Z-Net: an Asymmetric 3D DCNN for Medical CT Volume Segmentation

Peichao Li and Xiao-Yun Zhou and Zhao-Yang Wang and Guang-Zhong Yang

Abstract—Accurate volume segmentation from the Computed Tomography (CT) scan is a common prerequisite for pre-operative planning, intra-operative guidance and quantitative assessment of therapeutic outcomes in robot-assisted Minimally Invasive Surgery (MIS). The use of 3D Deep Convolutional Neural Network (DCNN) is a viable solution for this task but is memory intensive. The use of patch division can mitigate this issue in practice, but can cause discontinuities between the adjacent patches and severe class-imbalances within individual sub-volumes. This paper presents a new patch division approach - Patch-512 to tackle the class-imbalance issue by preserving a full field-of-view of the objects in the XY planes. To achieve better segmentation results based on these asymmetric patches, a 3D DCNN architecture using asymmetrical separable convolutions is proposed. The proposed network, called Z-Net, can be seamlessly integrated into existing 3D DCNNs such as 3D U-Net and V-Net, for improved volume segmentation. Detailed validation of the method is provided for CT aortic, liver and lung segmentation, demonstrating the effectiveness and practical value of the method for intra-operative 3D navigation in robot-assisted MIS.

I. INTRODUCTION

Medical volume segmentation, which labels the class of each voxel in a 3D volume, is a fundamental task in medical image analysis. Computed Tomography (CT), Magnetic Resonance Imaging (MRI) and 3D ultrasound are the common technologies for acquiring medical 3D volumes, and CT is the most popular one among all three types of medical imaging techniques, which is able to retrieve detailed and high-resolution volumetric representations of human structures and records them as voxel values. An illustration of a 3D CT volume and the definition of the three dimensions is shown in Fig. 1(a). MRI and 3D ultrasound are less popular than CT due to the long scan time for MRI and the low resolution for 3D ultrasound volume. In this paper, we mainly focus on medical CT volume segmentation.

In traditional open surgeries where they are operated through a large incision of more than 10 cm, CT volume segmentation was mostly used for pre-operative diagnosis and post-operative assessment. For example, Hybrid Densely Connected UNet (H-DenseUNet) was proposed to segment the liver and tumour from CT volumes, hence to diagnose the hepatocellular carcinoma [1]. CT volume segmentation of the abdominal aortic thrombus was used to assess the Endovascular Aneurysm Repair (EVAR) operation outcomes for treating Abdominal Aortic Aneurysm (AAA) [2].

Recently, due to the emerging of robot-assisted Minimally Invasive Surgeries (MISs) where intelligent robotic surgical tools are inserted through a small incision of less than 2 cm [3], i.e. Laparo-Endo-Scopic Single-site (LESS) surgery, or a natural orifice, i.e. Natural Orifice Transluminal Endoscopic Surgery (NOTES) [4], CT volume segmentation of organs and prostheses is becoming increasingly helpful in intra-operative surgical robotic navigation. For the volume segmentation of organs, the 3D Right Ventricle (RV) mesh from pre-operative CT volume segmentation was an essential input for the mapping vertex determination and efficient robotic path planning in robot-assisted Radio-frequency Cardiac Ablation (RFCA) [5]. 3D aortic segmentation from the pre-operative CT volume was essential to instantiate a safe robotic path for navigating Fenestrated Endovascular Aneurysm Repair (FEVAR) [6]. 3D CT volume segmentation of the liver was used to instantiate the intra-operative and instantaneous 3D liver shapes for navigating robot-assisted liver surgeries [7]. Prosthesis segmentation is less common than organ segmentation. For example, the 3D stent graft and marker shape segmented from the pre-operative CT volume were used to instantiate the intra-operative and 3D shape of a fully-compressed [8], partially-deployed [9] and fully-deployed fenestrated stent graft, improving the navigation for FEVAR from 2D to 3D. In this paper, organ segmentation is our main focus.

Volume segmentation can be realized by either stacking...
the results of multiple 2D image segmentation, which labels the class that each pixel belongs to in a 2D image, or direct volume segmentation, which considers the connection and information between voxels in 3D rather than 2D to achieve higher segmentation accuracy. An illustration of the difference between image and volume segmentation is shown in Fig. 1(c). In image segmentation, all operations including convolutional layers, max-pooling layers and transpose convolutional layers are in 2D, while in volume segmentation, all operations are in 3D. In this paper, we mainly focus on direct volume segmentation methodology. Since the success of AlexNet [11], Deep Convolutional Neural Network (DCNN) has been widely used to replace traditional hand-crafted feature extractors, i.e. edge detector with filters, as it can achieve automatic feature extraction and pixel probability regression with an end-to-end training manner.

