Prostate Structure Segmentation using U-Net

Paritosh Mittal
Email: paritosh.mittal09@gmail.com

I. Introduction

The objective of the prostate structure segmentation problem is to detect and segment the peripheral zone (PZ) and the less accessible central gland (CG) structures in the MRI image automatically. Considered in this form, each image has two categories of interest (PZ and CG) and three distinct image regions to be identified in the segmented output: PZ, CG, and everything else (background).

II. Database and Experimental Protocol

The NCI-ISBI 2013 challenge database consists of prostate MRI data in DICOM format for 60 subjects in the training set, 10 subjects in the leaderboard set, and 10 subjects in the test set with the ground truth provided in the NRRD format. Out of the 60 subjects, there is a mismatch in the DICOM and NRRD data for one subject (ProstateDx-01-0055) whose data has been excluded from the training data for simplicity. With the remaining data, there are a total of 1,521 images in the training set, 261 in the leaderboard set, and 271 in the test set. The leaderboard and train sets are combined together into the training set and a 80-20 split of the data is used for training and validation during the network training. The test set is used only to report the results of the network. In order to establish the baseline, experiments are conducted with various configurations of the input and network architecture. The three architectures are summarized in Table I. Training is conducted using a dropout keep fraction of 0.8, a momentum value of 0.99, learning rate of 0.001, and 30 epochs. The batch size is adjusted so as to best utilize the available memory. Since the standard network has a large number of parameters and takes a long time to train, results have been reported for lesser number of epochs due to computational limitations. Four metrics [5] are used for evaluating the performance of the model: Rand error, pixel error, Intersection over Union (IOU/Jaccard index), and the Dice coefficient. The pixel error is computed as the euclidean distance between the actual mask and the predicted mask. The rand error is given as follows:

$$RE = 1 - \frac{a+b}{\binom{N}{2}} \tag{1}$$

where,a denotes the pixels which belong to the same objects in both the ground truth and predicted mask, b denotes the pixels which belong to different objects in both the ground truth and the predicted mask, and N is the total number of pixels in either mask. Given two sets A and B, the IOU metric, also called the Jaccard index (denoted by $J(A,B)$) is computed as follows:

$$J(A,B) = \frac{|A \cap B|}{|A| + |B| - |A \cup B|} \tag{2}$$

TABLE I: Describing the different network architectures evaluated in the baseline.

Name	Layers	Parameters(96×96)	Starting Feature Channels
Standard U-Net	23	31,031,685	64
Simple U-Net	23	121,653	4
Reduced U-Net	13	7,397	4

The Dice coefficient (denoted by $D(A,B)$), on the other hand, is computed as follows:

$$D(A,B) = \frac{2|A \cap B|}{|A| + |B|} \tag{3}$$

Both the Dice coefficient and the Jaccard index have been used in medical imaging literature as evaluation metrics for evaluating the performance of segmentation algorithms. The existing codebase initially operated using the Dice coefficient, but the Jaccard index has also been incorporated for reporting the results. The two are related as given in the below equation:

$$J = \frac{S}{2 - S} \tag{4}$$

A. Preprocessing

The NRRD masks in the data contain three-class labels where the background pixels are marked with values close to 0, the PZ region pixels are marked with 1, and the CG region pixels are marked with 2. In order to derive the baseline, both the PZ and CG regions are considered to collectively be the foreground (value=1) and the problem is reduced to a two-class segmentation. The MRI images are of varying sizes, so they are resized to a common size while preserving the original aspect ratio of 1:1. Two sets of experiments are computed to gauge the effect of using two different input sizes (96×96 and 224×224) on the performance of the network. Histogram equalization is performed on the loaded DICOM images after resizing and they are then stored in the Numpy array format. Similarly, the NRRD masks are also resized and stored in the same format for further processing. Since interpolation during resizing changes the pixel values from 0 and 1 to a float value between the two, binary thresholding is applied again and any pixel less than 0.45 is considered to be 0 and all the other pixels are set to 1.

In order to extend the baseline algorithm to the full three-class problem, the NRRD mask has to be thresholded such that the resized mask has 0, 1, and 2 as the only unique pixel values. Two thresholds need to be decided for the same. Once the masks are prepared, the binary cross entropy loss function has to be changed to a multi-class cross entropy function such as the sparse categorical cross entropy function. Due to

TABLE II: Results on the Prostate segmentation database on the validation (leaderboard) and test sets when trained using dice coefficient loss. The networks are described in Table I.

