A Two-Stage U-Net Model for 3D Multi-class Segmentation on Full-Resolution Cardiac Data

Chengjia Wang[1,2](✉), Tom MacGillivray[2], Gillian Macnaught[1,2],
Guang Yang[3], and David Newby[1,2]

[1] BHF Centre for Cardiovascular Science, University of Edinburgh, Edinburgh, UK
chengjia.wang@ed.ac.uk
[2] Edinburgh Imaging Facility QMRI, University of Edinburgh, Edinburgh, UK
[3] National Heart and Lung Institute, Imperial College London, London, UK

Abstract. Deep convolutional neural networks (CNNs) have achieved state-of-the-art performances for multi-class segmentation of medical images. However, a common problem when dealing with large, high resolution 3D data is that the volumes input into the deep CNNs has to be either cropped or downsampled due to limited memory capacity of computing devices. These operations can lead to loss of resolution and class imbalance in the input data batches, thus downgrade the performances of segmentation algorithms. Inspired by the architecture of image super-resolution CNN (SRCNN), we propose a two-stage modified U-Net framework that simultaneously learns to detect a ROI within the full volume and to classify voxels without losing the original resolution. Experiments on a variety of multi-modal 3D cardiac images have demonstrated that this framework shows better segmentation performances than state-of-the-art Deep CNNs with trained with the same similarity metrics.

Keywords: Image segmentation · Convolutional neural networks · High resolution · Cardiac CT/MR

1 Introduction

Segmenting the whole heart structures from CT and MRI data is a necessary step for pre-precedural planing of cardiovascular diseases. Although it is the most reliable approach, manual segmentation is very labor-intensive and subject to user variability [1]. High anatomical and signal intensity variations make automatic whole heart segmentation a challenging task. Previous methods that separately segment specific anatomic structure [2] are often based on active deformation models. Others perform multi-class segmentation where atlas-based models play an important role. Active deformation models can suffer from limited ability to

This work is funded by BHF Centre of Cardiovascular Science and MICCAI 2017 Multi-Modality Whole Heart Segmentation (MM-WHS) challeng.

M. Pop et al. (Eds.): STACOM 2018, LNCS 11395, pp. 191–199, 2019.
https://doi.org/10.1007/978-3-030-12029-0_21

decouple pose variation [3], and the main disadvantage of atlas-based methods is requiring complex procedures to construct the atlas or non-rigid registration [4]. Recently, deep convolutional neural networks (DCNNs), such as U-Net [5], have been vastly used for cardiac segmentation and achieved start-of-the-art results. But due to limited computational power, the data input into U-Net are always down-sampled to a uniform resolution. The purpose of this study is to develop a DCNN which can perform multi-class segmentation on full-resolution volumetric CT and MR data with no post-prediction resampling or subvolume-fusion operations. This is necessary due to the loss of information introduced by interpolation and extra complexity of post-processing.

The original U-Net is entirely an 2D architecture. So are some DCNN-based full heart segmentation methods [6,7]. To process volumetric data, one solution is to take three perpendicular 2D slices as input and fuse the multi-view information abstraction for 3D segmentation [8]. Or one can directly replace 2D operations in the original U-Net with their 3D counterparts [9]. This 3D U-Net has been applied to multi-class whole-heart segmentation on MR and CT data [10]. However, due to limited memory of GPUs, these methods have to either down-sampled the input volumes, which leads to loss of resolution in the final results, or predict subvolumes of the data make extra efforts to merge the overlapped subvolume predictions. A strategy to reduce the memory load of 3D U-Nets has been used in [11], where a region of interests (ROI) was first extracted by a localization network then input to a segmentation U-Net. However, this method still requires to work on intensively downsampled data. Methods that preserve the original data resolution have to use a relatively shallow U-Net architecture. Inspired by the network structure of [11], we propose a two-stage DCNN framework which consists of two concatenated U-Net-like networks. A new multi-stage learning pipeline was adopted to the training process. This framework also segment 3D full-resolution MR and CT data within a ROI that is dynamically extracted, but the original resolution of the image is kept. Our method outperformed the well-trained 3D U-Net in our experiments.

