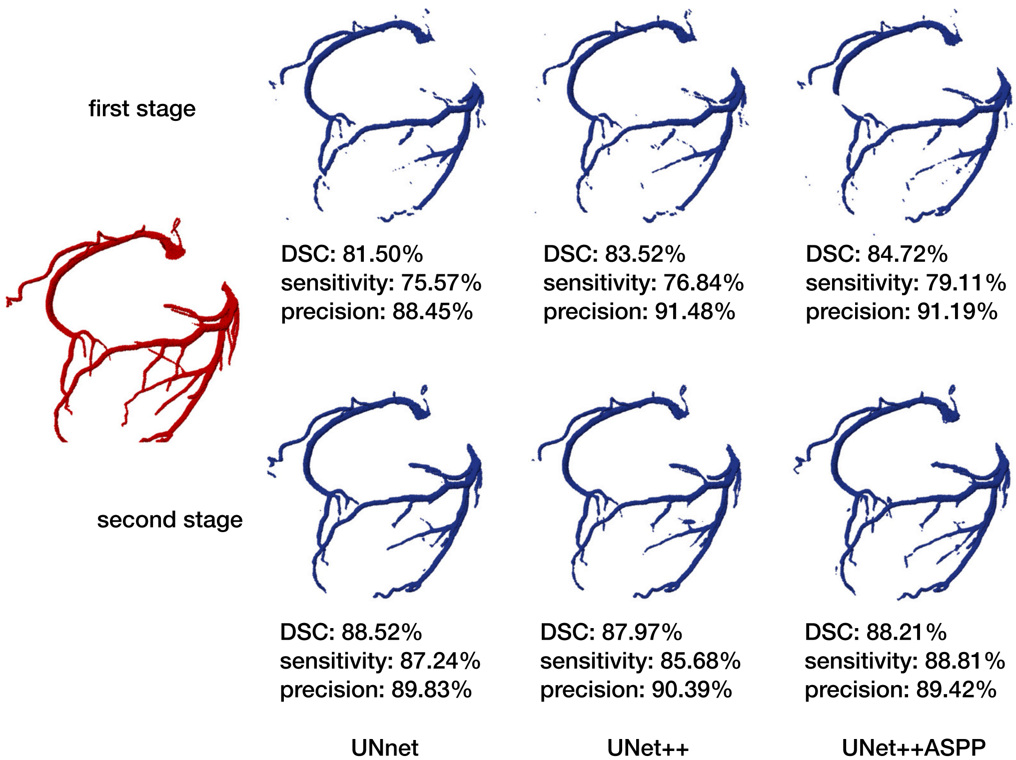


(b) UNet++

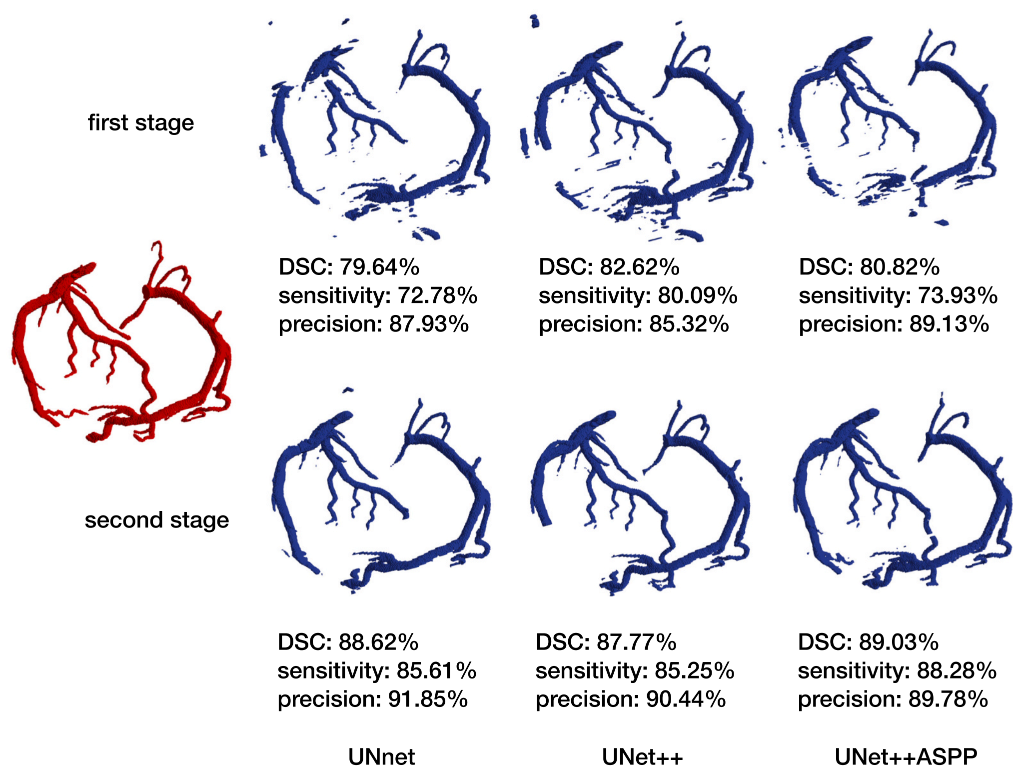


(c) UNet++ASPP

Fig. 4. Segmentation results of different orthogonal 2D planes performed by manual annotation (the first column), the first stage of our method (the second column), and the second stage of our method (the third column). UNet (a), UNet++ (b), and UNet++ASPP (c) are used in the first stage, respectively.



(a) Sample 1



(b) Sample 2

Fig. 5. The comparison between the first and second stage, and the comparison of the use of different 2D networks in the first stage. From left to right are the manual annotation, UNet, UNet++, and UNet++ASPP, respectively.

Table 5
Comparison of the performances on the testing data between our two-stage method and other methods, including methods based on pure 2D CNNs, pure 3D CNNs, and hybrid 2D-3D CNNs.

