Improving Echocardiography Segmentation by Polar Transformation

Zishun Feng[1(✉)], Joseph A. Sivak[2], and Ashok K. Krishnamurthy[1,3]

[1] Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, USA
fzs@cs.unc.edu

[2] Division of Cardiology, University of North Carolina at Chapel Hill, Chapel Hill, USA

[3] Renaissance Computing Institute, University of North Carolina at Chapel Hill, Chapel Hill, USA

Abstract. Segmentation of echocardiograms plays an essential role in the quantitative analysis of the heart and helps diagnose cardiac diseases. In the recent decade, deep learning-based approaches have significantly improved the performance of echocardiogram segmentation. Most deep learning-based methods assume that the image to be processed is rectangular in shape. However, typically echocardiogram images are formed within a sector of a circle, with a significant region in the overall rectangular image where there is no data, a result of the ultrasound imaging methodology. This large non-imaging region can influence the training of deep neural networks. In this paper, we propose to use polar transformation to help train deep learning algorithms. Using the r-θ transformation, a significant portion of the non-imaging background is removed, allowing the neural network to focus on the heart image. The segmentation model is trained on both x-y and r-θ images. During inference, the predictions from the x-y and r-θ images are combined using max-voting. We verify the efficacy of our method on the CAMUS dataset with a variety of segmentation networks, encoder networks, and loss functions. The experimental results demonstrate the effectiveness and versatility of our proposed method for improving the segmentation results.

Keywords: Echocardiography · Segmentation · Polar transformation · Deep learning

1 Introduction

Echocardiography (echo) is a radiation-free and cost-effective imaging modality. Thus, echo is the first-line imaging technique for diagnosing most cardiac diseases. Accurate segmentation of echo images can significantly help the quantitative measurement of the heart and diagnosis of cardiac diseases. For example, segmentation of the left ventricle at end-systolic and end-diastolic frames can be used to calculate ejection fraction (EF), which is an essential cardiac function metric; segmentation of the myocardium is used to calculate wall thickness, which

O. Camara et al. (Eds.): STACOM 2022, LNCS 13593, pp. 133–142, 2022.
https://doi.org/10.1007/978-3-031-23443-9_13

is widely used in cardiac disease diagnosis, such as left ventricular hypertrophy [1–3]. Due to the ultrasound imaging method, a typical echocardiography image is formed within a circular sector, and there is a significant area of the entire rectangular image for which no data are available.

In recent years, deep learning-based segmentation methods have achieved great success in computer vision and medical imaging. Convolutional neural network (CNN) based models, like ResNet, VGG and UNet, have been widely used for medical images analysis in different modalities, such as CT, MRI, and ultrasound.There have been many different kinds of deep learning based echocardiogram analysis applications: 1) view identification [4,5], 2) chamber segmentation [6–9], and 3) disease and abnormality identification[10–12]. For chamber segmentation of echocardiograms specifically Ouyang et al. [7] chose DeepLabV3 [14] to segment the left ventricle for ejection fraction calculation in apical-2-chamber view echocardiograms; Leclerc et al. [6] used UNet [15] to simultaneously segment the left ventricle, myocardium, and left atrium; Liu et al. [8] designed a pyramid local attention architecture for echocardiograms segmentation; Wu et al. [9] proposed a semi-supervised methods for left ventricle segmentation in echocardiography vidoes. All the above-mentioned echocardiogram applications, including classification and segmentation, did not consider the large non-imaging region in the whole echo image, which can influence the training of the networks. Tan et al. [13] applied polar transformations to segment the left ventricle in MRI images. However, the polar transformation was only applied to a small pre-defined region of interest and not the entire imaging area.

In this paper, to address this problem, we propose to use polar transformation to help the segmentation of echo. First, we remove a significant portion of the non-imaging background using r-θ transformation, which helps the network focus on the heart image area. Second, we train the segmentation model on x-y and r-θ spaces simultaneously to let the model capture information from both spaces. Third, during the testing phase, we use max-voting to combine the predictions from the original (x-y) image and the polar (r-θ) image to make the final prediction.