2 Method

Architecture. As shown in Fig. 1 the model consists of two concatenated modified U-Nets. Each basic U-Net block has two convolutional layers, followed by a nonlinear activation and a pooling layer. Both the encoding and the decoding paths of the two U-Nets have 4 basic U-Net convolution blocks. The final output of each network is produced by a softmax classification layer. The first network (*Net1* in Fig. 1) aim to produce a coarse segmentation on down-sampled full 3D volume. We use dilated $5 \times 5 \times 5$ convolutional kernel with zero-padding which preserves shapes of feature maps. In the nth block of the encoding path, the dilation rate of the convolutional kernel is $2n$. This pattern is reversed in the expansive path. Each convolutional layer is followed by a rectified linear unit (ReLU), and a dropout layer with a 0.2 dropout rate is attached to each U-Net block. In the test phase, a dynamic-tile layer is introduced between *Net1* and

Net2 to crop out a ROI from both the input and output volume of *Net1*. This layer is removed when performing end-to-end training. The output of *Net1* is resampled to the original resolution before input into *Net2*. *Net2* is a 3D-2D U-Net segmenting the central slice of a subvolume. The structure of *Net2* is inspired by the deep Super-Resolution Convolutional Neural Network (SRCNN) [12] with skip connections [13]. The input of this network is a two-channel 4D volume consist of the output of *Net1* and the original data. The convolutional kernel size in the encoding path is $3 \times 3 \times 3$, and $5 \times 5 \times 5$ in the decoding path. The size of the 3D pooling kernels in the contracting path of *Net2* is $2 \times 2 \times 1$ so that the number of axial slices is preserved. A 3D-2D slice-wise convolution block with $1 \times 1 \times (K-1)$ convolutional kernels is introduced before the decoding path, where K is the number of neighboring slices used to label one single axial slice. No zero-padding is used to ensure that every K input slices will generate only one axial feature map. Furthermore, K should always be an odd number to prevent generating labels for interpolated slices. The layers in the decoding path of *Net2* perform 2D convolutions and pooling.

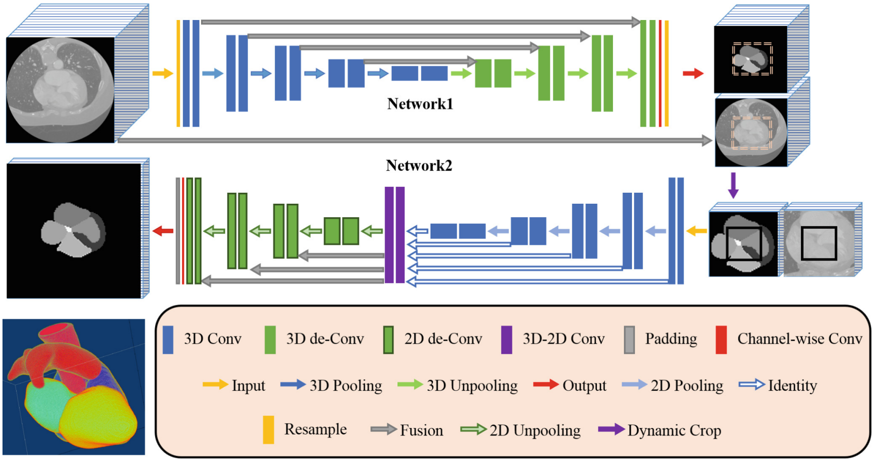

Fig. 1. The concatenated U-Net architecture proposed in this work. The nonlinear activation and pooling layer within each U-Net block are not shown for demonstration purpose.