Network structure		Coronary artery					Aorta		
		DCS (%)	HD (mm)	ASSD (mm)	Sensitivity (%)	Precision (%)	DCS (%)	HD (mm)	ASSD (mm)
Hybrid 2D-3D nets	2D-3D UNet [21]	71.32 ± 8.55	20.53 ± 15.39	2.31 ± 1.54	61.99 ± 13.06	86.36 ± 5.28	96.70 ± 2.41	5.37 ± 10.23	0.64 ± 0.72
	2D-3D ResUNet	71.78 ± 7.03	31.86 ± 12.68	3.07 ± 1.26	66.09 ± 12.02	80.59 ± 6.81	95.88 ± 2.92	3.80 ± 4.85	0.57 ± 0.48
	2.5D CNN [24]	78.31 ± 7.37	12.91 ± 11.02	1.25 ± 0.73	72.10 ± 12.43	**87.43** ± **4.78**	94.34 ± 4.45	7.96 ± 10.68	1.06 ± 0.83
	2.5D UNet++ASPP	80.78 ± 6.91	11.32 ± 9.21	1.21 ± 0.79	77.08 ± 11.23	85.99 ± 5.45	93.12 ± 6.90	6.04 ± 7.17	0.99 ± 1.06
	Multi-view CNN [20]	74.37 ± 6.76	38.95 ± 25.47	2.96 ± 1.33	67.01 ± 11.09	85.62 ± 6.83	73.28 ± 12.91	100.32 ± 33.74	15.63 ± 8.36
	Multi-view UNet++ASPP	81.04 ± 5.17	32.61 ± 22.46	2.74 ± 1.97	82.05 ± 8.01	81.31 ± 9.51	92.59 ± 1.87	130.89 ± 16.85	38.35 ± 10.80
3D nets	3D ResUNet	76.40 ± 8.60	61.76 ± 13.42	8.70 ± 5.54	88.03 ± 6.33	68.77 ± 13.74	–	–	–
	VoxResNet [11]	80.68 ± 6.65	52.59 ± 19.26	5.44 ± 3.97	86.67 ± 6.38	76.67 ± 11.78	–	–	–
	VNet [17]	59.19 ± 8.85	73.83 ± 12.59	17.87 ± 5.69	85.81 ± 6.67	45.75 ± 10.19	–	–	–
2D nets	UNet++ASPP	84.11 ± 4.57	7.40 ± 5.96	0.82 ± 0.55	83.48 ± 7.82	85.55 ± 7.25	97.51 ± 0.75	2.31 ± 2.48	0.51 ± 0.39
Our two-stage method (UNet++ASPP & 3D ResUNet)		**86.62** ± **3.96**	**5.57** ± **5.51**	**0.61** ± **0.43**	**89.52** ± **5.39**	84.54 ± 7.73	**97.54** ± **0.71**	**2.01** ± **2.13**	**0.46** ± **0.35**

The bold indicates the best metrics among all the network structures.

of coronary arteries that are correctly identified and is defined as:

$$sensitivity(G, S) = \frac{|G \cap S|}{|G|} \cdot 100\%, \tag{14}$$

where S and G denote the segmentation result and the ground truth, respectively. A larger sensitivity shows a higher recall rate or a lower missed detection rate. The sensitivity measure is helpful in evaluating the performance of the 3D network of the second stage in reducing the missed detection rate. Furthermore, to further verify whether our two-stage method does not sacrifice the accuracy of non-coronary artery segmentation to improve the sensitivity, we use another metric named specificity that measures the proportion of non-coronary artery that are correctly identified, which is given by:

$$specificity(G, S) = \frac{\left|\overline{G} \cap \overline{S}\right|}{\left|\overline{G}\right|} \cdot 100\%, \tag{15}$$

where $\overline{S}$ and $\overline{G}$ denote the binary negation of S and G, respectively.

To better assess the effect of the transition between the first and second stage, and the impact of the different considered models in reducing false positive samples, we consider the precision score metric, which is the number of voxels correctly labelled as the positive class divided by the total number of voxels labelled as the positive class and is defined as:

$$precision(G, S) = \frac{|G \cap S|}{\left|G \cap S\right| + \left|\overline{G} \cap S\right|} \cdot 100\%. \tag{16}$$

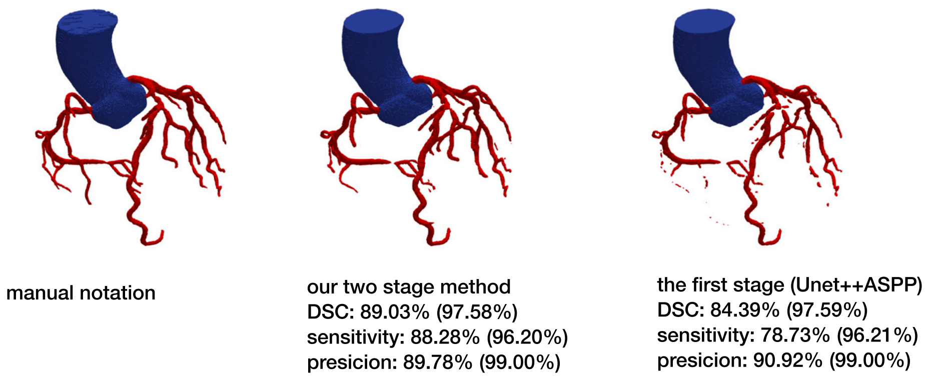
3.2. Implementation details

This section provides the experimental settings. First, we introduce the experimental settings of the 2D CNN and the 3D CNN of the proposed two-stage method in Section 3.2.1. Additionally, to validate the efficacy of our methods in fusing 2D and 3D context information, we further compare our method with some methods based on pure 3D CNNs and other hybrid 2D-3D CNNs, of which the experimental settings are described in Section 3.2.2. The experiments are carried out using the PyTorch framework on a workstation with 2 NVIDIA Tesla V100 32G GPUs. In addition, all the models are trained for 30 epochs and optimized using the mini-batch Adam optimization algorithm [42] with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and an L2 penalty of 0.0001.