DCNN has also been a popular technique for direct 3D volume segmentation. 3D U-Net was first proposed with a contracting encoder part, an expanding decoder part and long skip connections for kidney volume segmentation [12]. Similar to 3D U-Net architecture, V-Net was introduced with a larger $5 \times 5 \times 5$ convolutional kernel and residual learning for prostate volume segmentation [13]. 3D Deeply Supervised Network (DSN) was proposed to demonstrate the effect of both supervising the lower and upper DCNN layers for liver and cardiac volume segmentation [14]. Multi-scale DCNN and Conditional Random Field (CRF) were combined for brain lesion volume segmentation [15]. Holistic Decomposition Convolution (HDC) was proposed to use a larger size of input through decomposed convolutions, resulting in an improved segmentation accuracy for medical volume segmentation [16]. Although these algorithms are different, they share a common technique - patch division which crops the original and large medical CT volume, typically with a size of $512 \times 512 \times L$, $L > 400$, into small patches either randomly or selectively. For example, 3D U-Net [12] cropped the original CT volume into $132 \times 132 \times 116$ patches as the network input. V-Net [13] cropped with a size of $128 \times 128 \times 64$. A size of $160 \times 160 \times 72$ was used in DSN [14] while sizes of $25 \times 25 \times 25$ and $19 \times 19 \times 19$ were used in multi-scale DCNN [15]. An illustration of cropping the sub-volume patches from the original CT volume is shown in Fig. 1(b). There are two reasons for patch division: 1) GPU memory limitation: current GPU memory capacity is usually not able to hold the input of an entire CT volume of a patient; 2) training data limitation: smaller patches allow the augmentation of one training volume into multiple training patches, current public dataset is usually of less than 100 training volumes and this number is insufficient for training 3D DCNNs. However, deficiencies exist for current patch division methods: 1) class-imbalance: some patches might contain very few or no foreground, while some others may have the entire sub-volume labelled as foreground; 2) limited field-of-view in each patch: the network can only perceive a small portion of the entire volume of a patient, and thus is not able to extract a global context. 3) discontinuous segmentation results: the segmentation of the entire CT volume is stacked together from the results of sub-volume patches, which may introduce discontinuities at the boundaries.

In this paper, we propose the Patch-512 method for sub-volume division, which crops the original CT volume into multiple patches with a size of $512 \times 512 \times 8$. Patch-512 has four advantages:

- A full field-of-view along the axial slices (i.e. XY-plane) is maintained.
- Segmentation results are continuous in the axial slices (XY-plane).
- Less class-imbalance exists in the generated patches;
- Sufficient training data in 3D volumes can be augmented;
- Inter-slice information across up to 8 slices can be extracted.

Experiments on the aortic, liver and lung CT volume segmentation will prove a noticeable segmentation accuracy improvement achieved by the proposed Patch-512 method.

Since traditional convolutional kernels are usually symmetrical in all dimensions, whereas Patch-512 introduces asymmetric volume input size along the Z axis for 3D DCNN, it is reasonable to process the features in the XY-plane and the features in the Z axis separately with spatial separable convolutions. The basic principle behind the spatial separable convolution is to divide a high dimensional kernel into several lower dimensional kernels for convolution to reduce the amount of computation, i.e. a $3 \times 3 \times 3$ kernel can be divided into a $3 \times 1$ kernel and a $1 \times 3 \times 3$ kernel. Therefore, the number of multiplications is reduced from 9 to 6. Early separable convolution can be traced back to 2012 when Mamata et al. proposed several methods to simplify the filters in convolutional networks [17]. Sironi et al. further analyzed the separable convolution mathematically, and proposed to use tensor decomposition to get a basis of separable filters for approximation of the high-rank kernels [18]. It was demonstrated that the decomposed convolutions derived from low-rank approximation can reduce the computational complexity without significant changes in the accuracy. Peng et al. adopted the separable convolutions in their network design for semantic segmentation [19]. This research argued that contrary to image classification tasks where smaller kernels and deeper networks might be ideal, segmentation accuracy may benefit from larger kernels. A global convolutional network with massive convolutional kernels was proposed, and spatial separable filters were used to reduce the computational cost.

In this paper, in order to process the asymmetric training patches specifically generated from the proposed Patch-512 method, Z-Net is proposed which separates the convolutions along Z axis from the ones along the XY-planes. For example, a 3D convolutional kernel of size $3 \times 3 \times 3$ is separated into a 2D convolutional kernel of size $3 \times 3 \times 1$ and a 1D convolutional kernel of size $1 \times 1 \times D$, where $D$ is the depth of the input feature map. The kernel size of 1D convolutional kernel is set to $1 \times 1 \times D$ rather than $1 \times 1 \times 3$ to fully extract the inter-slice context among all slices without significantly increase the computational cost. This proposed
Z-Net framework can be seamlessly integrated to the current popular medical volume segmentation DCNNs, i.e. 3D U-Net or V-Net, resulting in ZU-Net and ZV-net respectively. To summarize, the main contributions of this paper are:

