Semantic Segmentation of Left Ventricles with Deep Learning

Dao Dai Vi Tran[1], Fanwei Kong[2], and Dr. Shawn Shadden[2], Principal Investigator
[1]Orange Coast College, [2]Shadden Research Group at UC Berkeley

Abstract - Semantic segmentation of cardiac structures is an important task in the development of clinical application. For example, segmentation of left ventricles can contribute in computation of cardiac functional indices, such as ejection fraction. In this research, a left ventricle image dataset was constructed and a fully convolution neural networks (FCN) based on the U-Net architecture was used as a deep learning model to segment left ventricle from medical images. Along with U-Net, we applied a combination of various image processing methods to augment the dataset. The model achieves 93% in terms of Dice score metric and produces a segmentation in millisecond. This result suggests that deep learning model has a very high potential in replacing traditional segmentation methods which require labor-intensive manual intervention.

I. Introduction

Segmentation of cardiac structures is very important in detecting any abnormal morphological changes and evaluating the progression of cardiovascular diseases (CVDs). Segmentation is often a starting point in the process of analyzing CVDs. For example, ejection fraction, systolic and diastolic volumes of the left ventricle can be computed from segmentation and evaluate ventricle contractibility. However, the current gold-standard of the segmentation process remains a large bottleneck in terms of the time and human efforts required. Cardiac structures are segmented by using traditional methods such as region growing and thresholding which are tedious and slow when there are hundreds of thousands of data samples. Thus, there exists a need for a fast, accurate, reproducible, and fully automated segmentation method to help facilitate the diagnosis of CVDs.

In this paper, we show how deep learning models could yield great results in the segmentation of the left ventricle. U-Net was chosen as a backbone semantic segmentation network since it proved itself very useful for segmentation problems with limited amounts of medical images [1]. We also used deep stack transformation (DST) [2] as our main technique for data augmentation to improve the generalization of the model.

II. Methodology

A. Dataset

There was a total of 63378 2D images with different sizes collected from two sources. The first source was 3D CT scans of 3 patients 's hearts from the Shadden Research Lab that were split into 62359 2D images with resolutions of 512 x 512 or 90 x 512. The second source was 2D images of different patients from www.ctisus.com in different modality (CT or MRI) and with varied sizes. All images were labeled using MITK software. For the first source, since the volume was large, MITK functions (mainly interpolation) were used to facilitate the process. The pixel values in label images were either 1 for left ventricle or 0 for the rest.

B. Preprocessing

All images from the second source were converted from three channels to one channel (grayscale). All images from both sources were resized into 256 x 256 images after being padded to become square images. The pixel values in the paddings are equal to the minimum pixel value of the image. Furthermore, pixel values that are above minimum pixel value plus 85% of range or below minimum pixel value plus 2% of range were eliminated from the image. Tensorflow was used as a library to build and train the model. All images were split into training set, validation set and test set with ratio of 70 : 15 : 15, and the Tensorflow data pipeline was used to speed up data input and output.

C. Network Architecture

C1. Modified U-Net

The U-Net in our model was slightly modified by adding one encoder block with 2 convolution layers of 32 filters at the beginning and one encoder block with 2 convolution layers of 32 filters at the end. Batch Normalization was added after each convolution layer to stabilize and speed up training [3].

C2. Deep Stack Transformation (DST)

DST is a series of n stacked augmentation transformation applied input images during training. Each transformation is an image processing function with two hyper-parameters: probability p and magnitude m.

$$(\hat{x}_s, \hat{y}_s) = \tau^{n}_{p_n,m_n}(\tau^{n-1}_{p_{n-1},m_{n-1}}(...\tau^{1}_{p_1,m_1}(x_s, y_s)))$$

where x_s and y_s are input image and its corresponding label, respectively and τ represents the image processing functions. Augmentation transforms alter the image quality, appearance, and spatial structure. Specifically, DST in this model consisted of the following transforms: sharpening, blurring, perturbation, brightness, contrast, scaling, shifting, flip, rotation and distortion in addition to [-1, 1] normalization per image at the beginning. Augmenting image sets during training can result in models with more robust segmentations than if data processing was performed at the inference stage [2].

Image Quality was related to sharpness and blurriness of medical images. Gaussian filtering was used to blur the image with a magnitude ranging between [0.25, 1.75]. Sharpness had a reverse effect by using an unsharp masking with strength [0.5, 2.0].

Image Appearance was associated with the statistical characteristics of image intensity such as brightness, contrast and perturbation. Brightness augmentation referred to random shift [-0.15, 0.15] in the intensity space. Contrast augmentation referred to gamma correction ranging between [0.5, 1.5]. Perturbation augmentation referred to random scaling and shifting with scaling magnitude ranging between [0.9, 1.1] and shifting magnitude ranging between [-0.1, 0.1].