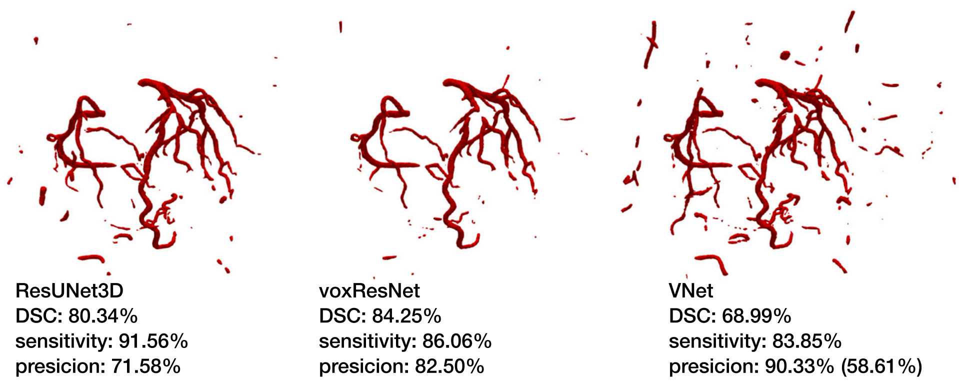
3.2.1. Experimental settings of the two-stage method

3.2.1.1. Training setting for the first stage. For the 2D CNN in the first stage, the CT images are fed into the network slice-by-slice. The size of each slice is 512 × 512. To reduce the variations in the input data, the intensities of each slice are normalized with zero mean and unit variance, and no other image augmentation is used. For the training samples, since the adjacent slices of images are similar, we adopt one slice of every two slices as the training samples to reduce the training time. The learning rate is initially set to 0.0001 and reduced by a factor of 0.2 in the 10th and 20th epochs. The batch size is set as 24. Additionally, the loss function (7) is used to train the 2D CNN, which will be further studied in Section 3.3.2.

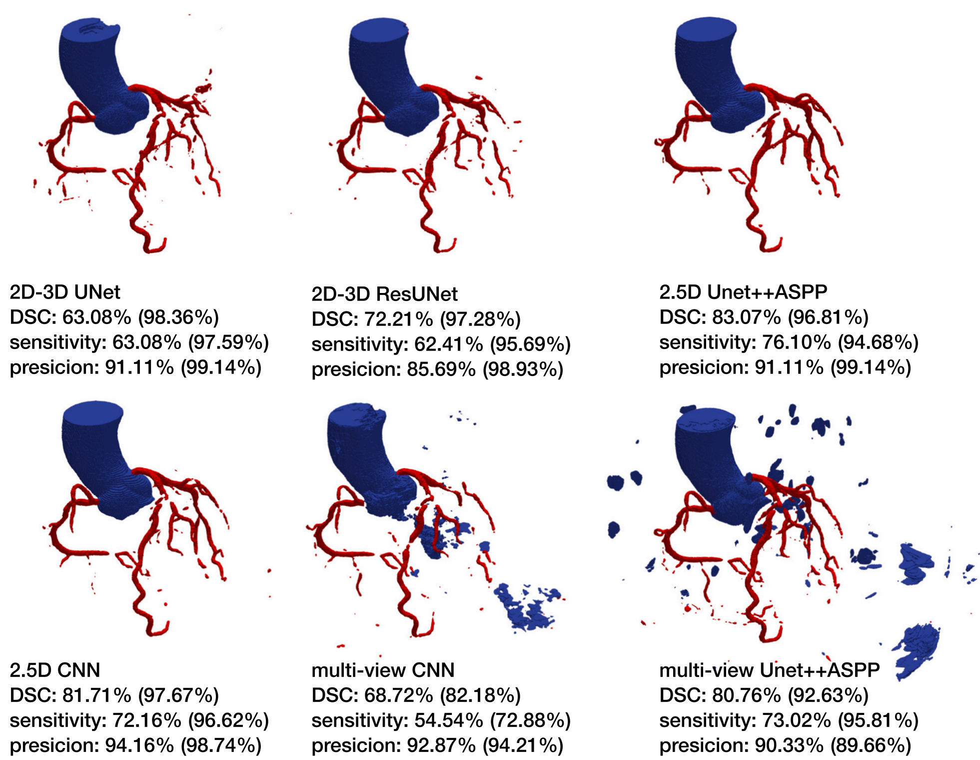
3.2.1.2. Training setting for the second stage. For the 3D CNN in the second stage, we also normalize the intensities of each set of data with zero mean and unit variance. Moreover, for the training and validation phase, the datasets are augmented by using several data augmentation techniques, including random flipping, random rotation, elastic deformation, random contrast, and the addition of random Gaussian or Poisson noise. For the training samples, we first crop sub-volume samples of size 64 × 64 × 64 from the whole volume with stride 6 × 12 × 12. Among these 3D cubes, as stated in Section 2.3, two kinds of cubes are adopted as the training samples: cubes that have more than 160 voxels as coronary arteries in the 21 × 21 × 21 volume in the center of the cube (called positive samples) and cubes randomly picked with a probability of 20% from those cubes (excluding the positive samples) of which the center, of size 21 × 21 × 21, has at least one false-positive voxel from the first stage (called false-positive samples). The validation samples are obtained in a similar way but with a larger stride of 12 × 24 × 24. When UNet++ASPP is used in the first stage, a total of 35,323 sub-volume samples are extracted for training the 3D network, including 28,913 positive samples and 6410 false-positive samples. A total of 704 sub-



(a) Manual notation and segmentation result of our two stage method



(b) Segmentation result of 3D networks



(c) Segmentation result of hybrid 2D-3D networks

Fig. 6. Segmentation results of our two-stage method ((a) for both the first stage and the second stage) and other methods, including pure 3D CNNs (b), and hybrid 2D-3D CNNs (c). Blue for the aorta and red for the coronary artery. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 6
The number of learnable parameters and receptive field size of different networks compared in Table 5.

