Automatic 3D Atrial Segmentation from GE-MRIs using Volumetric Fully Convolutional Networks

Qing Xia[1], Yuxin Yao[2], Zhiqiang Hu[3], and Aimin Hao[1]

[1] State Key Lab of Virtual Reality Technology and Systems, Beihang University
{xiaqing, ham}@buaa.edu.cn
[2] School of Automation Science and Electrical Engineering, Beihang University
yyxdawang78@buaa.edu.cn
[3] School of Electronics Engineering and Computer Science, Peking University
huzq@pku.edu.cn

Abstract. In this paper, we propose an approach for automatic 3D atrial segmentation from Gadolinium-enhanced MRIs based on volumetric fully convolutional networks. The entire framework consists of two networks, the first network is to roughly locate the atrial center based on a low-resolution down-sampled version of the input and cut out a fixed size area that covers the atrial cavity, leaving out other pixels irrelevant to reduce memory consumption, and the second network is to precisely segment atrial cavity from the cropped sub-regions obtained from last step. Both two networks are trained end-to-end from scratch using 2018 Atrial Segmentation Challenge dataset which contains 100 GE-MRIs, and our method achieves satisfactory segmentation accuracy, up to 0.923 in Dice Similarity Coefficient score.

Keywords: Automatic Atrial Segmentation · Fully Convolutional Networks · Gadolinium-enhanced-MRI

1 Introduction

Atrial fibrillation (AF) is one of the most common type of cardiac arrhythmia, which greatly affects human health throughout the world [11]. But it is still challenging to develop successful treatment because of the gaps in understanding the mechanisms of AF [4]. Magnetic resonance images (MRI) can produce pictures of different structures within the heart, and gadolinium contrast agencies are usually used to improve the clarity of these images. These Gadolinium-enhanced MRIs (GE-MRIs) are widely used to study the extent of fibrosis across the atria [9] and recent studies on human atria imaged with GE-MRIs have suggested the atrial structure may hold the key to understanding and reversing AF [4, 15]. However, due to the low contrast between the atrial tissue and surrounding background, it is very challenging to directly segment the atrial chambers from GE-MRIs. Most of the existing atrial structural segmentation methods are based on hand-crafted shape descriptors or deformable models on

non-enhanced MRIs [14], which can not be directly applied on GE-MRIs because of the low contrast. As for GE-MRIs, current atrial segmentation approaches are still labor-intensive, error/bias-prone, which are obviously not suitable for practical and clinical medical use.

In the past decade, deep learning techniques, in particular Convolutional Neural Networks (CNNs), have achieved great progress in various computer vision tasks, and rapidly become a methodology of choice for analyzing medical images [7]. Ciresan et al. [3] firstly introduced CNNs to medical image segmentation by predicting a pixel's label based on the raw pixel values in a square window centered it. But this method is quite slow because the network must run separately for every pixel within every single image and there is a lot of redundancy due to overlapping windows actually. Later on, Ronneberger et al. proposed U-Net [13], which consists of a contracting path to capture context and a symmetric expanding path that enables precise localization and can be trained end-to-end from very few images built upon the famous Fully Convolutional Network (FCN) [8]. Then, Çiçek et al. [2] replaced the convolution operations in 2D U-Net with 3D counterparts and proposed 3D U-Net for volumetric segmentation. Furthermore, Milletari et al. [10] proposed V-Net, wherein they introduce a novel loss function based on Dice coefficient and learn a residual function inspired by [6] which ensures convergence in less training time and achieves good segmentation accuracy.

In this paper, we develop an automatic 3D atrial segmentation framework using volumetric fully convolutional networks for 2018 Atrial Segmentation Challenge. The overall pipeline of our method is shown in Fig. 1, it consists of two main stages: 1) in the first stage, we use a segmentation based localization strategy to estimate a fixed size target region that covers the whole atria, and leave out pixels outside this region to cut down memory consumption; 2) in the second stage, we train a fine segmentation network based on the cropped target region obtained in the first stage, and transform the predicted masks in target region to the original size volume. The segmentation networks in these two stages are both adapted from V-Net, which can be trained end-to-end and used to segment the atrial cavity fully-automatically.