Spatial Transforms included scaling, shifting, flip, rotation and distortion. Random scaling was used with magnitude [0.5, 1.5]. Random rotation was used with angle [-180, 180]. Distortion transform was created by applying a non-linear warp to the image. The warp for x and y direction was created by applying gaussian filter of mean 0 and std 4.0 into two matrices with random numbers from -1 to 1, and multiplying them by random alpha between [2, 10]. Each transformation in DST had probability of 0.23.

D. Experiments

The approach was implemented in Tensorflow [7] and train on 8 cores of 1080Ti offered by UC Berkeley high performance computing RIT. The mini batch sizes chosen for the training and validation was 15 with 50 epochs. We used the summation of soft Dice loss and cross entropy loss and default Adam optimizer from Tensorflow.

III. Results

The following table summarizes our results and the figures show the loss curve of the U-Net with DST during training. As the training loss curve (blue) and validation loss curve (yellow) follow each other, the figure suggests that our model did not overfit the data.

Model	Training Loss	Validation Loss	Validation Accuracy	Test Accuracy
U-Net	0.0522	0.0691	93.09%	92.86%
U-Net with DST	0.0479	0.0690	93.10%	93%

Table 1. Comparison of baseline U-Net and U-Net with DST

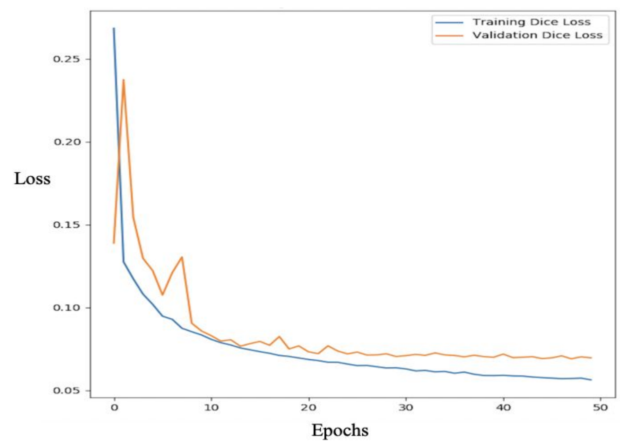

Figure 2. Training and Validation Dice Loss

Although the overall result achieved was 93%, it is important to point out that the prediction accuracy for the ctisus images test set was about 66.7%. This was due to larger data variation and the small number of images in the Ctisus dataset compared to the total data of 63378. However, the model gave an amazing result on the data from the Shadden Research Lab, indicating the model overfitted to Shadden Research Lab data. This result shows the need to acquire more data from various sources and modalities for the deep learning model to generalize better to unseen data.

IV. Conclusion and Future Work

Result suggests that deep learning model is capable of replacing traditional segmentation methods with its high segmentation accuracy in milisecond. One immediate method that may improve the overall performance of the model is to improve the amount of data and data quality. Having experts to manually label the data will help reduce uncertainties for the model. The second method is to apply transfer learning from other models that also perform on medical images since transfer learning has shown promise in reducing training cost, reducing the amount of data required and improving the segmentation performance when applying to tasks trained with similar image datasets[4][5]. Lastly, since the hyperparameter in this model was chosen manually, it is highly recommended to apply random search strategy to tune all parameters [6].

V. References

1. O. Ronneberger, P. Fischer, T. Brox, *U-Net: Convolution Networks for Biomedical Image Segmentation*, Computer Science Department and BIOSS Centre for Biological Signalling Studies, University of Freiburg, Germany, 2015

2. L. Zhang, X. Wang, D. Yang, T. Sanford, S.Harmon, B. Turkbey, H. Roth, A. Myronenko, D. Xu, and Z. Xu, *When Unseen Domain Generalization is Unnecessary ? Rethinking Data Augmentation. arXiv:1906.03347, 2019.*

3. S. Ioffe and C. Szegedy, *Batch Normalization: Accelerating deep network by reducing internal covariate shift.* arXiv: 1502.03167, 2015.

4. J. Yosinski, J. Clune, Y. Bengio, H. Lipson, *How transferable are features in deep neural networks ?*. Advanced in neural information processing systems, pp. 3320-3328 (2014).

5. V. Iglovikov, A. Shvets, *TernausNet: U-Net with VGG11 Encoder Pre-trained on ImageNet for Image Segmentation*. arXiv: 1801.05746, 2018.

6. J. Bergstra, Y. Bengio, *Random Search for Hyper-Parameter Optimization*, JMLR, 2012

7. https://github.com/Daividao/berkeley_reu.git

VI. Acknowledgement

I would like to thank my mentor, Fanwei Kong, for helping me throughout the research. I also want to thank Professor Shawn Shadden, for giving me the opportunity to be an intern in his research group. Thank you, Nicole and Kimmy, for organizing the TTE REU program and for your help throughout the program.