Network structure		Number of parameters	Receptive field size
Hybrid 2D-3D nets	2D-3D UNet [21]	6.24 M	8 × 205 × 205
	2D-3D ResUNet	8.83 M	4 × 283 × 283
	2.5D CNN [24]	3.50 M	4 × 199 × 199
	2.5D UNet++ASPP	13.62 M	4 × 512 × 512
	Multi-view CNN [20]	13.30 M	95 × 95
	Multi-view UNet++ASPP	40.87 M	512 × 512 (512 × 275)
3D nets	3D ResUNet	35.32 M	64 × 64 × 64
	VoxResNet [11]	6.91 M	64 × 64 × 64
	VNet [17]	45.60 M	64 × 64 × 64
2D nets	UNet++ASPP	13.62 M	512 × 512
Our two-stage method (UNet++ASPP & 3D ResUNet)		13.62 M (first stage)	512 × 512 (first stage)
		35.32 M (second stage)	64 × 64 × 64 (second stage)

volume samples are extracted as the validation data. The learning rate is initially set to 0.0002 and reduced by a factor of 0.2 in the 15th, 20th, and 25th epochs. The batch size is set as 64. We use the dice loss (DL) to optimize the 3D CNN, which originates from the dice score coefficient (11) and is defined by [17]:

$$\mathscr{L}_{\mathrm{DL}} = L - \sum_{\ell=1}^{L} \frac{\sum_n^N 2 r_{ln} p_{ln} + \varepsilon}{\sum_n^N r_{ln} + p_{ln} + \varepsilon}, \tag{17}$$

where the notations are the same as those in Eq. (4).

3.2.1.3. Testing setting. In the testing phase, each slice of CT image is first segmented by the 2D CNN in the first stage. In the second stage, each sub-volume of size 128 × 128 × 128 cropped from the whole volume with stride 64 × 96 × 96 is segmented by the 3D CNN. Then the probability map of the whole volume is generated by an overlap-tiling strategy to stitch the sub-volume results, where the overlapping probabilities are averaged to obtain the final probabilities. At last, the segmentation result of the 3D CNN in the second stage is used to refine the candidate regions of the coronary arteries obtained by the 2D CNN in the first stage according to Eqs. (8) and (9).

3.2.2. Experimental settings of pure 3D CNNs and other hybrid 2D-3D CNNs for comparison

3.2.2.1. Pure 3D CNNs. The pure 3D CNNs are used to segment only the coronary arteries, as the limited field of view of 3D models makes it difficult to distinguish the aorta and other tissues and organs of similar characteristics and thus results in poor segmentation. Three different 3D models are considered: 3D ResUNet (i.e., the model architecture used in the second stage of our method), VNet [17], and VoxResNet [11]. Unlike our two-stage strategy, in which the vast majority of negative voxels are excluded in the first stage, many more training samples need to be included to enhance the robustness of the network when using the pure 3D CNN. For the training samples, we first crop sub-volume samples of size 64 × 64 × 64 from the whole volume with stride 8 × 16 × 16. Two kinds of samples are used to train these pure 3D CNNs, including cubes that have more than 160 voxels as coronary arteries and cubes randomly picked with a probability of 5% from the other cubes. The validation samples are obtained in a similar way but with a larger stride of 16 × 32 × 32. A total of 162,276 training samples (including 127,359 cubes with more than 160 voxels as coronary arteries) are extracted, which is much more than the number of training samples in the second stage of our method. A total of 3303 sub-volume samples are extracted as the validation data. The normalization and data augmentation used in the second stage of our method are also applied to the dataset for these 3D CNNs. The other training settings of the 3D CNNs are identical to those of the 3D CNN in the second stage of our method, except that the initial learning rate is set to 0.002 for VNet since the training progresses very slowly when using an initial learning rate of 0.0002. In the test phase, for the pure 3D CNNs, the testing settings are the same as the 3D CNN in the second stage.

3.2.2.2. Hybrid 2D-3D CNNs. Three different hybrid 2D-3D CNNs are compared with our method:

- 2D-3D UNet [21], which extends the 3D version of UNet by retaining the large field of view of the 2D case but reducing the number of feature maps by half due to the memory constraints.
- 2.5D CNN [24], which takes volumetric image input as multichannel vector images (known as a 2.5D representation) that pass through the first 2D multichannel convolutional layer. The subsequent convolutional operations function exactly the same as those in the 2D methods, excluding the last layer, which outputs 3D segmentation results.
- multi-view CNN [20], which uses a multi-view scheme by utilizing a separate 2D CNN for each orthogonal 2D plane (i.e., the axial plane, sagittal plane, and coronal plane), followed by an adaptive fusion strategy [20] to fuse these three segmentation results.

To show that the performance gain yielded by the two-stage method is not simply due to the well-designed architecture of the network, we apply the strategies of [20,21,24] to the model architecture used in our method, i.e., UNet++ASPP and 3D ResUNet. More specifically, we apply the strategies of [20] and [24] to UNet++ASPP to obtain hybrid 2D-3D CNNs, which are called 2.5D UNet++ASPP and multi-view UNet++ASPP, respectively. In addition, the hybrid 2D-3D strategy in [21] is applied to 3D ResUNet to obtain a hybrid 2D-3D CNN, called 2D-3D ResUNet.

These hybrid 2D-3D CNNs are used to segment the aorta and coronary arteries simultaneously, as in the first stage of our method. We use the following method to obtain training samples for these CNNs:

- for the 2D-3D UNet (or 2D-3D ResUNet) and 2.5D CNN (or 2.5D UNet++ASPP), k consecutive slices of images are taken as the input for the network; that is, the size of the input is k × 512 × 512. To augment the training data, the training samples are extracted with overlap from the whole volumes in a random manner. More precisely, in each interval of $k/2$ slices, we randomly select a slice as the starting slice to extract k consecutive slices. k is chosen from 4,8,16,32,64 based on the performance on the validation data.
- in the multi-view CNN (or multi-view UNet++ASPP), for each orthogonal 2D plane, we adopt one slice out of every two slices as the training samples, as in the first stage of our method. That is, the size of the input for the axial plane is 512 × 512, and the size of the input for the sagittal plane and coronal plane is 275 × 512.