2 Methodology

Our proposed method is shown in Fig. 1 and contains 3 steps: 1) polar transformation that transforms images from the x-y space to r-θ space in order to reduce the non-imaging area, 2) joint model training on both original and polar images and 3) combining original and polar results when testing.

2.1 Polar Transformation of Imaging Region

The ultrasound imaging method results in echocardiography images being formed in a sector of a circle, with large non-imaging regions in the whole rectangular image. The model training can be influenced by this large non-imaging area. To address this problem, we use polar transformation to enlarge the imaging area.

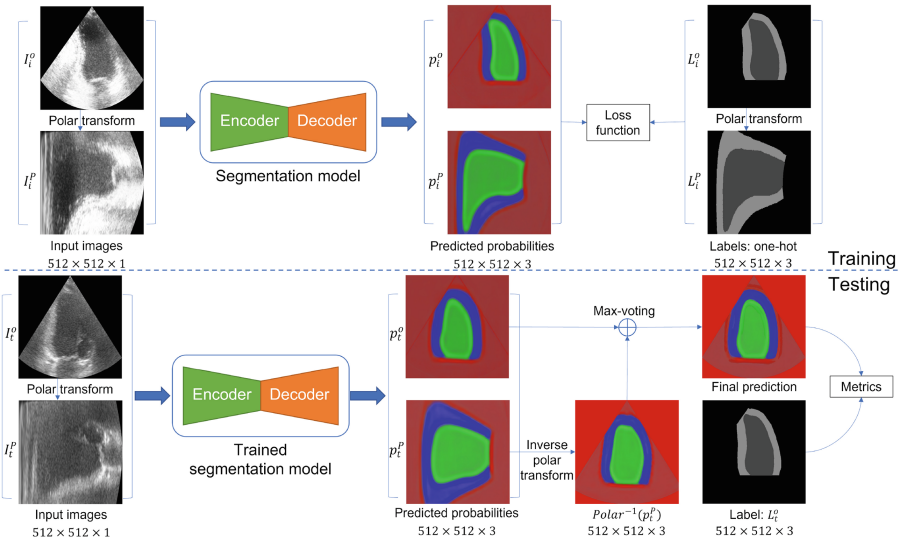

Fig. 1. Training and testing pipeline of the proposed approach. In the training phase, the loss function is calculated in both x-y and r-θ spaces. In the testing phase, performance metrics are only calculated in x-y space.

As shown in Fig. 2, we choose a sector of a circle (orange) in the original echo image, which contains all the imaging areas. We denote original echo image as I^o with size $X \times Y$; and polar image as I^p with size $R \times \Theta$. We use a simple binary segmentation method to identify the center for the sector and the angular extent Θ of the image in polar coordinates. Specifically, we compute the angle between OA and OB, where A and B are the left most point and right most point of the imaging area, and O is the center of the sector. To cover all imaging areas, the radius R is set to be the same as the side of the original image, where $R = Y$. So the value of the polar image I^p at coordinate r-θ can be obtained by:

$$x^* = r\cos(\theta) \tag{1}$$

$$y^* = r\sin(\theta) \tag{2}$$

$$I^p(r,\theta) = I^o(x^*, y^*) \tag{3}$$

where $r \in [0, R], \theta \in [0, \Theta]$. When x^* and y^* are not integer, $I^o(x^*, y^*)$ is calculated by bicubic interpolation.