Image Size	Network Architecture	Validation				Testing			
		PE	RE	Dice Coefficient	Jaccard Index	PE	RE	Dice Coefficient	Jaccard Index
96 × 96	Standard U-Net	Did not converge							
96 × 96	Simple U-Net	0.01	0.45	0.68	0.52	0.01	0.43	0.68	0.52
96 × 96	Reduced U-Net	0.02	0.31	0.65	0.48	0.02	0.31	0.68	0.52
224 × 224	Standard U-Net	Did not converge							
224 × 224	Simple U-Net	0.01	0.32	0.70	0.54	0.01	0.30	0.73	0.53
224 × 224	Reduced U-Net	0.03	0.65	0.54	0.37	0.02	0.67	0.60	0.43

TABLE III: Results on the Prostate segmentation database on the validation (leaderboard) and test sets when trained using binary cross entropy loss. The networks are summarized in Table I. [1] Due to high runtime per epoch, the results reported for this configuration are obtained after 10 epochs. [2] Due to high runtime per epoch, the results reported for this configuration are obtained after 5 epochs.

Image Size	Network Architecture	Validation				Testing			
		PE	RE	Dice Coefficient	Jaccard Index	PE	RE	Dice Coefficient	Jaccard Index
96 × 96	Standard U-Net [1]	0.02	0.42	0.76	0.60	0.02	0.38	0.75	0.60
96 × 96	Simple U-Net	0.02	0.37	0.67	0.51	**0.01**	0.43	0.70	0.54
96 × 96	Reduced U-Net	0.05	0.72	0.56	0.39	0.04	0.72	0.56	0.39
224 × 224	Standard U-Net [2]	0.02	0.37	0.76	0.60	0.02	0.36	**0.77**	**0.63**
224 × 224	Simple U-Net	0.01	0.38	0.64	0.49	**0.01**	**0.34**	0.70	0.53
224 × 224	Reduced U-Net	0.05	0.72	0.52	0.35	0.05	0.73	0.53	0.36

computational limitations, this experiment has not yet been conducted as part of this report.

B. Data Augmentation

U-Net is based around the extensive use of data augmentation to enable deep learning based solutions for medical imaging problems where the amount of labeled data is originally quite low. The same goes for this problem where the total number of original images for training is around 2000. The Keras library is used to augment the training data and generate close to 150,000 training images from the original training data using operations such as rotation, flipping, scaling+cropping, and translation. Elastic deformation is used to supplement the Keras data generation algorithm.

III. Experiments and Analysis

A total of 6 combinations of image size and network architecture have been evaluated as part of this baseline experiment. It was observed that the standard U-Net has issues with converging when trained from scratch using either image size when the Jaccard index/Dice coefficient is used as the loss function (Table II). Noticeably better results are observed when binary cross entropy is used as the loss function instead (Table III). Both sets of results are presented in the report for comparison.

As is evident, the original architecture with the most number of parameters achieves the best case performance of 0.63 Jaccard index when used with larger 224 × 224 images. The best case predictions made by the two top performing networks are presented in 1. It is observed that both networks achieve their best performance on disjoint input cases, indicating that image resolution has substantial impact on how the networks behave for this problem. Uncorrelated outputs make an encouraging case for potential performance improvement to be had with the usage of ensembles to create a more robust algorithm. The smaller architectures train faster due to the lesser parameters but are also unable to match up to the bigger networks' performance when Jaccard index/dice coefficient are considered. In terms of rand error, however, the best performance of 0.34 belongs to the simple U-Net when trained with 224 × 224 images. The best and second best network perform similarly in rand error to each other and with the simple U-Net with 0.36 and 0.38 rand error, respectively.

IV. Software Engineering Decisions

In order to quickly process the medical imaging data and get started with the U-Net architecture, multiple related existing open-source projects were referenced [2], [3], [4]. All pieces of the end-to-end processing pipeline which were already available in these existing code-bases were used after verifying that they are functioning as intended. Any additional functionality required to complete the pipeline were coded from scratch. There were some compatibility issues when reusing existing code in terms of version mismatch of libraries and Python itself. As explained in the analysis, dice coefficient based loss did not work as well as the binary cross entropy loss function.

Since a GPU-powered machine was not available throughout the exercise, experiments with the full-fledged version of U-Net and high image sizes were limited. The size of the training data after augmentation was also kept to a modest 150,000 images. A single epoch with the augmented training data and original U-Net architecture running on a 16 GB Core i7 CPU @ 3.9 GHz had an ETA of approximately 90 minutes for 96 × 96 images. This motivated the choice to also evaluate smaller variants of the original U-Net (simple and reduced) with much less parameters to be learned leading to faster