Table 7
Comparison of the deep learning methods in recent years by mean DSC.

Method	Our method	Multi-task CNN ([25], 2016)	DeepMedic ([26], 2017)	3D UNet ([27], 2018)	Multi-scale 3D CNN ([30], 2018)	Context aware 3D FCN ([3], 2018)	Multi-Channel (3D UNet [2], 2019)	Tree-structured ConvGRU ([32], 2020)
DSC (%)	86.62 ± 3.96	60 − 70	58.12	71 − 78	86.49 ± 3.29	79.5 ± 3.6	80.60	86.83

The other training settings are identical to those of the 2D CNN in the first stage of our method, except the batch size, which is adjusted according to the memory constraints. In the test phase, for the 2D-3D UNet (or 2D-3D ResUNet) and 2.5D CNN (or 2.5D UNet++ASPP), each k consecutive slices cropped from the whole volume with the stride of $k/2$ slices is segmented by the networks, and then the probabilities of the overlapping slices are averaged to obtain the final probabilities.

3.3. Choosing network structures and loss functions for the first stage

3.3.1. Network structure

In Section 2.2, we introduced a 2D CNN as shown in Fig. 3. To evaluate the new network, we compare it with two network structures: the traditional UNet (also called "UNet") and the UNet with nested and dense skip connections (called "UNet++"). All the models are trained on the same training dataset with identical training procedures. The performances measured by the three metrics on the testing dataset are summarized in Table 2. Table 1 shows the number of learnable parameters, which indicates that the ASPP module produces an increase of about 4.5 M, and the increase yielded by the nested and dense skip connections in UNet++ is 1.3 M. Compared with the UNet, the UNet++ performs better on the DSC and precision but performs worse on the distance metrics (i.e., the HD and ASSD). Additionally, as shown in Table 1, the ASPP module enlarges the receptive field size of the network, and it also significantly improves the segmentation accuracy with regard to all metrics for both the UNet and UNet++. On the whole, the UNet++ASPP performs the best among the models.

3.3.2. Loss function

We validate the performance of different loss functions in training the 2D CNN UNet++ASPP. As stated in Section 2.2, we compare six loss functions, including the generalized dice loss, the cross-entropy loss, the focal loss, and the hybrid losses that combine of two of these three losses. The trade-off parameter in the hybrid losses (i.e., α in Eq. (7)) is chosen from the three values 0.5, 1 and 2. Based on the average DSCs of the aorta and coronary arteries on the validation dataset, the best parameters are found to be 1, 0.5 and 1 for the hybrid loss composed of GDL and FL, that composed of GDL and CEL, and that composed of CEL and FL, respectively.

The performances of the model trained with the different loss functions are shown in Table 3, with the three evaluation metrics on the testing dataset. According to the results, we conclude that 1) among all the loss functions, the generalized dice loss combined with the cross-entropy loss has the best performance; 2) when using one simple loss, the generalized dice loss outperforms the other two losses for the coronary arteries, but performs poorly for the aorta; and 3) between the hybrid losses, the losses consisting of the generalized dice loss have better performances for the coronary arteries. In general, since the generalized dice loss places more weight on the class of less volume, the generalized dice loss offers a more robust and accurate segmentation for the coronary arteries but leads to worse segmentation for the aorta; however, combined with the cross-entropy loss, which treats all classes equally, this hybrid loss can boost the performance of the aorta segmentation without a loss in accuracy for the coronary arteries.

Based on these observations, the hybrid loss composed of the generalized dice loss and the cross-entropy loss is used to train the 2D CNNs in the first stage and the hybrid 2D-3D CNNs.

3.4. The efficacy of 2D and 3D context information fusion

3.4.1. Refinement of the segmentation of the coronary arteries in the second stage

As shown in Fig. 1, a 2D CNN is used to segment the aorta and coronary arteries simultaneously in the first stage. In the second stage, a 3D CNN is applied to further refine the segmentation of the coronary arteries in the candidate regions. We evaluate this refinement by comparing the segmentation performances on the testing data between the first and the second stages. In the first stage, we use three different networks: UNet, UNet++, and UNet++ASPP. In the second stage, the 3D ResUNet is used. The comparisons between the performances of the first and the second stages are shown in Table 4, Figs. 4 and 5. From the results, we conclude that 1) for the case of UNet or UNet++ used in the first stage, there is a gain in the precision score obtained by the 3D network of the second stage; but in the case of UNet++ASPP, the precision score in the second stage is reduced, which indicates that although the sensitivity score (also known as the recall rate) has increased, the number of false positive samples has also increased, 2) for all the three different 2D CNNs used in the first stage, the 3D network of the second stage can improve the segmentation of the coronary arteries for almost all the evaluation metrics except the precision score for the case of UNet++ASPP, and 3) the 3D network of the second stage can improve the continuity between slices, and reduce the missed detection rate of coronary artery segmentation results without compromising the accuracy of the non-coronary artery segmentation (according to the sensitivity and specificity metrics in Table 4).