- A new and effective patch division method - Patch-512 is proposed for CT volume segmentation, eliminating the class-imbalance and limited field-of-view problem caused by traditional Patch-128 or Patch-64 method.
- A new 3D DCNN framework - Z-Net is proposed, which is suitable for the proposed Patch-512 method. This framework involves separating traditional 3D convolutional kernels into combinations of 2D and 1D convolutional kernels. It can be seamlessly integrated to current and popular 3D DCNNs for CT volume segmentation, including 3D U-Net and V-Net, to Z-U-Net framework can be seamlessly integrated to the current and popular 3D DCNNs for CT volume segmentation, including 3D U-Net and V-Net, to Z-Net variants, namely ZU-Net and ZV-Net. In addition, the proposed Z-Net decreases the trainable parameters and training time significantly.

- aortic, liver and lung CT volumes are used as the validation with detailed ablation analysis. The proposed Patch-512 can achieve an improvement in segmentation accuracy of up to 10.2% compared to Patch-128, while the proposed Z-Net can achieve a further 4.2% improvement in terms of the Intersection over Union (IoU).

The proposed Patch-512, Z-Net and experimental setup are introduced in Sec. II. Detailed validations comparing the proposed Patch-512 and Z-Net to baselines, also segmentation examples and the training loss curve are stated in Sec. III. Discussion and conclusion are shown in Sec. IV and Sec. V respectively.

II. METHODOLOGY

A. Patch-512

A typical volume size for CT scan is $512 \times 512 \times L$, where $L$ is the number of slices along the XY-planes which is varied for different subjects, typically larger than 400. Traditionally, Patch-128 and Patch-64 are common methods for cropping the original CT volume, which represents the patch size of $128 \times 128 \times 64$ and $64 \times 64 \times 64$ respectively. In this paper, Patch-512 is proposed as a new patch division method. For example, to generate the training data, the original CT volume, which represents the patch size of $512 \times 512 \times 64$, and the stride between the successive crops is 1, which leads to $(L - 1)$ augmented training patches from each patient's CT volume.

Patch-512 maintains a full field-of-view in the XY slices while it becomes smaller along the Z axis to feed into a single GPU. As sub-volume divisions will limit the effective field-of-view for the network to perceive the entire volume of a subject, it is of utter importance to keep the spatial integrity of the features as much as possible. Cropping may introduce discontinuities along edges and misalignment between adjacent patches, and this is harmful for the dense volume segmentation task. Therefore, Patch-512 chooses to only crop along the Z axis, instead of cropping along all three dimensions like Patch-64 and Patch-128.

Patch-512 also helps mitigating the class-imbalance problems, which is common in Patch-128 and Patch-64 method, especially for segmentation of small organs. Take aortic CT data as an example, in which only voxels around the center are labelled as the foreground. Patch-64 and Patch-128 cropped successively from the original volume will result in a large portion of sub-volumes cropped from the border regions having no foreground labels at all. If the patches are sampled selectively, the model might produce more false positives along the border regions. Patch-512 ensures the presence of both foreground and background in all the sub-volumes.

B. Z-Net

1) Traditional 3D U-Net and V-Net: The input feature map for a 3D DCNN can be represented as $\mathbf{F} \in \mathbb{R}^{N \times H \times W \times D \times C_{in}}$, where $N$ is the batch size, $H$ is the height, $W$ is the width, $D$ is the depth and $C_{in}$ is the number of channels. Unlike natural RGB images with three channels, CT scan is with a single channel, hence $C_{in} = 1$ when this feature map represents the input patch.

3D convolution uses a trainable 3D convolutional kernel $\mathbf{T} \in \mathbb{R}^{H_T \times W_T \times D_T}$ which slides through the height, width and depth of the input with a stride of $S$ to calculate the convolution. This operation can be represented as:

$$\hat{\mathbf{F}} = \phi(\mathbf{F} \ast \mathbf{T} + b)$$

(1)

where $\hat{\mathbf{F}} \in \mathbb{R}^{N \times H' \times W' \times D' \times C_{out}}$ represents the output feature map, $b$ is the bias, $\phi(\cdot)$ represents the activation function to add non-linearity to the networks, typically Sigmoid function or Rectified Linear Unit (ReLU) function. $C_{out}$ is the number of output feature channels, $H' = H/S$, $W' = W/S$, $D' = D/S$ where $//$ represents the floor division. When $S = 1$, this is normal convolution, resulting in feature maps with the same size of the input feature map, provided that a proper padding method has been adopted. When $S > 1$, the convolutional operation generates a down-sampled feature map. When $S < 1$, the convolutional operation generates an up-sampled feature map, which is named transposed convolution. Another commonly used down-sampling operation is max-pooling, where the maximum values within the $H_T \times W_T \times D_T$ region are extracted to represent the area. After the convolutional operation, $\hat{\mathbf{F}}$ is then passed into the normalization layer, where the mean and variance are calculated as:

$$\mu_{n,c} = \frac{1}{H \times W \times D} \sum_{h=1}^{H} \sum_{d=1}^{D} \sum_{w=1}^{W} \hat{f}_{n,h,w,d,c}$$