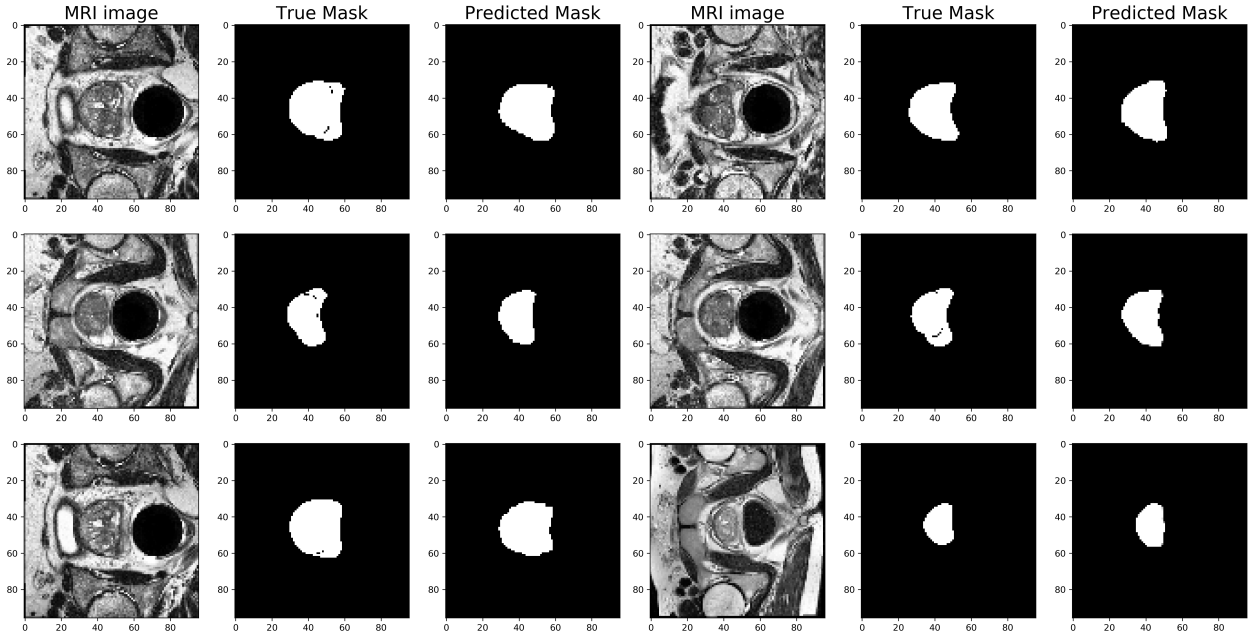

Fig. 1: Top three predictions made by the (a) best (standard U-Net on 224 $\times$ 224 images) and (b) second best (standard U-Net on 96 $\times$ 96 images) network configuration.

training times. The training code is ready to be run in either GPU/CPU mode depending on the hardware and version of Tensorflow that is installed in the development environment. Time was mostly spent on running and validating the different experiments.

V. Conclusion and Future Work

In this report, a baseline network for prostate segmentation is designed and evaluated using the NCI-ISBI 2013 data. The best baseline network achieves an average Jaccard index of 0.63 on the test data. Judging by the results, it appears to provide a satisfactory baseline for the problem, being able to handle a large majority of the test cases well when both PZ and CG are classified together as the region of interest (ROI). Listed below are several directions for future work to further improve upon this baseline:

- Extending the baseline network to handle the three-class case where PZ and CG are identified as separate objects. This can be accomplished by (a) using either a newly trained network, or (b) using the existing two-class segmentation output to perform hierarchical segmentation to distinguish between PZ and CG.
- Trying a related and larger database with more annotated samples to pre-train the network before fine-tuning it for prostate segmentation.
- Trying more data augmentation techniques and generating more augmented data.
- Trying an ensemble of two or more networks (variants of U-net or other segmentation networks) at the predicted mask level with multiple networks providing a combined class probability for each pixel.
- Trying larger input image sizes.
- Tuning the various hyperparameters such as learning rate, decay, momentum, and number of epochs.
- Trying different evaluation metrics: topology preserving warp error in addition to or instead of the Jaccard index.
- Trying different preprocessing operations in addition to histogram equalization.
- Combining with other existing segmentation techniques (non deep-learning based).
- Using the visual saliency and/or entropy map(s) of the image as additional channel(s) to be fed along with the original greyscale image as input to the network. These can act as auxiliary sources of information for the network.

References

[1] O. Ronneberger and P. Fishcer and T. Brox, *U-Net: Convolutional Networks for Biomedical Image Segmentation*, http://arxiv.org/abs/1505.04597, 2015.

[2] Inom Mirzaev, *https://github.com/mirzaevinom/prostate*$_s egmentation$

[3] J. Akeret and C. Chang and A. Lucchi and A. Refregier, *Radio frequency interference mitigation using deep convolutional neural networks*, Astronomy and Computing Vol. 18, pages 35-39, 2017.

[4] *https://github.com/zhixuhao*

[5] WWW: Web page of the em segmentation challenge, *http://brainiac2.mit.edu/isbi*$_c hallenge/$