3.4.2. Comparison with other methods

To validate the efficacy of our two-stage method, we further compare our method with those based on pure 3D CNNs and those based on hybrid 2D-3D CNNs. As stated in Section 3.2, the 3D CNNs for comparison include 3D ResUNet, VNet [17], and VoxResNet [11], and the hybrid 2D-3D CNNs include the 2D-3D UNet [21], the 2.5D CNN [24], and the multi-view CNN [20]. In addition, as stated in Section 3.2, to show that the performance gain yielded by our two-stage method is not simply due to the well-designed architecture of the network, we apply the hybrid 2D-3D strategies to the 3D ResUNet and UNet++ASPP to obtain hybrid 2D-3D CNNs including 2D-3D ResUNet, 2.5D UNet++ASPP, and multi-view UNet++ASPP. The comparison of the performances of these models on the testing data are shown in Table 5. Fig. 6 shows the comparison of the segmentation results for a single case between manual annotations, our method, and other methods. Besides, Table 6 shows the number of learnable parameters and receptive field size of all the networks compared in Table 5. Note that, for multi-view CNN and multi-view UNet++ASPP, since a separate 2D network is used for each orthogonal 2D plane, the number of parameters is three times that of one network. For the receptive field size, it is 95×95 for all the networks in the case of multi-view CNN. For the case of multi-view UNet++ASPP, it is 512×512 for the network for the axial plane, and 512×275 for the sagittal plane coronal plane. For the pure 2D or 3D models and the hybrid 2D-3D models, we observe that: 1) The 2D network and the hybrid 2D-3D networks, which receive one or several slices of images as input, can obtain satisfactory segmentation results for the aorta (with the DCS of more than 90%), except the multi-view CNN [20], which has a limited field of view of size 95×95 as shown in Table 6. 2) The 3D networks have higher sensitivity to the coronary arteries than the 2D network and the hybrid 2D-3D networks but have higher false-positive rate (or lower precision score). Note that the trade-off between high sensitivity and low false-positive rate is a challenge that occurs in the highly unbalanced segmentation problem. 3) The result of multi-view UNet++ASPP is even worse than that of UNet++ASPP that receives the image of the axial plane as its input, because the poor results of UNet++ASPP that takes the images of the sagittal plane or the coronal plane as input affect the fusion result. Compared with the pure 2D or 3D models and the hybrid 2D-3D models, the advantages of our two-stage method are as follows: 1) for all the evaluation metrics and for both the aorta and the coronary arteries, our two-stage method outperforms all other methods; 2) compared with the pure 2D models, our two-stage method can improve the continuity between slices and achieve a lower missed detection rate for the coronary arteries; and 3) compared with the pure 3D models, our two-stage method can reduce the training time for the 3D CNN in our second stage and decrease the false positive rate for the coronary arteries (according to the precision score in Table 5), since the vast majority of negative voxels are excluded in the first stage. Additionally, from 5 and Fig. 6, we can see that the results of our method have high similarity with those of manual annotations, with the average values of the dice score coefficients for the aorta and coronary arteries being 97.54% and 86.62%, respectively. However, in the results of our method, there is some discontinuity in the coronary artery segmentation; moreover, some distal coronary arteries are not detected. In addition, Table 7 shows the comparison of the mean DSC using different deep learning methods in recent years, which shows that our method attains competitive performance with other methods. Besides, [32] attains the best performance among all the methods, which uses the tree-structured LSTM and centerlines to model the underlying tree structures of coronary arteries. For further improvement, we intend to utilize the centerlines to improve the continuity and the detection of the distal coronary arteries.

3.5. Discussion

Automated segmentation of the aorta and coronary arteries from 3D CT images has great significance in the diagnosis of coronary artery disease. In the current clinical practice, the process of manual segmentation is time-consuming, laborious, and error-prone. To facilitate the diagnosis, we present a two-stage method to circumvent the trade-off between the field of view and the utilization of inter-slice information when using 2D or 3D CNNs for 3D segmentation. The new method retains and combines the merits of 2D and 3D CNNs. Compared with the pure 2D models, the two-stage method can improve the continuity between slices and achieve a lower missed detection rate for the coronary arteries. Compared with the pure 3D models, the two-stage method can reduce the training time and decrease the false positive rate for the coronary arteries, since the vast majority of negative voxels are excluded in the first stage. Extensive experiments demonstrated that our method is also superior to other hybrid 2D-3D methods.

From the experimental results in Section 3.3, the well-designed network architecture (e.g., multi-scale mechanism, and nested and dense connections) and the loss function (e.g., the hybrid loss composed of the generalized dice loss and the cross-entropy loss) together contribute to the better segmentation results. Therefore, we anticipate that the performance of our method will be further improved with the more sophisticated network architecture and training strategy. For example, experiences show that attention mechanism and the fusion of multi-level contextual information would help resolve ambiguous cases and the large inter-subject variations, and thus results in more robust and accurate segmentation. Moreover, in the training of deep neural networks, it usually demands a large number of training samples due to the large number of parameters in the network. We did some comparative experiments with the training dataset of 44 sets (i.e., 10 sets more than the experiments in this paper) and found the results have some noticeable improvement, which indicates that the performance of our method can be further improved with more training data.

Although our method achieved appealing results in most cases, there are still some limitations. As shown in Figs. 5 and 6, there is some discontinuity in the coronary artery segmentation and some distal coronary arteries are not detected. Moreover, since only the neighbourhood of the voxels classified as coronary arteries in the first stage is refined by the 3D CNN in the second stage, the performance of our method depends heavily on the results of the first stage. The blurred and noisy medical images as well as the large inter-subject variations lead to the low accuracy of the segmentation. In future work, we will investigate some techniques to pre-process the images to further improve the performance, such as image deblurring and image enhancement (e.g., vesselness filters [6,28,43]). Furthermore, the use of centerline has been shown to contribute to better results [27,44]. In the future, we shall investigate how to incorporate the information of the centerline into CNNs to further improve the segmentation.

4. Conclusions

In this paper, we present a two-stage strategy to achieve segmentation of the aorta and coronary arteries from CT images, which can retain and combine the merits of 2D and 3D networks. In the first stage, a 2D CNN is used to segment the aorta and coronary arteries simultaneously in a slice-by-slice fashion, which can extract long-range contextual information and thus obtain accurate location information. Then, in the second stage, a 3D CNN is applied to extract the inter-slice information for further refining the segmentation of the coronary arteries obtained in the first stage, which can improve the continuity between slices and improve the recall rate for the coronary arteries. Extensive experiments on clinical CT data show that our method can obtain appealing results and outperform some pure 2D or 3D methods and hybrid 2D-3D methods.