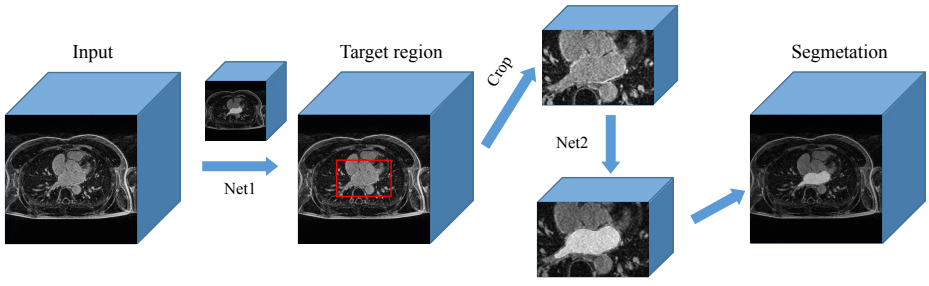

Fig. 1. The overall pipeline of our automatic 3D atrial segmentation framework.

2 Method

2.1 Dataset and Preprocessing

Our framework is trained and tested using 2018 Atrial Segmentation Challenge dataset, which contains 100 3D GE-MRIs for training. Each 3D MRI data was acquired using a clinical whole-body MRI scanner and contained raw MRI scan and the corresponding ground truth labels for the left atrial (LA) cavity. The original resolution of these data is 0.625*0.625*0.625 mm^3, 47 of them are with 576*576*88 voxels and 53 of them are with 640*640*88 voxels, and it is very hard to apply neural networks to directly segment from such high-resolution volumes on a normal personal computer due to memory restriction. Actually, the LA cavity, even the whole heart, takes only a very small fraction of the entire MRI volume and other places in the volume are irrelevant tissues or even nothing, and such extreme class imbalance between the foreground atrial cavity and background also makes the segmentation task hard. So we divide the segmentation into two steps, the first is to locate the atria in the beginning and the second is to segment the cavity from a much smaller cropped sub-volume, which can be used to train a network on a normal PC. To make the input data uniformly sized and suitable for V-Net architecture, we firstly crop and zero-pad all volumes to with size 576*576*96 and the predictions are transformed to the original size 576*576*88 or 640*640*88 in a post-precessing step. Then we use a 3D version of contrast limited adaptive histogram localization (CLAHE) [12] to enhance the contrast of GE-MRIs, and finally apply sample-wise normalization wherein each volume is subtracted with the mean value of intensity and divided by the deviation of intensity.

2.2 Network Architecture

The segmentation network involved in our framework is adapted from V-Net [10] as illustrated in Fig. 2. It is a fully convolutional neural network, in which convolution operations are used to both extract features in different scales from the data and reduce the resolution by applying appropriate stride. The left part of the network is an encoding path following a typical architecture of a standard convolutional network, which captures the context information in a local-to-global sense, and the right part decodes the signal to its original size and output two volumes indicating the probability to be foreground and background respectively.

The left side of the network is divided in a few stages that operate at different resolutions, each stage consists of one or two convolutional layers, and learns a residual function, that is, the input of each stage is added to the output of the last convolutional layer of that stage. The convolutions performed in each layer use volumetric kernels with size 5*5*5, and the pooling is achieved by convolution operation with size 2*2*2 and stride 2. Moreover, the number of feature channels doubles at each stage of the encoding path while the resolutions halves. In the end of each layer, batch normalization and PRelu non linearities are used.

The right side of the network is a symmetric counterparts of the left that extracts features and expands the spatial support to output a two channel volumetric segmentation. Similar to the left part of the network, each stage of the right part contains one or two convolutional layers, and also learns a residual function. The convolutions performed in each layer also use volumetric kernels with size 5*5*5, and the up-pooling is achieved by de-convolution operation with size 2*2*2 and stride 2. The features extracted from the left part of the network are forwarded to the corresponding stage of the right part, which is shown as horizontal connections in Fig. 2. The same as those in the left part, batch normalization and PRelu non linearities are used in the end of each layer.