Training Pipeline and Losses. The duo-U-Net framework can be trained either separately or end-to-end. We combined both approaches into a four-step training procedure. At the beginning, $Net1$ is pre-trained for initial localization of the object. Then the whole framework is trained with different combinations of $Net1$ and $Net2$ loss functions for quick convergence. Details of training batches and loss functions used in each step are shown in Table 1. A commonly used similarity metric for single-class segmentation is soft Dice score. Let $p^i_{n,c}$ denote the probability that a voxel belongs to class $c, c \in \{0, \cdots, C\}$ (background is defined by $c = 0$), given by the softmax layer of $Neti$, and $t_{n,c} \in \{0, 1\}$ represent the ground truth one-hot label. The soft Dice score can be defined as

Table 1. Purposes and loss functions of each step in the training process

Step	Input	Purpose	Loss
1	full volumetric data	foreground localization	$\mathcal{L}_{ROI}^1$
2	partial volumetric data	coarse multi-class segmentation	$\mathcal{L}_{ROI}^1 + \mathcal{L}^1$
3	partial volumetric data	coarse+fine segmentation	$\mathcal{L}^1 + \mathcal{L}^2$
4	stack of full axial slices	fine multi-class segmentation	$\mathcal{L}^2$

$\mathcal{S}_c^i = \frac{2\sum_n^{N_c} t_{n,c}p_{n,c}^i+\epsilon}{\sum_n^{N_c}\left(t_{n,c}+p_{n,c}^i\right)+\epsilon}$, where N_c is number of voxels labeled as class c and ϵ is a smooth factor. To perform multi-class segmentation, we just define our loss function using weighted Dice scores weighted by voxel counts for simplicity:

$$\mathcal{L}^i = 1 - \sum\nolimits_c^C \frac{S_c^i}{N_c}. \tag{1}$$

But nothing stops using a more sophisticated loss functions as shown in [14]. In different training steps, losses of the two nets are combined for different purposes. In the first step, *Net1* is trained with full volumetric data to roughly localize the foreground, or a soft ROI, which is the segmented object. Other contents in the data are considered as background. Parameters of $Net2$ is frozen after initialized. The input data is firstly resampled to very coarse resolution, for example, $3 \times 3 \times 3\,\mathrm{mm}^3$ as used in our experiments. To encourage localization of foreground, the loss function is defined by combining the foreground Dice score with the multi class Dice score. The foreground Dice score $\mathcal{L}_{ROI}^1$ computed from $Net1$ output is defined as $\mathcal{S}_{ROI}^1 = \frac{2\sum_n^{N_0}(1-t_{n,0})\left(1-p_{n,0}^1\right)+\epsilon}{\sum_n^{N_0}\left(2-t_{n,0}-p_{n,0}^1\right)+\epsilon}$, where N_0 is the number of the background points as the background is defined as class 0. The corresponded foreground loss is:

$$\mathcal{L}_{ROI}^1 = 1 - \frac{S_{ROI}^1}{N_0}. \tag{2}$$

We use reversed label to calculate foreground score rather than the Dice score of the background class ($c = 0$) to reduce the imbalance introduced by large background. $Net1$ is trained to minimize the loss $\mathcal{L}_{ROI}^1$ for locating the foreground of the object. After pre-training with $\mathcal{L}_{ROI}^1$, $\mathcal{L}^1 + \mathcal{L}_{ROI}^1$ is used as the loss for coarse multi-class segmentation in the second step, where $\mathcal{L}^1$ is the *Net1* loss defined by Eq. 1. In this step, $Net1$ is trained using subvolumes of the data. Dimensions of the data are varied in different training batches as an augmentation strategy. In the third step, the whole framework (both $Net1$ and $Net2$) is trained end-to-end with the loss $\mathcal{L}^1 + \mathcal{L}^2$ to evolve both coarse 3D segmentation and the fine-level axial 2D segmentation. In the final step, inputs of the framework are subvolumes, each consists of K axial slices. The output of $Net2$ is the segmentation of the $\frac{K+1}{2}$th slice of a input subvolume. The parameters of $Net1$ are frozen, and $Net2$ is finetuned using the loss $\mathcal{L}^2$.