CRediT authorship contribution statement

Linyan Gu: Conceptualization of this study, Methodology, Writing - Original Draft. **Xiao-Chuan Cai**: Supervision, Writing - Review and Editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research was partially supported by the Cooperation on Scientific and Technological Innovation in Hong Kong, Macao and Taiwan as Part of National Key Research and Development Programs: 2021YFE0204300, the National Natural Science Foundation of China under Grant No. 12101589, and the Guangdong Natural Science Foundation under Grant No. 2020A1515110951.

References

[1] Virani SS, Alonso A, Aparicio HJ, Benjamin EJ, Bittencourt MS, Callaway CW, et al. Heart disease and stroke statistics—2021 update: a report from the American heart association. Circulation 2021;143(8):e254–743. https://doi.org/10.1161/CIR.0000000000000950.

[2] Y. Chen, Y. Lin, C. Wang, C. Lee, W. Lee, T. Wang, C. Chen, Coronary artery segmentation in cardiac CT angiography using 3D multi-channel U-net, arXiv: image and video processing.

[3] Duan Y, Feng J, Lu J, Zhou J. Context aware 3D fully convolutional networks for coronary artery segmentation. In: Statistical atlases and computational models of the heart. Atrial segmentation and LV quantification challenges. Springer International Publishing; 2019. p. 85–93. https://doi.org/10.1007/978-3-030-12029-0_10.

[4] Kitslaar P, Frenay M, Oost E, Dijkstra J, Stoel B, Reiber JH. Connected component and morpholgy based extraction of arterial centerlines of the heart (cocomobeach). Midas J 2008;(1). https://doi.org/10.1186/s13640-015-0062-9.

[5] Wang C, Moreno R, Smedby Ö. Vessel segmentation using implicit model-guided level sets. In: MICCAI workshop "3D cardiovascular imaging: a MICCAI segmentation challenge", Nice France, 1st of October 2012; 2012.

[6] D. Han, H. Shim, B. Jeon, Y. Jang, Y. Hong, S. Jung, S. Ha, H. Chang, Automatic coronary artery segmentation using active search for branches and seemingly disconnected vessel segments from coronary CT angiography, PLOS ONE 11 (8). doi:https://doi.org/10.1371/journal.pone.0156837.

[7] Orlando JI, Prokofyeva E, Blaschko MB. A discriminatively trained fully connected conditional random field model for blood vessel segmentation in fundus images. IEEE Trans Biomed Eng 2017;64(1):16–27. https://doi.org/10.1109/TBME.2016.2535311.

[8] Hesamian MH, Jia W, He X, Kennedy P. Deep learning techniques for medical image segmentation: achievements and challenges. J Digit Imaging 2019;32(4):582–96. https://doi.org/10.1007/s10278-019-00227-x.

[9] Haskins G, Kruger U, Yan P. Deep learning in medical image registration: a survey. Mach Vision Appl 2020;31(1):1–18. https://doi.org/10.1007/s00138-020-01060-x.

[10] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. In: International conference on medical image computing and computer-assisted intervention. Springer; 2015. p. 234–41.

[11] Chen H, Dou Q, Yu L, Qin J, Heng P-A. VoxResNet: deep voxelwise residual networks for brain segmentation from 3D MR images. NeuroImage 2018;170: 446–55. https://doi.org/10.1016/j.neuroimage.2017.04.041.

[12] Öztürk S. Stacked auto-encoder based tagging with deep features for content-based medical image retrieval. Expert Syst Appl 2020;161:113693. https://doi.org/10.1016/j.eswa.2020.113693.

[13] Öztürk S. Class-driven content-based medical image retrieval using hash codes of deep features. Biomed Signal Process Control 2021;68:102601. https://doi.org/10.1016/j.bspc.2021.102601.

[14] Hasan MK, Alam MA, Elahi MTE, Roy S, Martí R. DRNet: segmentation and localization of optic disc and fovea from diabetic retinopathy image. Artif Intell Med 2021;111:102001. https://doi.org/10.1016/j.artmed.2020.102001.

[15] Zhou Z, Siddiquee MMR, Tajbakhsh N, Liang J. Unet++: a nested U-net architecture for medical image segmentation. In: Deep learning in medical image analysis and multimodal learning for clinical decision support. Springer; 2018. p. 3–11.

[16] K. Lee, J. Zung, P. H. Li, V. Jain, H. S. Seung, Superhuman accuracy on the SNEMI3D connectomics challenge, arXiv: computer vision and pattern recognition.

[17] Milletari F, Navab N, Ahmadi S-A. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: 2016 fourth international conference on 3D vision (3DV). IEEE; 2016. p. 565–71. https://doi.org/10.1109/3DV.2016.79.

[18] Kamnitsas K, Ledig C, Newcombe VF, Simpson JP, Kane AD, Menon DK, et al. Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. Med Image Anal 2017;36:61–78. https://doi.org/10.1016/j.media.2016.10.004.

[19] Zheng H, Zhang Y, Yang L, Liang P, Zhao Z, Wang C, et al. A new ensemble learning framework for 3D biomedical image segmentation. Proc AAAI Conf Artif Intell 2019:5909–16. https://doi.org/10.1609/aaai.v33i01.33015909.

[20] Mortazi A, Karim R, Rhode K, Burt J, Bagci U. Cardiacnet: Segmentation of left atrium and proximal pulmonary veins from MRI using multi-view CNN. In: International conference on medical image computing and computer-assisted intervention. Springer; 2017. p. 377–85. https://doi.org/10.1007/978-3-319-66185-8_43.

[21] Patravali J, Jain S, Shubham, Chilamkurthy S. 2D-3D fully convolutional neural networks for cardiac MR segmentation. In: International workshop on statistical atlases and computational models of the heart. Springer; 2017. p. 130–9. https://doi.org/10.1007/978-3-319-75541-0_14.

