ProstateZone Classifier: Enhancing Transition Zone Lesion Classification with Deep Learning

Abhinav Kumar
Stanford University
abhi1@stanford.edu

Hannah Prausnitz-Weinbaum
Stanford University
hannahpw@stanford.edu

Samrat Thapa
Stanford University
samthapa@stanford.edu

Abstract

Classification of suspected lesions is critical for timely intervention and preventing unnecessary treatments in prostate cancer cases. In the clinic, urologists often misclassify lesions in the transition zone (TZ) of the prostate, resulting in a high false positive classification rate. We classify prostate lesions in two steps: zone segmentation and lesion classification. We use the PROSTATEx dataset of 204 T2-weighted MRI images with labeled lesions. For zone segmentation, we apply a 2D U-Net, 3D U-Net, and 3D UNETR. For lesion classification, we apply a YOLOV5 model. We additionally classify lesions in a separate model based on masked TZ input, in order to better classify lesions in this zone. We find that model performance is best for 2D segmentation.

1. Introduction

Classification of suspected lesions is critical for timely intervention and preventing unnecessary treatments in prostate cancer cases. The Prostate Imaging Reporting and Data System (PI-RADS) was created to predict and characterize lesions from MRI images on a scale of 1 to 5, with 5 being a high suspicion of malignancy (Barentz et al [1]). In the clinic, urologists often misclassify lesions in the transition zone (TZ) of the prostate, resulting in a high false positive classification rate (Gaudiano et al [7]). This is because the PI-RADS score for a lesion varies depending on whether it occurs in PZ and TZ boundaries. However, if the boundaries of these regions are difficult to discern, then the classification task becomes difficult. In addition, the effects of aging or disease lead to significant structural changes in the TZ (Clark et al [4]) and can make this task difficult. We perform PZ and TZ region segmentation and then aim to detect and classify lesions in these regions more accurately.

We classify TZ lesions in two steps. First, we segment the PZ and transition zone TZ using a modified U-Net. The input to this step is raw MRI images, and the output is a mask that classifies each pixel as PZ, TZ, or background. Second, we detect and classify lesions in the PZ and TZ regions using YOLOV5. The input to this step is also raw MRI images, and the outputs are bounding boxes around each lesion labeled with a classification of benign or malignant.

2. Related Work

Previous work has used deep learning methods on MRI data to classify prostate lesions. Many studies have found success with U-Net, including Mehralivand et al [13], Pellicer-Valero et al [15], and Sanyal et al [18]. Vente et al [21] propose a 2D U-Net with MRI slices as input. Other studies use pretrained CNNs such as VGGNet, including Chen et al [3], and Hassan et al [10], who also apply a fusion model. Other work proposes a capsule network (Li et al [12]), explainable AI (Hamm et al [9]), MRRN-DS (Simeth et al [19]), and a customized CNN (Cao et al [2]).

Much recent work uses a single model to perform legion detection and classification. Some work, including Yu et al [22], first identify regions of potential lesions, and then probes each region to classify the lesion as malignant or benign. Duran et al [6] use a 2-stage approach, where outputs of prostate segmentation are used as attention inputs to a second stage that predicts malignancy.

However, previous work does not focus deep learning approaches on predicting lesions within the TZ alone, which is an area of high importance in this field. We take on this challenge in our project.

3. Methods

3.1. Zonal Segmentation

3.1.1 Models

The first step of our task was to segment the prostate zones into three classes: TZ, PZ, and background. Because our data was presented as 2D slices of a 3D prostate MRI, we utilized and compared 2D and 3D segmentation techniques.

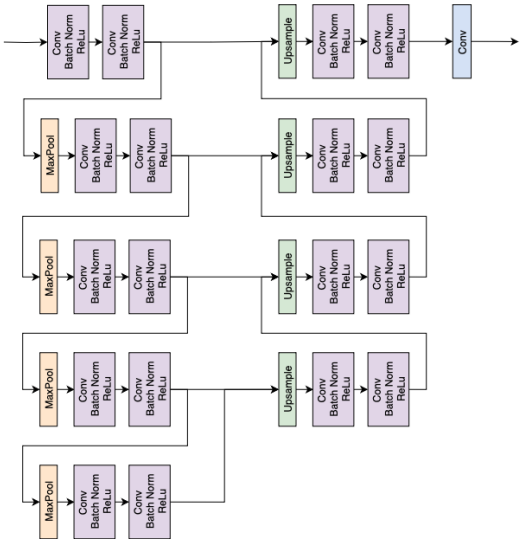

Figure 1. 2D U-Net Model Architecture

We performed 2D segmentation using a U-Net, and 3D segmentation using a 3D U-Net and UNETR.

For 2D segmentation, we used U-Net. We used an existing Pytorch model implementation [14] from the original U-Net paper [17], which consisted of four down layers followed by four up layers. Each down layer consists of max pooling followed by two iterations of convolution-batchnorm-ReLU, while each up layer consists of upsampling followed by two iterations of convolution-batchnorm-ReLU. Each up layer receives a residual connection from the corresponding down layer. The final U-Net architecture is shown in Figure 1.