(2)

$$\delta_{n,c}^2 = \frac{1}{H \times W \times D} \sum_{h=1}^{H} \sum_{d=1}^{D} \sum_{w=1}^{W} (\hat{f}_{n,h,w,d,c} - \mu_{n,c})^2$$

(3)

where $\hat{f}_{n,h,w,d,c}$ represents each individual voxel value inside the $\hat{\mathbf{F}}$. All data inside the feature map is normalized to a mean of 0 and a variance of 1 to facilitate the training, then is re-scaled by $\gamma_{n,c}$ and re-translated by $\beta_{n,c}$ to maintain the DCNN representation ability.
the expanding path.

layers increases in the contracting path, but it decreases in convolutional layers before down-sampling and up-sampling layers respectively. The number of and transposed convolutional layers are used as the down-convolutional layers. Convolutional layers with a stride of 2 to residual learning [20]. Kernel size of 5 is used for all maps from deep layers to facilitate the training, similar are used to add feature maps from shallow layers to feature 3, is similar to 3D U-Net. In addition, residual connections or 3D transposed convolutional layers.

convolutional layers are used before each 3D max-pooling layers are with a kernel size/stride of 2. Two 3D Max-pooling layers and transposed convolutional layers are used to concatenate feature maps from the contracting path they reach the original resolution. Long skip connections are used to recover the feature maps until they reach the original resolution. Long skip connections are decomposed into 2D convolutions along XY-plane and segmentation, where traditional 3D convolutional operations are decomposed into 2D convolutions along XY-plane and a 1D convolution along Z axis. Such decomposition is illustrated in Fig. [4]

The 2D convolution applies a trainable 2D convolutional kernel \( T_2 \in \mathbb{R}^{H_T \times W_T \times 1} \) that sums the multiplications along the height, width, and depth of the input volume:

\[
\hat{F} = \phi (F \ast T_2 + b)
\] 

(5)

The 1D convolution uses a 1D convolutional kernel \( T_1 \in \mathbb{R}^{1 \times 1 \times D_T} \) that moves along the height, width, and depth of the input:

\[
\hat{F} = \phi (F \ast T_1 + b)
\] 

(6)

In Z-Net, except the down-sampling and up-sampling layers, all 3D convolutional layers are replaced with spatial separable convolutions. It is easy to integrate the proposed Z-Net to traditional 3D U-Net and V-Net architecture. Two versions of the modified DCNN architectures for each network are explored: 1) replace all 3D convolutional layers in 3D U-Net and V-Net with 2D convolutional layers followed by 1D convolutional layers. These changes are reflected by ZU-Net v1 and ZV-Net v1, as shown in Fig. [5]; 2) replace all 3D convolutional layers in 3D U-Net and V-Net with 2D convolutional layers, and only add a single 1D convolutional layer before each down-sampling or up-sampling layer, which means all intermediate 1D convolutions in ZU-Net v1 and ZV-Net v1 are removed. The network architectures, namely ZU-Net v2 and ZV-Net v2, are shown in Fig. [6] Given that Instance Normalization (IN) outperforms other normalization methods including Batch Normalization (BN), Layer Normalization (LN), Group Normalization (GN) as proved in [21], IN was used in this paper for all DCNNs as the default normalization method.

C. Experimental Setup

1) Data collection: 20 aortic CT volumes from VIS-CERAL dataset [22] are used in our experiment. All 20 volumes was randomly shuffled and divided into two groups for 2-fold cross validation. Each group contains 10 volumes for training, 2 groups for evaluation and 8 groups for testing. For each volume, Patch-512 of size \( 512 \times 512 \times 8 \) was sampled, with a stride of 1 along the Z axis for training and a stride of 8 for the evaluation and testing. For comparison, Patch-128 of size \( 128 \times 128 \times 64 \) was generated, with strides of 128, 128, 8 along the X-, Y- and Z- axes for

\[ \hat{f}_{n,h,w,d,c} = \frac{f_{n,h,w,d,c} - \mu_{n,c}}{\sqrt{\sigma^2_{n,c} + \epsilon} \times \gamma_{n,c} + \beta_{n,c}} \] 

(4)

Most 3D DCNNs consist of several convolutional, max-pooling, transposed convolutional, normalization and ReLU layers. Typical examples are 3D U-Net [12] and V-Net [13]. In 3D U-Net, as illustrated in Fig. [2] the contracting encoder part, which contains 3D convolutional layers and 3D max-pooling layers, is used to down-sample the input patch to feature maps in different resolutions, while the expanding decoder part with 3D convolutional layers and 3D transposed convolutional layers is used to recover the feature maps until they reach the original resolution. Long skip connections are used to concatenate feature maps from the contracting path to the expanding path to facilitate information propagation. Max-pooling layers and transposed convolutional layers are used as the down-sampling and up-sampling layers respectively. All convolutional layers are with a kernel size of 3 and max-pooling layers are with a kernel size/stride of 2. Two 3D convolutional layers are used before each 3D max-pooling or 3D transposed convolutional layers.