[22] Zagoruyko S, Komodakis N. Wide residual networks. In: Proceedings of the British Machine Vision Conference (BMVC). BMVA Press; 2016. p. . 87.1–87.12. https://doi.org/10.5244/C.30.87.

[23] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. Proc IEEE Conf Comput Vision Pattern Recogn 2016:2818–26. https://doi.org/10.1109/CVPR.2016.308.

[24] Duan J, Bello G, Schlemper J, Bai W, Dawes TJ, Biffi C, et al. Automatic 3D bi-ventricular segmentation of cardiac images by a shape-refined multi-task deep learning approach. IEEE Trans Med Imaging 2019;38(9):2151–64. https://doi.org/10.1109/TMI.2019.2894322.

[25] Moeskops P, Wolterink JM, van der Velden BH, Gilhuijs K, Leiner T, Viergever MA, et al. Deep learning for multi-task medical image segmentation in multiple modalities. In: International conference on medical image computing and computer-assisted intervention. Springer; 2016. p. 478–86.

[26] Kjerland O. Segmentation of coronary arteries from CT-scans of the heart using deep learning (Master's thesis). NTNU; 2017.

[27] Huang W, Huang L, Lin Z, Huang S, Chi Y, Zhou J, et al. Coronary artery segmentation by deep learning neural networks on computed tomographic coronary angiographic images. In: 2018 40th annual international conference of the IEEE engineering in medicine and biology society (EMBC). IEEE; 2018. p. 608–11.

[28] Blaiech AG, Mansour A, Kerkeni A, Bedoui MH, Abdallah AB. Impact of enhancement for coronary artery segmentation based on deep learning neural network. In: Iberian conference on pattern recognition and image analysis. Springer; 2019. p. 260–72.

[29] Shen Y, Fang Z, Gao Y, Xiong N, Zhong C, Tang X. Coronary arteries segmentation based on 3d fcn with attention gate and level set function. IEEE Access 2019;7: 42826–35. https://doi.org/10.1109/ACCESS.2019.2908039.

[30] Chen F, Li Y, Tian T, Cao F, Liang J. Automatic coronary artery lumen segmentation in computed tomography angiography using paired multi-scale 3d cnn. In: Medical imaging 2018: Biomedical applications in molecular, structural, and functional imaging. vol. 10578. International Society for Optics and Photonics; 2018. 105782R. https://doi.org/10.1117/12.2293289.

[31] Wu D, Wang X, Bai J, Xu X, Ouyang B, Li Y, et al. Automated anatomical labeling of coronary arteries via bidirectional tree lstms. Int J Comput Assist Radiol Surg 2019; 14(2):271–80. https://doi.org/10.1007/s11548-018-1884-6.

[32] Kong B, Wang X, Bai J, Lu Y, Gao F, Cao K, et al. Learning tree-structured representation for 3d coronary artery segmentation. Comput Med Imaging Graph 2020;80:101688. https://doi.org/10.1016/j.compmedimag.2019.101688.

[33] Tetteh G, Efremov V, Forkert ND, Schneider M, Kirschke J, Weber B, et al. Deepvesselnet: vessel segmentation, centerline prediction, and bifurcation detection in 3-d angiographic volumes. Front Neurosci 2020;14:1285. https://doi.org/10.3389/fnins.2020.592352.

[34] Liang D, Wang L, Han D, Qiu J, Yin X, Yang Z, et al. Semi 3d-tenet: semi 3d network based on temporal information extraction for coronary artery segmentation from angiography video. Biomed Signal Process Control 2021;69:102894. https://doi.org/10.1016/j.bspc.2021.102894.

[35] Chen L-C, Papandreou G, Kokkinos I, Murphy K, Yuille AL. Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Trans Pattern Anal Mach Intell 2017;40(4):834–48. https://doi.org/10.1109/TPAMI.2017.2699184.

[36] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. Proc IEEE Conf Comput Vision Pattern Recogn. 2016:770–8. https://doi.org/10.1109/CVPR.2016.90.

[37] Araujo A, Norris W, Sim J. Computing receptive fields of convolutional neural networks. Distill 2019;4(11):e21. https://doi.org/10.23915/distill.00021.

[38] Sudre CH, Li W, Vercauteren T, Ourselin S, Cardoso MJ. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In: Deep learning in medical image analysis and multimodal learning for clinical decision support. Springer; 2017. p. 240–8. https://doi.org/10.1007/978-3-319-67558-9_28.

[39] Lin T-Y, Goyal P, Girshick R, He K, Dollár P. Focal loss for dense object detection. In: IEEE transactions on pattern analysis & machine intelligence PP (99); 2017. p. 2999–3007. https://doi.org/10.1109/TPAMI.2018.2858826.

[40] Huttenlocher DP, Klanderman GA, Rucklidge WJ. Comparing images using the Hausdorff distance. IEEE Trans Pattern Anal Mach Intell 1993;15(9):850–63. https://doi.org/10.1109/34.232073.

[41] Heimann T, Ginneken BV, Styner MA, Arzhaeva Y, Aurich V, Bauer C, et al. Comparison and evaluation of methods for liver segmentation from CT datasets. IEEE Trans Med Imaging 2009;28(8):1251–65. https://doi.org/10.1109/TMI.2009.2013851.

[42] D. P. Kingma, J. Ba, Adam: a method for stochastic optimization, arXiv preprint arXiv:1412.6980.

[43] Salem NM, Salem SA, Nandi AK. Segmentation of retinal blood vessels based on analysis of the hessian matrix and clustering algorithm. In: 2007 15th European signal processing conference. IEEE; 2007. p. 428–32.

[44] Shahzad R, Kirilis H, Metz C, Tang H, Schaap M, van Vliet L, et al. Automatic segmentation, detection and quantification of coronary artery stenoses on CTA. Int J Cardiovasc Imaging 2013;29(8):1847–59. https://doi.org/10.1007/s10554-013-0271-1.