Implementation Details. Because the framework is mostly trained with subvolumes of the 3D data except the first step, we use a hierarchical sampling strategy similar with [15]. Each batches are generated from a small number of data. As the ground-truth labels are imbalanced, we sample the label of the central voxel in the subvolume from a uniform distribution. Once the label of the central voxel is fixed, the subvolume is generated by randomly picking its centre from all voxels labeled as the selected class. In this way, the probabilities that the central voxel belongs to any of the classes should be $\frac{1}{C+1}$. For optimization, we use Adam optimizer with initial learning rate 0.0001. We use only 1 full volume in each batch in the first step. In step 2 and 3, the size of each subvolume is randomly selected from $\{64, 128, 256\}$. In the final training step, we set $K = 9$, which means $Net2$ take a subvolume that contains 9 axial slices and predict the labels for the 5th slice of this subvolume. Tensorflow code was adopted on Microsoft Azure virtual machine with one NVidia Tesla K40 GPU. Data augmentation includes random rotation, translation, sheering, scaling, flipping and elastic deformations.

3 Experiments and Data

The MICCAI 2017 Multi-Modality Whole Heart Segmentation (MM-WHS) challenge recently benchmarks existing whole heart segmentation algorithms. For training, the challenge provides 40 volumes (20 cardiac CT and 20 cardiac MR). The data were acquired with different scanners, which leads to varying voxel sizes, resolutions and imaging qualities. An extra 80 testing images are available from the challenge for a final test. In this dataset, anatomical structures which are manually delineated include, the left ventricle blood cavity (LV), the myocardium of the left ventricle (Myo), the right ventricle blood cavity (RV), the left atrium blood cavity (LA), the right atrium blood cavity (RA), the ascending aorta (AA) and the pulmonary artery (PA).

One may argue that the proposed framework can be trained directly using the final training step. To demonstrate effects of the proposed training procedure, we trained our model by omitting one of the first three steps, and visually assessed the segmentation results which can be found in the next section. The first three steps do not necessarily need to converge as long as the final step converges. In this step-wise experiment, all results were obtained with 200 epochs in each step, and each epoch includes 16 iterations of backpropogation. The whole training process contains 12800 iterations in total.

Besides qualitatively evaluating the visualized segmentation, for each modality, we use 15 volumes for 3-fold cross-validation training, and 5 volumes for validation. To compare our framework with state-of-the-art U-Net-based models, we trained two 3D U-Nets for each modality which predict on data resampled to resolution of $2 \times 2 \times 2\,\mathrm{mm}^3$. Then the output volumes are resampled to the original resolution using 2nd order BSpline interpolation. Intensities of all images are rescaled to $[-1, 1]$ with no further preprocessing. Three metrics were used to assess segmentation quality for each class: binary Dice score (Dice), binary Jaccard index (Jaccard).

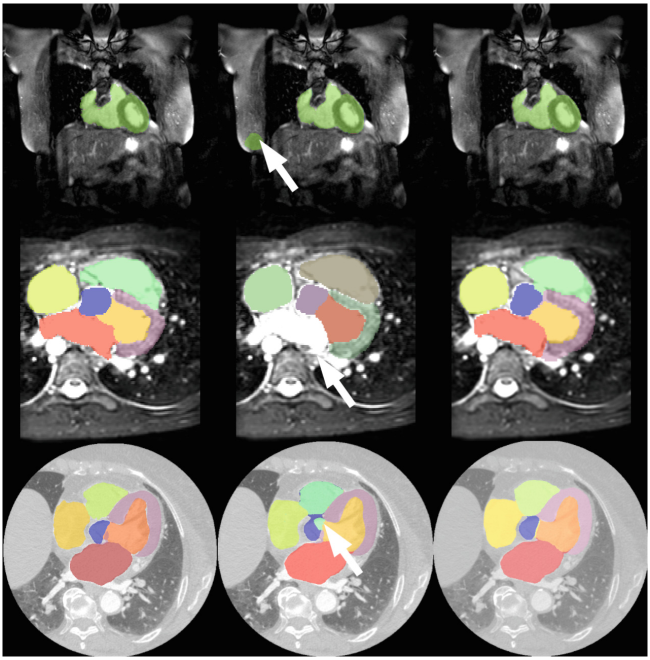

Fig. 2. Visualization of segmentation results overlapped with the input images: Ground-truth segmentations are shown on the left; the middle column shows the results obtained by omitting the first, second and third training step (from top to bottom); on the right is the results obtained with the proposed training process. White arrows point to misclassifications caused by omitting a training step.