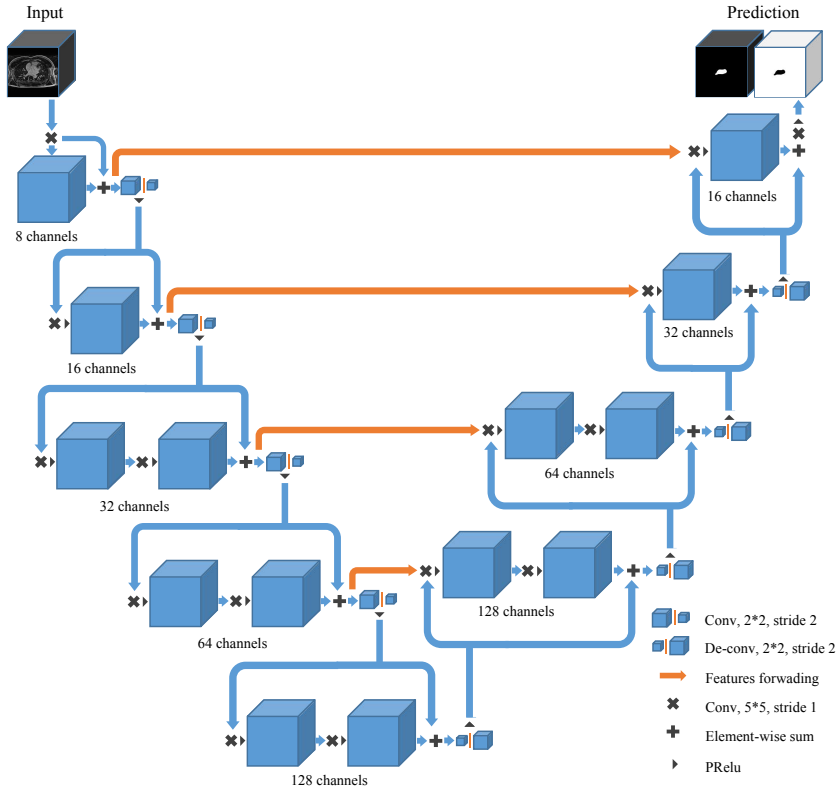

Fig. 2. The architecture of our segmentation network adapted from V-Net [10].

2.3 Loss Function

The segmentation network predict two volumes of the same size of the input, which are computed after a voxel-wise softmax activation in the final layer and

indicate the probability of each voxel to be foreground or background. In segmentation tasks, our aim is to train a network whose foreground prediction is as similar as the given ground truth mask. As the left atrial cavity only takes a small fraction of the volume, we adopt the dice coefficient to define the loss function that to be minimize. The dice coefficient is used to measure the similarity between two given binary data, and be expressed as

$$Dice = \frac{2|\mathbf{a} \cdot \mathbf{b}|}{|\mathbf{a}|^2 + |\mathbf{b}|^2}, \tag{1}$$

where $\mathbf{a}, \mathbf{b}$ are two binary vectors. If $\mathbf{a} = \mathbf{b}$, $Dice = 1$, and if $\mathbf{a_i} \neq \mathbf{b_i}$ for all i, $Dice = 0$. In our implementation, we use the foreground prediction (probability) and the given ground truth as $\mathbf{a}, \mathbf{b}$ respectively to compute the loss of the network, which is simply defined as

$$Loss = 1 - Dice. \tag{2}$$

This formulation do not need to assign weights to samples of different classes to establish the right balance between foreground and background voxels and is very easy to understand and implement. We also compared dice loss with traditional cross-entropy loss, the results can be found in Fig. 3, wherein we can see the segmentation networks converge faster and reach higher dice coefficients when using dice loss function.

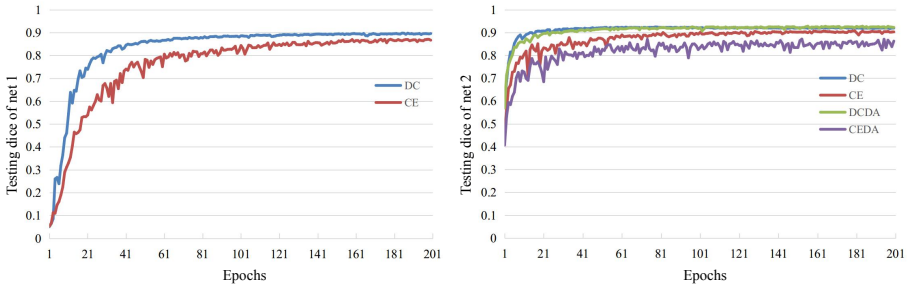

Fig. 3. The dice coefficients on validation data after training different epochs of the two segmentation networks in our framework using dice loss (DC) and cross-entropy loss (CE). Moreover, data augmentation (DCDA & CEDA) is applied in the second network when using both loss functions.