The polar image label L^p can also be obtained from the original image label L^o by following Eqs. (1)–(2) . However, the label is one-hot coded and the values only contain integers, so we choose nearest interpolation when calculating the polar image labels:

$$L^p(r,\theta) = L^o(round(x^*), round(y^*)) \tag{4}$$

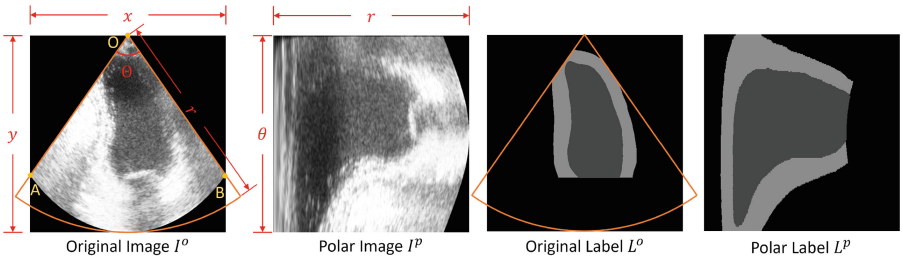

Fig. 2. An example of polar transformation. Before the polar transformation, the region of interest is shown inside the orange lines. (Color figure onlline)

2.2 Joint Training and Testing

Joint Training. The original training set S^o contains N image-label pairs $\{(I_i^o, L_i^o) \mid i \in [1, N]\}$. After the polar transformation, we generate a new polar set $S^p = \{(I_i^p, L_i^p) \mid i \in [1, N]\}$. To let the segmentation networks capture information from both x-y space and r-θ space, we combine these two sets: $S = S^o \cup S^p = \{(I_i^o, I_i^p, L_i^o, L_i^p) \mid i \in [1, N]\}$. So a training batch contains both original and polar images of a sample i in this batch. The pseudo-code is shown below.

Algorithm 1 Joint training and testing

Training:
Input: Original set S^o, network W
Output: Trained network W'
$S^p \leftarrow Polar(S^o)$
while training **do**
$p_i^o \leftarrow W(I_i^o)$
$p_i^p \leftarrow W(I_i^p)$
$loss \leftarrow LossFunc(\{L_i^o, L_i^p\}, \{p_i^o, p_i^p\})$
Update W
end while
return Trained network W'

Testing:
Input:
Test sample I_t^o, L_t^o,
Trained network W'
Output: Output metrics m
$I_t^p \leftarrow Polar(I_t^o)$
$p_t^o \leftarrow W'(I_t^o)$
$p_t^p \leftarrow W'(I_t^p)$
$p_t \leftarrow MaxVoting(p_t^o, Polar^{-1}(p_t^p))$
$m \leftarrow Metric(p_t, L_t^o)$
return m

Model. In this work, we hope to propose a method that is not bound to a specifically designated network. We choose convolutional neural network (CNN) based segmentation models, which are widely used in medical image analysis, specifically UNet and DeeplabV3+ [16]. Within these segmentation models, we also pick different convolutional neural networks, like ResNet [17] and VGG [18], to be the encoder of the segmentation models.

Loss Function. Similarly, we train the segmentation networks with two different loss functions to evaluate effectiveness: cross-entropy loss and dice loss, are shown below:

$$CrossEntropy = -\sum_{i=1}^{N}\sum_{j=0}^{2} L_{i,j}\log p_{i,j} \tag{5}$$

$$DiceLoss = \sum_{i=1}^{N}(1 - \frac{1}{3}\sum_{j=0}^{2}\frac{2|L_{i,j} \cap p_{i,j}|}{|L_{i,j}| + |p_{i,j}|}) \tag{6}$$

where L and p are the label and the prediction of segmentation networks, i denotes the index of samples, j denotes the index of classes (0 for background, 1 for LV, 2 for wall). Note that the original images and polar images use the same loss function during training.