4 Results

Examples of visualized segmentation results are shown in Fig. 2. Omitting the foreground localization step in the first training step may lead to misclassification of the background voxels, as shown in the top row of Fig. 2. The middle row shows that without the coarse segmentation (second step) the model failed to label left atrium, and produced inhomogeneous segmentation for aorta when skipping the joint training of $Net1$ and $Net2$ (step three).

Tables 2 and 3 show the binary Dice and Jaccard scores for all assessed structures obtained by $Net1$ and $Net2$ from the proposed framework, compared to individually trained U-Nets. $Net2$ produced highest Dice and Jaccard scores for all segmented structures in CT data, $Net1$ gave better results than the individual U-Net trained on the same downsampled data. As the MR data have relatively lower resolution, the volume size changed less after resampling. $Net2$ still produced better segmentation accuracy except for RV and AA. $Net1$ gave better segmentations for 4 out of all 7 classes.

The quantitative results of $Net2$ obtained using the test dataset of MM-WHS competition are shown in Table 4. The proposed framework achieved obviously

Table 2. Comparison of CT segmentation results obtained by 3D U-Net, and our proposed $Net1$ and $Net2$.

	Metrics	LV	Myo	RV	LA	RA	AA	PA
N3D U-Net	Dice	0.6451	0.8301	0.7873	0.7768	0.6784	0.8306	0.7123
	Jaccard	0.4889	0.7126	0.6572	0.6397	0.5217	0.7143	0.5560
Net1	Dice	0.6774	0.8107	0.8136	0.8118	0.7997	0.8889	0.8086
	Jaccard	0.5399	0.6979	0.6977	0.6908	0.6717	0.8030	0.6802
Net1+Net2	Dice	**0.8374**	**0.8588**	**0.8600**	**0.8613**	**0.8620**	**0.9176**	**0.8846**
	Jaccard	**0.7823**	**0.8210**	**0.8233**	**0.8256**	**0.8268**	**0.8534**	**0.7959**

Table 3. Comparison of MR segmentation results obtained by 3D U-Net, and our proposed $Net1$ and $Net2$.

	Metrics	LV	Myo	RV	LA	RA	AA	PA
N3D U-Net	Dice	0.8296	0.9141	**0.9173**	0.8946	0.8792	**0.9202**	0.8847
	Jaccard	0.7106	0.8419	**0.8479**	0.8107	0.7894	**0.8526**	0.7938
Net1	Dice	0.8811	0.9367	0.9131	0.9334	0.8572	0.8750	0.9204
	Jaccard	0.7877	0.8813	0.8430	0.8757	0.7694	0.7833	0.8528
Net1+Net2	Dice	**0.8813**	**0.9377**	0.9125	**0.9338**	**0.9220**	0.8758	**0.9210**
	Jaccard	**0.7879**	**0.8829**	0.8422	**0.8764**	**0.8568**	0.7847	**0.8539**

Table 4. Quantitative results obtained using MM-WHS test data, evaluation metrics include: average of Dice score (Dice), average of Jaccard score (Jaccard), and Average surface distance (ASD).

Modality	Metrics	LV	Myo	RV	LA	RA	AA	PA	WH
CT	Dice	0.7995	0.7293	0.7857	0.9044	0.7936	0.8735	0.6482	0.8060
	Jaccard	0.6999	0.6091	0.6841	0.8285	0.6906	0.8113	0.5169	0.6970
	ASD	4.4067	5.4854	4.8816	1.3978	4.1707	3.7898	6.0041	4.1971
MR	Dice	0.8632	0.7443	0.8485	0.8524	0.8396	0.8236	0.7876	0.8323
	Jaccard	0.7693	0.6049	0.7469	0.7483	0.7404	0.7095	0.6657	0.7201
	ASD	1.9916	2.3106	1.8925	1.7081	2.7566	4.2610	2.9296	2.4718

higher accuracy for LA and AA. This is a sign of premature termination of training, although the framework still obtained much better results than the baseline algorithm of the competition. For MR data, the average Dice score of $Net2$ is 0.8323, which is comparable to the winner of the competition. However, our model was only trained 12800 iterations which is only 1/5 of the winner model. Notice that the purpose of this study is to generate the model gave better performance than state-of-the-art U-Net when segmenting high resolution data. This has been shown in the experiment described above.