The network architecture of V-Net, which is shown in Fig. [3] is similar to 3D U-Net. In addition, residual connections are used to add feature maps from shallow layers to feature maps from deep layers to facilitate the training, similar to residual learning [20]. Kernel size of 5 is used for all convolutional layers. Convolutional layers with a stride of 2 and transposed convolutional layers are used as the down-sampling and up-sampling layers respectively. The number of convolutional layers before down-sampling and up-sampling layers increases in the contracting path, but it decreases in the expanding path.
training and a stride of 128, 128, 64 for the evaluation and testing. In order to obtain enough training samples, 90°, 180° and 270° rotations about Z axis was applied for data augmentation. The maximum intensity value of the CT volume for each patient was used to normalize the CT volume intensity within [0, 1].

20 liver CT volumes from the SLiver07 [23] dataset were also used for the validation. All pre-processing procedures are the same as that for the VISCERAL dataset.

60 lung CT volumes from the Lung CT Segmentation Challenge 2017 [24] were used for the validation as well. We followed the instructions from the organizer and divided the 60 CT volumes into 36 and 24 volumes for the training and testing respectively. Hence 2-fold cross validation was not used for this dataset. A single 180° rotation was used for data augmentation. All other pre-processing procedures were the same as that for the VISCERAL dataset.

### 2) Training configurations:

For all the training processes, Stochastic Gradient Descent (SGD) with a momentum of 0.9 was used as the default optimizer. The activation function for all 3D DCNNs was ReLU for consistency, even though originally Parametric ReLU was used in V-Net [13]. Weights were initialized with a truncated normal distribution, and biases were initialized as 0.1. Four initial learning rates (0.1, 0.05, 0.01, 0.005) were tested, and the one that achieved the highest segmentation accuracy was selected as the final result. The learning rate was dropped by half after the first epoch, and was further divided by 10 if the training process was longer than 4 epochs. 6 epochs were trained for the aortic data, while 4 epochs were trained for the liver and lung data. IoU, also known as the Jaccard Index, served as the metric for evaluating the performance of the segmentation:

\[
\text{IoU} = \frac{|Y \cap P|}{|Y \cup P|}
\]

where \( Y \) is the ground truth and \( P \) is the binarized prediction result. Foreground IoU was used to evaluate the segmentation accuracy. The prediction from the network was encoded in the one-hot fashion and cross-entropy loss was used, which can be calculated as:

\[
\xi(y, p) = -(y \log(p) + (1 - y) \log(1 - p))
\]

for binary-classification tasks, where \( y \) is the ground truth value and \( p \) is the prediction value for the segmentation given by the softmax function. All networks were trained using a CPU of Intel Xeon® E5-1650 v4@3.60GHz × 12 and a GPU of Nvidia Titan Xp. The implementations of all the networks were based on TensorFlow.

### III. RESULT

In order to compare the proposed Patch-512 method with the Patch-128 method, the vanilla version of 3D U-Net and V-Net were trained using the training data generated with Patch-128 method, which served as the baseline. Patch-64 was not compared in our experiment due to the convergence difficulty with the aortic dataset, which was as expected. Even multiple pre-processing methods have been used to optimize the patch division, a heavy class-imbalance was still presented using Patch-64 method. The training patches generated from Patch-512 are only with a size of 8 in the Z axis, hence 3D U-Net and V-Net with three down-sampling layers are trained on the Patch-512 data for comparison. Detailed results are stated in Sec. III-A.

In order to validate the proposed Z-Net, ZU-Net v1, ZV-Net v1, ZU-Net v2 and ZV-Net v2 were trained on the training data generated with Patch-512 method. Detailed results are illustrated in Sec. III-B. CT volumes and 2D slices of patients are randomly selected to show the detailed segmentation difference between different methods in Sec. III-C while the training loss curves for different methods are shown in Sec. III-D.
TABLE I
THE IOU RESULTS FOR 3D U-NET (VANILLA) AND V-NET (THREE DOWN-Sampling LAYERS) ON THE TRAINING DATA GENERATED BY PATCH-512 METHOD, COMPARED WITH THE RESULTS FOR 3D U-NET (VANILLA) AND V-NET (VANILLA) ON THE TRAINING DATA GENERATED BY PATCH-128 METHOD. THE HIGHEST VALUES ARE HIGHLIGHTED IN BOLD.