Testing. In testing, we first transform an original image I_t^o into r-θ space and get the polar image I_t^p. Then, both original and polar images are fed into the trained segmentation model to get predictions p_t^o and p_t^p. To aggregate the predicted probabilities, we use inverse polar transformation to map p_t^p back to x-y space, namely $Polar^{-1}(p_t^p)$. The finally prediction p_t is obtained by max-voting between p_t^o and $Polar^{-1}(p_t^p)$. The final predicted value at (x, y) is calculated below:

$$p_{t,(x,y)} = \begin{cases} p^o_{t,(x,y)}, & max(p^o_{t,(x,y)}) \geq max(Polar^{-1}(p^p_t)_{(x,y)}), \\ Polar^{-1}(p^p_t)_{(x,y)}, & max(p^o_{t,(x,y)}) < max(Polar^{-1}(p^p_t)_{(x,y)}). \end{cases} \tag{7}$$

3 Experiments

3.1 Data

We use the CAMUS dataset to evaluate the effectiveness of our polar transformation method. The dataset contains 450 different patients. For each patient, there are 4 labeled echocardiogram frames: apical 2-chamber (A2C), end-systolic (ES) and end-diastolic (ED), apical 4-chamber (A4C), end-systolic (ES) and end-diastolic (ED). In total, the dataset contains 1800 labeled frames. Our segmentation models predict background, $\text{LV}_{chamber}$ and LV_{wall} regions. Additionally, we removed the labels on non-imaging regions, which are not included in the polar transformation. We used 350 patients (1400 images) for training, 50 patients (200 images) for validation, and 50 patients (200 images) for testing. All images and labels were resized to 512×512 for training and testing.

Table 1. The average percentage of each region.

Space	Non-imaging region	Imaging region				
		$LV_{chamber}$	LV_{wall}	LV_{all}	Background	Total
x-y	45.88%	9.27%	9.21%	18.48%	35.64%	54.12%
r-θ	2.10%	19.41%	22.30%	41.71%	56.19%	97.90%

3.2 Experimental Details

Comparison of x-y and r-θ Spaces. To show the effectiveness, we compared our method to 3 other methods: (1) training on x-y space only, (2) training on r-θ space only, (3) separately training on x-y and r-θ space then voting. For fairness, we controlled all methods trained with the same number of images. Specifically, the proposed method was trained for 30 epochs, and the methods (1) and (2) were trained for 60 epochs since the proposed method was trained on 2 spaces, but (1) and (2) were trained only on single space images. The loss function for the joint training also shows that the method has converged around 30 epochs.

Comparison of Segmentation Model Settings. Our base experimental model is a UNet model with ResNet34 encoder and dice loss. To test the effectiveness with different settings, we trained 3 other models: 1) change architecture to DeepLabV3+, 2) change encoder to VGG19, and 3) change loss function to cross-entropy loss. All segmentation models were trained with a batch size of 32 and optimized by the Adam algorithm with a learning rate of $1e^{-}4$.

We adopted dice similarity coefficient (DSC) as the metric to evaluate the segmentation performance for each region, which can be formulated as Eq. (7). We calculated DSC on 3 regions in x-y space: 1) $LV_{chamber}$, left ventricle outline inside the chamber wall, 2) LV_{wall}, the chamber wall, 3) LV_{all}, outline of the outside chamber that includes the wall.

$$DSC(P, L) = \frac{2 \mid P \cap L \mid}{\mid P \mid + \mid L \mid} = \frac{2TP}{TP + FP + TP + FN} \tag{8}$$

In this equation, P and L denote prediction and true label, and {TP, FP, TN} denote the number of True Positive, False Positive, False Negative pixels, respectively.

3.3 Experimental Results

We first transformed all original images and labels into r-θ space and calculate the average percentage of each region in the whole image across all images. The resulting statistics are shown in Table 1. From the table, the polar transformation can significantly reduce the non-imaging region in the whole image, and enlarge the size of foreground area and the region of interest.

Table 2. Dice similarity scores of different method with different loss functions, segmentation models and encoder networks. The scores are calculated in x-y space.