2.4 Training

As we mentioned in the beginning, our framework consists of two main stages, the first is to locate the target based on coarse segmentation, and the second is to segment the left atria cavity from the cropped target region. So, we need to train

two segmentation networks. For the first network, we firstly down-sample the input with sampling rate 0.25*0.25*0.5 and reduce the resolution from 576*576*96 to 144*144*48, which makes the network consumes much lower memory and can be trained on a normal personal computer. Here choosing sampling rate 0.5 in Z-axis instead of 0.25 is simply to avoid extreme narrow feature maps produced by pooling. Then we feed the input into the network, the weights are initialized using He initialization [5] and updated using Adam algorithm with a fixed learning rate 0.001. We choose 80 out of the 100 data as training data and the rest 20 as testing data, training is completed after 200 epochs and the model with best dice score is saved, and we use mini-batch of size 4 in the first network. For the second network, we firstly compute the barycenter of the given ground truth mask, and crop a region of size 240*160*96 centered with the barycenter from the original data. We have calculated the bounding box of all given masks, and found that the maximum widths along x, y, z axis are 209, 128, 73 voxels respectively, so a region of size 240*160*96 is big enough to cover the whole cavity. And then we feed the cropped input into the network and train it in the same way as that in the first stage except that batch size is 1 due to memory restriction. To further improve the generalization ability and segmentation accuracy of our framework, we also apply data augmentation in the second network. Before feeding the cropped volume into the network, we randomly choose to slightly translate, scale, rotate, or flip the input data in 3D, and 3D elastic deformation is also used to generate shape diversity.

2.5 Testing

In the testing phase, a previously unseen MRI volume is firstly down-sampled to 144*144*48, and fed into the first network. The network will output the probability map for both background and foreground, we apply a simple binary test on these two volumetric map where voxels are assign to be foreground or background according which corresponding probability is higher, and this binary mask is used to locate the target region. We compute the barycenter of the predicted mask, crop a region of size 240*160*96 centered with this barycenter and then feed it into the second network. The second network also output the probability map and we can compute a binary mask inside the target region and map it back to the original size volume, which is the final left atrial cavity segmentation result, as shown in Fig. 1.

3 Result

We implemented our framework using PyTorch [1] with cuDNN, and ran all experiments on a personal computer with 8 GB of memory, Intel Core i7 6700K CPU @ 4.00 Ghz, and a Nvidia GTX 1060 6G GPU. We tested our framework on the 100 GE-MRIs provided by 2018 Atrial Segmentation Challenge, and conducted a 5-fold cross validation, which leaves 80 volumes for training and the rest 20 for validation.

We firstly compared between using dice loss and traditional cross-entropy loss in our framework. The detailed statistics are listed in Table 1, each row contains the segmentation dice coefficients using dice loss and cross-entropy loss of the first, second network and the entire framework in each fold and the average is shown in the bottom. The segmentation accuracy are better when using dice loss than cross-entropy loss in both two networks, and the data augmentation applied in the second network also improves the performance when using dice loss. However, when using cross-entropy loss, data augmentation leads side effects on the accuracy instead. One possible reason is that operations for data augmentation, such as scale and deformation, greatly increase the variation of atrial volume sizes which makes the problem of class-imbalance between foreground and background more serious. For example, the size proportion between atrial cavity and the background varies from sample to sample, but the class weights used in cross-entropy are usually computed as fixed averages, and dice loss do not suffer from class-imbalance problem at all. This situation can also be found in plots of Fig. 3, where the dice coefficients oscillates when applying data augmentation with cross-entropy loss while the other plots are smoother and stabler in contrast. So when using cross-entropy in our framework, we do not apply data augmentation in the second network. Moreover, we can also see from the statistics that the segmentation accuracy of the entire framework is actually the same with the second network, that means the accuracy of of the first step do not affect the final segmentation accuracy of the entire framework as long as it gives relatively right target location and the cropped sub-region that covers the entire left atrial cavity. Thus, we can improve the final segmentation accuracy simply by further improving the second network's performance, this is also why we apply data augmentation only for the second network when using dice loss.