5 Conclusion and Discussion

In this paper, we described a two-stage U-Net-like framework for multi-class segmentation. Unlike other U-Net based 3D data segmentation DCNN, the proposed method can directly make prediction for data with original resolution due to its SRCNN-inspired architecture. A novel 4-step training procedure were applied to the framework. Validated using data from MM-WHS2017 competition, it produced more accurate multi-class segmentation results than state-of-the-art U-Net. With much less training iterations and without any further post-processing, our method achieved segmentation accuracies comparable to the winner of MM-WHS2017 competition.

References

1. Pace, D.F., Dalca, A.V., Geva, T., Powell, A.J., Moghari, M.H., Golland, P.: Interactive whole-heart segmentation in congenital heart disease. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 80–88. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_10
2. Arrieta, C., Uribe, S., Sing-Long, C., Hurtado, D., Andia, M., Irarrazaval, P., Tejos, C.: Simultaneous left and right ventricle segmentation using topology preserving level sets. Biomed. Sig. Process. Control **33**, 88–95 (2017)
3. Gonzalez-Mora, J., De la Torre, F., Murthi, R., Guil, N., Zapata, E.L.: Bilinear active appearance models. In: IEEE 11th International Conference on Computer Vision, ICCV 2007, pp. 1–8. IEEE (2007)
4. Marsland, S., Twining, C.J., Taylor, C.J.: Groupwise non-rigid registration using polyharmonic clamped-plate splines. In: Ellis, R.E., Peters, T.M. (eds.) MICCAI 2003. LNCS, vol. 2879, pp. 771–779. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-39903-2_94
5. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
6. Wolterink, J.M., Leiner, T., Viergever, M.A., Išgum, I.: Dilated convolutional neural networks for cardiovascular MR segmentation in congenital heart disease. In: Zuluaga, M.A., Bhatia, K., Kainz, B., Moghari, M.H., Pace, D.F. (eds.) RAMBO/HVSMR -2016. LNCS, vol. 10129, pp. 95–102. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-52280-7_9
7. Moeskops, P., et al.: Deep learning for multi-task medical image segmentation in multiple modalities. In: Ourselin, S., Joskowicz, L., Sabuncu, M.R., Unal, G., Wells, W. (eds.) MICCAI 2016. LNCS, vol. 9901, pp. 478–486. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46723-8_55
8. Mortazi, A., Burt, J., Bagci, U.: Multi-planar deep segmentation networks for cardiac substructures from MRI and CT. arXiv preprint arXiv:1708.00983 (2017)
9. Roth, H.R., et al.: Hierarchical 3D fully convolutional networks for multi-organ segmentation. arXiv preprint arXiv:1704.06382 (2017)
10. Yu, L.: Automatic 3D cardiovascular MR segmentation with densely-connected volumetric convnets. In: Descoteaux, M., Maier-Hein, L., Franz, A., Jannin, P., Collins, D.L., Duchesne, S. (eds.) MICCAI 2017. LNCS, vol. 10434, pp. 287–295. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66185-8_33

11. Payer, C., Štern, D., Bischof, H., Urschler, M.: Multi-label whole heart segmentation using CNNs and anatomical label configurations. In: Pop, M., et al. (eds.) STACOM 2017. LNCS, vol. 10663, pp. 190–198. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-75541-0_20
12. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. IEEE Trans. Pattern Anal. Mach. Intell. **38**(2), 295–307 (2016)
13. Kim, J., Kwon Lee, J., Mu Lee, K.: Deeply-recursive convolutional network for image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1637–1645 (2016)
14. Berger, L., Hyde, E., Cardoso, J., Ourselin, S.: An adaptive sampling scheme to efficiently train fully convolutional networks for semantic segmentation. arXiv preprint arXiv:1709.02764 (2017)
15. Girshick, R.: Fast R-CNN. arXiv preprint arXiv:1504.08083 (2015)