| Patch     | Dataset          | Aorta | Liver | Lung |
|-----------|------------------|-------|-------|------|
|           | Cross Validation | Fold 1| Fold 2| Fold 1| Fold 2 |
| Patch-128 | 3D U-Net (Vanilla) | 0.514 | 0.583 | 0.752 | 0.598 | 0.865 |
|           | V-Net (Vanilla)   | 0.555 | 0.584 | 0.790 | 0.728 | 0.871 |
| Patch-512 | 3D U-Net (Vanilla) | 0.616 | 0.614 | 0.764 | 0.827 | 0.736 | 0.873 |
|           | V-Net (three down-sampling layers) | 0.632 | 0.625 | 0.8 | 0.736 | 0.873 |

TABLE II
COMPARISON BETWEEN THE 3D U-NET (VANILLA), V-NET (THREE DOWN-Sampling LAYERS), ZU-NET V1, ZV-NET V1, ZU-NET V2 AND ZV-NET V2 ON THE AORTA, LIVER AND LUNG TRAINING DATA GENERATED BY PATCH-512 METHOD. THE HIGHEST VALUES ARE HIGHLIGHTED IN BOLD.

| Dataset          | Aorta | Liver | Lung | Parameters | Training Time (per 100 iterations) |
|------------------|-------|-------|------|------------|-----------------------------------|
| Cross Validation | Fold 1| Fold 2| Fold 1| Fold 2     |                                    |
| 3D U-Net (Vanilla) | 0.616 | 0.614 | 0.764 | 0.613 | 0.866 | 1.49 x 10^4 | 70.36s |
| ZU-Net v1        | 0.630 | 0.652 | 0.791 | 0.630 | 0.868 | 0.19 x 10^4 | 67.91s |
| ZU-Net v2        | 0.640 | 0.655 | 0.783 | 0.652 | 0.869 | 5.73 x 10^4 | 67.02s |
| V-Net (three down-sampling layers) | 0.632 | 0.625 | 0.827 | 0.736 | 0.873 | 1.54 x 10^4 | 157.4s |
| ZV-Net v1        | 0.674 | 0.650 | 0.852 | 0.756 | 0.877 | 3.51 x 10^6 | 82.08s |
| ZV-Net v2        | 0.674 | 0.660 | 0.851 | 0.770 | 0.876 | 3.33 x 10^6 | 80.11s |

Fig. 7. One 2D slice from the CT volume of a randomly selected patient is shown with the ground truth, the segmentation result of training 3D U-Net (Vanilla) and V-Net (Vanilla) on patch-128 data and of training ZU-Net v2 and ZV-Net v2 on Patch-512 data. The regions in the red circles show the misalignment in prediction results due to the Patch-128 method.

A. Patch-512
The mean segmentation IoUs of training vanilla 3D U-Net and V-Net on the aorta, liver and lung training data generated from Patch-128, and the mean segmentation IoUs of training 3D U-Net, V-Net with three down-sampling layers on the aorta, liver and lung training data generated by Patch-512 are shown in Tab. I. It can be observed that even though 3D U-Net and V-Net with three down-sampling layers contain less trainable parameters and less layers than vanilla V-Net, the proposed Patch-512 still achieved 3.1% − 10.2%, 0.8% − 3.7%, and 0.1% − 0.2% mean IoU improvements for both the 3D U-Net and V-Net, compared to the traditional Patch-128 method.

One observation is that the order of IoU improvement is aorta > liver > lung, which is in the reverse order of their physical size. This result indicates that large organs such as lung and liver appear to be less affected by different patch size settings, whereas small organs such as aorta rely heavily on how the original volume is divided. An up to 10.2% IoU improvement proves the severe issue of class-imbalance presented by the Patch-128 method in aortic dataset.

B. Z-Net
The mean segmentation IoUs of 3D U-Net and V-Net with three down-sampling layers, as well as ZU-Net v1, ZU-Net v2, ZV-Net v1 and ZV-Net v2 on the aorta, liver and lung
One patient is randomly selected for visualizing the detailed volume segmentation result of vanilla 3D U-Net and vanilla V-Net on the aorta, liver and lung training data generated by Patch-128 and with training ZU-Net v2 and ZV-Net v2 on the aorta, liver and lung training data generated by Patch-512.

Patch-512 training data are shown in Tab. II. We can see that for the aorta and liver segmentation, the ZU-Net and ZV-Net in both modes outperform the original 3D U-Net and V-Net by around 3%. On the other hand, the segmentation IoU improvement for the lung segmentation is much smaller, being 0.1% and 0.3%. It can be concluded that the proposed Z-Net outperforms all baselines in all validations. It also can be concluded that mode 2 outperforms mode 1 in most tests, except cross validation 1 for the liver segmentation.

The number of trainable parameters and the training time for 3D U-Net (Vanilla), V-Net with three down-sampling layers, ZU-Net v1, ZU-Net v2, ZV-Net v1, ZV-Net v2 are also shown in Tab. II. It can be seen that the proposed Z-Net variants possess significantly less trainable parameters. The modified versions of ZU-Net contain less than a half the number of trainable parameters than the 3D U-Net (Vanilla), and the number of parameters for ZV-Net is only around \( \frac{1}{3} \) of the V-Net with three down-sampling layers. The table also shows faster training speed after the modifications of the networks resulting from fewer trainable parameters, especially for V-Net. The training speed is measured as the average amount of time in seconds for the networks to train 100 iterations.