Setting	Method	$LV_{chamber}$	LV_{wall}	LV_{all}
UNet-R34-Dice	x-y only	0.9395	0.8877	0.9647
	r-θ only	0.9363	0.8779	0.9613
	x-y+r-θ+voting	0.9405	0.8886	0.9648
	proposed	**0.9418**	**0.8900**	**0.9652**
UNet-R34-CE	x-y only	0.9383	0.8849	0.9635
	r-θ only	0.9358	0.8779	0.9611
	x-y+r-θ+voting	**0.9409**	0.8882	0.9649
	proposed	**0.9409**	**0.8900**	**0.9661**
UNet-V19-Dice	x-y only	0.9386	0.8843	0.9629
	r-θ only	0.9358	0.8716	0.9579
	x-y+r-θ+voting	0.9397	0.8846	0.9630
	proposed	**0.9400**	**0.8857**	**0.9634**
DL-R34-Dice	x-y only	0.9389	0.8869	0.9644
	r-θ only	0.9370	0.8782	0.9621
	x-y+r-θ+voting	0.9415	0.8892	0.9654
	proposed	**0.9426**	**0.8929**	**0.9663**

CE: cross-entropy loss; Dice: dice loss;
DL: DeepLabV3+; R34: Resnet34; V19: VGG19.

We show our segmentation results in Table 2, with each row showing a different model choice. In each row, the proposed joint training method achieves the best results, since the proposed method can learn information from both spaces. Training on polar images only does not achieve better results compared with training on original images only; we speculate that this is because the supervision is in r-θ space, but the evaluation metric is in x-y space. Therefore, errors are accumulated when transforming from r-θ space to x-y space. Comparing "x-y only" and "r-θ only" with "x-y+r-θ+voting" results shows that the models trained on different spaces can capture more information. The voting of two single-space-trained models aggregated information from different spaces and improved the segmentation results. But the two models were trained separately, so they could not extract information as well as the jointly trained model.

We also list three groups of results with different settings: changing loss function to cross-entropy, changing encoder to VGG19, and changing segmentation model architecture to DeeplabV3+. Within each group, the proposed method outperformed all other methods, which demonstrated that the proposed method was effective with different loss functions, encoder networks, and segmentation model architectures.