The first network takes about 4.7 hours to train and consume 4.2 GB GPU memory in average (input size is 144*144*48 and bath size is 4), and the second network takes about 13.6 hours to train and consume about 4GB GPU memory in average (input size is 240*160*96 and batch size is 1). At test time, our framework can generate the entire segmentation output within 2s using about 2.6 GB GPU memory. This show our approach's great potential for practical clinical use, because of its simplicity, effectiveness, high accuracy and efficiency. 5 selected atrial segmentation results are listed in Fig. 4 comparing to the given ground truth, the first 4 are those with top dice coefficients and the last one is the worst case among all MRIs. For those MRIs with relative high equality, our frameworks works pretty well, but when facing with MRIs that are unclear and blurry, our method still struggles for higher segmentation accuracy.

4 Conclusion

This paper detailed a simple but effective approach for automatic 3D atrial segmentation from GE-MRIs, which consists of two volumetric fully convolutional networks adapted from V-Net. The first network is used to coarsely segment the

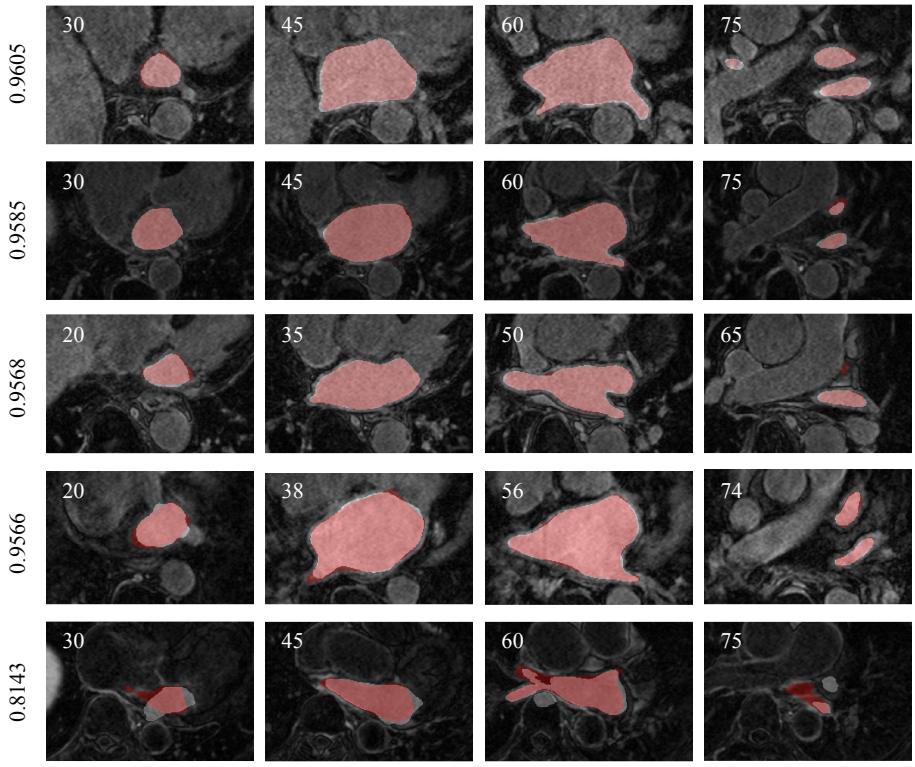

Fig. 4. Segmentation results of 5 patients comparing to given ground truth. The ground truths are given in red color and the predictions are colored in gray. The dice coefficients of all predictions are shown in the left most column and the slice number (from axial view) of each picture is shown in the upper left corner of itself.

Table 1. Segmentation accuracy. From left to right: barycenter estimation error (BCE1) (in voxels), segmentation dice coefficients using dice loss (DC1) and cross-entropy loss (CE1) of the first network, dice coefficients of the second network using dice loss and cross-entropy without (DC2 & CE2) and with (DCDA2 & CEDA2) data augmentation, and the entire framework on validation data using dice loss (DC) and cross-entropy loss (CE).