### C. Segmentation examples

One patient was selected to show the detailed 3D segmentation result of the aorta, liver and lung, with the 3D U-Net (Vanilla) and V-Net (Vanilla) training on data generated by Patch-128 method, and with the proposed ZU-Net and ZV-Net under mode 2 training on data generated by Patch-512 method, as shown in Fig. 8. We can see that the proposed method in this paper achieved noticeable better segmentation result with much less false positives and noises.

For better visualization, we also show some detailed 2D slice segmentation results of the aorta, liver and lung with different methods in Fig. 7. It is also obvious that the proposed method in this paper achieved visually better segmentation results without misalignment between patches along X and Y axes.

### IV. DISCUSSION

Patch-512 method was proposed to crop original CT volumes with a size of \( 512 \times 512 \times L \) into patches with a size of \( 512 \times 512 \times 8 \). Comparing to traditional patch methods, i.e. Patch-128 with a size of \( 128 \times 128 \times 64 \) or Patch-64 with a size of \( 64 \times 64 \times 64 \), Patch-512 maintains a full field-of-view in the XY slices while maintains as many slices as possible in the Z axis according to the GPU memory. The effect of cropping along both X and Y axis and then assembling the prediction result back can be seen clearly in Fig. 7, where discontinuities between adjacent predictions result in the large gaps and holes inside the prediction mask. Patch-512 retains the spatial integrity of features along the XY plane, giving a better result observed from 2D slices.

Furthermore, the Patch-512 method not only compensates
the class-imbalance issue caused by Patch-128 and Patch-64, but also augments the number of training data and keeps the GPU memory under an affordable value. Promising improvements on the segmentation accuracy, especially for small targets like the aorta, can be observed from these results, demonstrating the effect of the proposed Patch-512 method.

Two modes of Z-Net were also explored in this paper, either with or without the intermediate 1D convolutions between 2D convolutions. According to the validation results, mode 2 slightly out-performed mode 1, and the number of parameters and the training time are also reduced. This indicates that a single 1D convolution before each down-sampling or up-sampling layer is sufficient for inter-slice context extraction.

Overall, the total improvements in IoU of 12.3%, 7.4% and 0.5% were achieved for the aortic, liver and lung CT volume segmentation respectively with the proposed Z-Net framework and the Patch-512 method, compared to the original 3D U-Net and V-Net with the Patch-128 method. The segmented 3D shape in this paper is very useful for many advanced medical tasks, i.e. 3D shape instantiation and registration for 3D navigation in robot-assisted MIS. 3D DCNN based medical CT volume segmentation also automate these 3D robotic navigation algorithms.

V. CONCLUSION

To summarize, Patch-512 is proposed to alleviate the issue for traditional patch methods in class-imbalance and limited field-of-view for each individual sub-volume. Z-Net is proposed along with the Patch-512 method to extract the information in the XY-planes and Z-axis separately for training with $512 \times 512 \times 8$ patches. Both Patch-512 and Z-Net framework have improved the current volume segmentation accuracy noticeably. The medical CT volume segmentation in this paper both automates and supplies an essential pre-operative knowledge for achieving intra-operative 3D navigation for robot-assisted MISs. In the future, we will work on deeper network designs and better kernel decomposition methods for medical volume segmentation with more trainable parameters to further improve the segmentation accuracy.

VI. ACKNOWLEDGEMENT

We greatly thank Qing-Biao Li and Jian-Qing Zheng for their constructive suggestions and gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