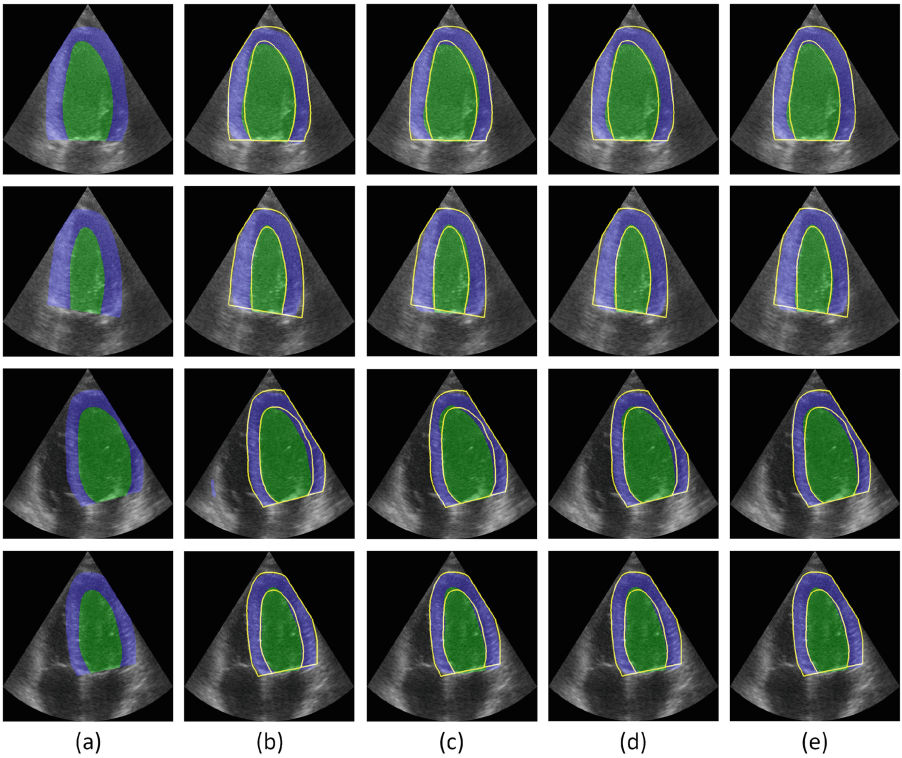

Fig. 3. Segmentation results of one subject with DL-R34-Dice setting. From top to bottom: A2C-ED, A2C-ES, A4C-ED, A4C-ES. (a) Ground-truth. (b) x-y only. (c) r-θ only. (d) x-y+r-θ+voting. (e) Propsed. The yellow contour denotes ground-truth in (b)-(e). $\text{LV}_{chamber}$: green. LV_{wall}: blue. LV_{all}: blue+green. (Color figure online)

A visualization of the segmentation results is shown in Fig. 3. The qualitative results show the effectiveness of our method, which is able to give better predictions for pixels near the region boundary.

4 Conclusion

In this paper, we have proposed a new polar transformation based method to improve echocardiography segmentation performance. The polar transformation helps the segmentation models focus more on the imaging region of the image, and the joint training on the original and polar image lets the models capture information from x-y and r-θ spaces. The max-voting aggregates prediction from 2 spaces and achieves better performance. The experimental results show our method can effectively improve the performance with different segmentation models, encoders and loss functions.

Acknowledgments. This work was funded by NSF Grant 1633295 BIGDATA: F: Collaborative Research: From Visual Data to Visual Understanding.

References

1. Troy, B.L., et al.: Measurement of left ventricular wall thickness and mass by echocardiography. Circulation **45**(3), 602–611 (1972)
2. Devereux, R.B., et al.: Echocardiographic assessment of left ventricular hypertrophy: comparison to necropsy findings. Am. J. Cardiol. **57**(6), 450–458 (1986)
3. Lang, R.M., et al.: Recommendations for cardiac chamber quantification by echocardiography in adults: an update from the american society of echocardiography and the european association of cardiovascular imaging. Euro. Heart J. Cardiovas. Imaging **16**(3), 233–271 (2015)
4. Madani, A., et al.: Fast and accurate view classification of echocardiograms using deep learning. NPJ Digit. Med. **1**(1), 1–8 (2018)
5. Østvik, A., et al.: Real-time standard view classification in transthoracic echocardiography using convolutional neural networks. Ultrasound Med. Biol. **45**(2), 374–384 (2019)
6. Leclerc, S., et al.: Deep learning for segmentation using an open large-scale dataset in 2D echocardiography. IEEE TMI **38**(9), 2198–2210 (2019)
7. Ouyang, D., et al.: Video-based AI for beat-to-beat assessment of cardiac function. Nature **580**(7802), 252–256 (2020)
8. Liu, F., et al.: Deep pyramid local attention neural network for cardiac structure segmentation in two-dimensional echocardiography. Med. Image Anal. **67**, 101873 (2021)
9. Wu, H., et al.: Semi-supervised segmentation of echocardiography videos via noise-resilient spatiotemporal semantic calibration and fusion. Med. Image Anal. **78**, 102397 (2022)
10. Madani, A., et al.: Deep echocardiography: data-efficient supervised and semi-supervised deep learning towards automated diagnosis of cardiac disease. NPJ Digit. Med. **1**(1), 1–11 (2018)
11. Zhang, J., et al.: Fully automated echocardiogram interpretation in clinical practice: feasibility and diagnostic accuracy. Circulation **138**(16), 1623–1635 (2018)
12. Feng, Zi., et al.: Two-stream attention spatio-temporal network for classification of echocardiography videos. In: 2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI), pp. 1461–1465. IEEE (2021)
13. Tan, L.K., et al.: Fully automated segmentation of the left ventricle in cine cardiac MRI using neural network regression. J. Magn. Reson. Imaging **48**(1), 140–152 (2018)
14. Chen, L.-C., et al. Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587 (2017)
15. 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
16. Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11211, pp. 833–851. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01234-2_49

17. He, K., et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016)
18. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)