Fold	BCE1	DC1	CE1	DC2	DCDA2	CE2	CEDA2	DC	CE
1	0.69,0.21,0.51	0.885	0.871	0.917	0.923	0.904	0.872	0.923	0.904
2	0.68,0.44,0.63	0.864	0.834	0.902	0.909	0.888	0.869	0.909	0.888
3	0.45,0.34,0.52	0.883	0.877	0.913	0.924	0.917	0.884	0.924	0.917
4	0.36,0.27,0.61	0.889	0.870	0.906	0.932	0.911	0.879	0.932	0.911
5	0.52,0.27,0.42	0.894	0.871	0.920	0.929	0.908	0.871	0.929	0.908
Avg	0.54,0.31,0.54	0.884	0.865	0.912	0.923	0.906	0.875	**0.923**	0.906

atria from a low-resolution version of the input and estimate the location of the atrial cavity. The second is used to further precisely segment the atria from the cropped sub-region that covers the whole atria. This multi-resolution solution has low memory costs, allowing the network to be trained on a normal personal computer, and the high efficiency make it very easy to apply our segmentation method to clinical use, for example, to reconstruct the structure of human atria and to help researchers develop effective treatments for atrial fibrillation.

References

1. Pytorch. http://pytorch.org/
2. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3d u-net: learning dense volumetric segmentation from sparse annotation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 424–432. Springer (2016)
3. Ciresan, D., Giusti, A., Gambardella, L.M., Schmidhuber, J.: Deep neural networks segment neuronal membranes in electron microscopy images. In: Advances in neural information processing systems. pp. 2843–2851 (2012)
4. Hansen, B.J., Zhao, J., Csepe, T.A., Moore, B.T., Li, N., Jayne, L.A., Kalyanasundaram, A., Lim, P., Bratasz, A., Powell, K.A., et al.: Atrial fibrillation driven by micro-anatomic intramural re-entry revealed by simultaneous sub-epicardial and sub-endocardial optical mapping in explanted human hearts. European heart journal **36**(35), 2390–2401 (2015)
5. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision. pp. 1026–1034 (2015)
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
7. Litjens, G., Kooi, T., Bejnordi, B.E., Setio, A.A.A., Ciompi, F., Ghafoorian, M., van der Laak, J.A., Van Ginneken, B., Sánchez, C.I.: A survey on deep learning in medical image analysis. Medical image analysis **42**, 60–88 (2017)
8. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015)
9. McGann, C., Akoum, N., Patel, A., Kholmovski, E., Revelo, P., Damal, K., Wilson, B., Cates, J., Harrison, A., Ranjan, R., et al.: Atrial fibrillation ablation outcome is predicted by left atrial remodeling on mri. Circulation: Arrhythmia and Electrophysiology **7**(1), 23–30 (2014)
10. Milletari, F., Navab, N., Ahmadi, S.A.: V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: 3D Vision (3DV), 2016 Fourth International Conference on. pp. 565–571. IEEE (2016)
11. Nishida, K., Nattel, S.: Atrial fibrillation compendium: historical context and detailed translational perspective on an important clinical problem. Circulation research **114**(9), 1447–1452 (2014)
12. Pizer, S.M., Johnston, R.E., Ericksen, J.P., Yankaskas, B.C., Muller, K.E.: Contrast-limited adaptive histogram equalization: speed and effectiveness. In: Visualization in Biomedical Computing, 1990., Proceedings of the First Conference on. pp. 337–345. IEEE (1990)

13. 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. pp. 234–241. Springer (2015)
14. Tobon-Gomez, C., Geers, A.J., Peters, J., Weese, J., Pinto, K., Karim, R., Ammar, M., Daoudi, A., Margeta, J., Sandoval, Z., et al.: Benchmark for algorithms segmenting the left atrium from 3d ct and mri datasets. IEEE transactions on medical imaging **34**(7), 1460–1473 (2015)
15. Zhao, J., Hansen, B.J., Wang, Y., Csepe, T.A., Sul, L.V., Tang, A., Yuan, Y., Li, N., Bratasz, A., Powell, K.A., et al.: Three-dimensional integrated functional, structural, and computational mapping to define the structural fingerprints of heart-specific atrial fibrillation drivers in human heart ex vivo. Journal of the American Heart Association **6**(8), e005922 (2017)