REFERENCES

[1] X. Li, H. Chen, X. Qi, Q. Dou, C.-W. Fu, and P.-A. Heng, “H-DenseUNet: hybrid densely connected UNet for liver and tumor segmentation from CT volumes,” IEEE TMI, vol. 37, no. 12, pp. 2663–2674, 2018.
[2] K. López-Linares, N. Aranjuelo, L. Kabongo, G. Maclair, N. Lete, M. Ceresa, A. García-Familiar, I. Macia, and M. A. G. Ballester, “Fully automatic detection and segmentation of abdominal aortic thrombus in post-operative CTA images using deep convolutional neural networks,” MedIA, vol. 46, pp. 202–214, 2018.
[3] V. Vitiello, S.-L. Lee, T. P. Cundy, and G.-Z. Yang, “Emerging robotic platforms for minimally invasive surgery,” IEEE reviews in biomedical engineering, vol. 6, pp. 111–126, 2012.
[4] N. Liu, M. E. Abdelaziz, M. Shen, and G.-Z. Yang, “Design and kinematics characterization of a laser-profiled continuum manipulator for the guidance of bronchoscopy instruments,” in ICRA. IEEE, 2018, pp. 25–31.
[5] X.-Y. Zhou, S. Ernst, and S.-L. Lee, “Path planning for robot-enhanced cardiac radiofrequency catheter ablation,” in IEEE ICRA. IEEE, 2016, pp. 4172–4177.
[6] Q. Zheng, X.-Y. Zhou, C. Riga, and G.-Z. Yang, “Towards 3D path planning from a single 2D fluoroscopic image for robot assisted fenestrated endovascular aortic repair,” in IEEE ICRA. IEEE, 2019, pp. 8747–8753.
[7] X.-Y. Zhou, G.-Z. Yang, and S.-L. Lee, “A real-time and registration-free framework for dynamic shape instantiation,” MedIA, vol. 44, pp. 86–97, 2018.
[8] X. Zhou, G. Yang, C. Riga, and S. Lee, “Stent graft shape instantiation for fenestrated endovascular aortic repair,” in Proceedings of The Hamlyn Symposium on Medical Robotics. The Hamlyn Symposium on Medical Robotics, 2016.
[9] J.-Q. Zheng, X.-Y. Zhou, C. Riga, and G.-Z. Yang, “Real-time 3D shape instantiation for partially deployed stent segments from a single 2D fluoroscopic image in fenestrated endovascular aortic repair,” IEEE RAL, vol. 4, no. 4, pp. 3703–3710, 2019.
[10] X.-Y. Zhou, J. Lin, C. Riga, G.-Z. Yang, and S.-L. Lee, “Real-time 3D shape instantiation from single fluoroscopy projection for fenestrated stent graft deployment,” IEEE RAL, vol. 3, no. 2, pp. 1314–1321, 2018.
[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in NeurIPS, 2012, pp. 1097–1105.
[12] O. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, “3D U-Net: learning dense volumetric segmentation from sparse annotation,” in MICCAI. Springer, 2016, pp. 424–432.
[13] F. Milletari, N. Navab, and S.-A. Ahmadi, “V-Net: Fully convolutional neural networks for volumetric medical image segmentation,” in JDV. IEEE, 2016, pp. 565–571.
[14] Q. Dou, L. Yu, H. Chen, Y. Jin, X. Yang, J. Qin, and P.-A. Heng, “3D deeply supervised network for automated segmentation of volumetric medical images,” MedIA, vol. 41, pp. 40–54, 2017.
[15] K. Kamnitsas, C. Ledig, V. F. Newcombe, J. P. Simpson, A. D. Kane, D. K. Menon, D. Rueckert, and B. Glocker, “Efficient multiscale 3D CNN with fully connected CRF for accurate brain lesion segmentation,” MedIA, vol. 36, pp. 61–78, 2017.
[16] G. Zeng and G. Zheng, “Holistic decomposition convolution for effective semantic segmentation of medical volume images,” MedIA, vol. 57, pp. 149–164, 2019.
[17] F. Mamalet and C. Garcia, “Simplifying convnets for fast learning,” in ICANN. Springer, 2012, pp. 58–65.
[18] A. Sironi, B. Tekin, R. Rigamonti, V. Lepeit, and P. Fuu, “Learning separable filters,” IEEE TPAI, vol. 37, no. 1, pp. 94–106, 2014.
[19] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, “Large kernel matters—improve semantic segmentation by global convolutional network,” in CVPR, 2017, pp. 4353–4361.
[20] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in CVPR, 2016, pp. 770–778.
[21] X.-Y. Zhou and G.-Z. Yang, “Normalization in training U-Net for 2D biomedical semantic segmentation,” IEEE RAL, vol. 4, no. 2, pp. 1792–1799, 2019.
[22] O. Jimenez-del Toro, H. Müller, M. Krenn, K. Gruenberg, A. A. Taha, M. Winterstein, I. Eggel, A. Foncubierta-Rodriguez, O. Goksel, A. Jakab et al., “Cloud-based evaluation of anatomical structure segmentation and landmark detection algorithms: VISCERAL anatomy benchmarks,” IEEE TMI, vol. 35, no. 11, pp. 2459–2475, 2016.
[23] T. Heimann, B. Van Ginneken, M. A. Styner, Y. Aizhavia, V. Aurich, C. Bauer, A. Beck, C. Becker, R. Beichel, G. Bekes et al., “Comparison and evaluation of methods for liver segmentation from CT datasets,” IEEE TMI, vol. 28, no. 8, pp. 1251–1265, 2009.
[24] J. Yang, H. Veeraraghavan, S. G. Armatto III, K. Farahani, J. S. Kirby, J. Kalpathy-Kramer, W. van Elmpit, A. Dekker, X. Han, X. Feng et al., “Autosegmentation for thoracic radiation treatment planning; A grand challenge at AAPM 2017,” Medical physics, vol. 45, no. 10, pp. 4568–4581, 2018.