For our 2D segmentation baseline, we used a simplified version of this model, which consists of a basic fully convolutional network containing three convolutions and three up-samples.

For 3D segmentation, we used an existing Pytorch model implementation of 3D U-Net [8]. This implementation was adapted and optimized for 3D using the 2D U-Net by modifying all 2D transformations (convolution, batchnorm, max pooling, upsampling) to be 3D transformations instead. All other elements of the architecture remained unchanged.

We also experimented with a UNETR, a 3D transformer-based segmentation model. This model uses a ViT Transformer backbone as its encoder, with 12 transformer blocks and uses a 3D convolutional decoder (segmentation head). The ViT Transformer encoder divides the image into patches, which are similarly embedded into sequences and assigned positional encodings, and processes the patches through transformer blocks. The UNETR decoder architecture resembles U-Net: features from multiple resolutions of the transformer encoder (z3, z6, z9, z12) are skip-connected to the decoder. The input to the decoder is a sequence of representations from the encoder, which are upsampled repeatedly for higher segmentation resolution. We used an existing PyTorch implementation of UNETR as well as part of the training details [5]. The architecture can be found in Hatamizadeh et al [11].

3.1.2 Loss

For segmentation, we use DiceCELoss, a composite loss defined as follows:

$$\mathcal{L} = \mathcal{L}_{dice} + \mathcal{L}_{ce} \tag{1}$$

We minimize this loss function during training to improve both the segmentation and classification performance of the network. Dice loss, $\mathcal{L}_{dice}$, is defined as:

$$\mathcal{L}_{dice} = 1 - \frac{2\sum_{i=1}^{N} p_i g_i}{\sum_{i=1}^{N} p_i^2 + \sum_{i=1}^{N} g_i^2} \tag{2}$$

where p_i is the predicted probability for pixel i, g_i is the ground truth label for pixel i, and N is the total number of pixels in the segmentation mask. The Dice loss measures the overlap between the predicted segmentation masks, p_i, and the ground truth masks, g_i.

Cross Entropy loss, $\mathcal{L}_{ce}$, is defined as:

$$\mathcal{L}_{ce} = -\sum_{c} g_c \log(p_c) \tag{3}$$

where p_c is the predicted probability for class c and g_c is the ground truth label for class c.

3.2. 2D Lesion Detection and Classification

The second step of our task is to use the segmented prostate inputs to bound and classify lesions with the TZ and PZ. Due to our notably higher results of the 2D segmentation model compared to the 3D segmentation models, we use 2D slices of the MRIs as input as opposed to the 3D volume. Therefore, we frame the problem of lesion classification as a 2D object detection task. We apply YOLO [16] to prostate inputs and TZ-only segmented inputs separately. YOLO predicts bounding boxes around the lesions and predicts class probabilities (either benign or malignant) directly in a single evaluation.

3.2.1 Model

We used YOLOv5. YOLO (You Only Look Once) is an object detection and classification framework that predicts bounding boxes and class probabilities of detected objects, lesions, in a single forward propagation step. The model architecture consists of a single convolutional neural network (CNN) with Darknet-52 for feature detection from the input image, several detection layers, anchors for bounding boxes (Redmon et al., 2015). We use an existing Pytorch implementation of YOLOv5 [20].

3.2.2 Loss

We train YoloV5 using another composite loss function defined as follows:

$$\mathcal{L} = \mathcal{L}_{box} + \mathcal{L}_{obj} + \mathcal{L}_{cls} \tag{4}$$

Box loss, $\mathcal{L}_{box}$, is defined as:

$$\mathcal{L}_{box} = \frac{1}{N}\sum_{i=1}^{N}(1 - \text{IoU}(p_{box,i}, g_{box,i})) \tag{5}$$

where $p_{box,i}$ is the predicted bounding box for target i, $g_{box,i}$ is the ground truth bounding box for target i, and IoU is the intersection over union between the predicted and ground truth bounding boxes. Objectness loss, $\mathcal{L}_{box}$, is defined as:

$$\mathcal{L}_{obj} = \frac{1}{N}\sum_{i=1}^{N}\text{BCE}(p_{obj,i}, t_{obj,i}) \tag{6}$$

where $p_{obj,i}$ is the predicted objectness score for target i, $t_{obj,i}$ is the ground truth objectness score for target i, and BCE is Binary Cross Entropy loss.

Classification loss, $\mathcal{L}_{cls}$, is defined as:

$$\mathcal{L}_{cls} = \frac{1}{N}\sum_{i=1}^{N}\text{BCE}(p_{cls,i}, t_{cls,i}) \tag{7}$$

where $p_{cls,i}$ is the predicted class probability for target i, $t_{cls,i}$ is the ground truth class label for target i, and BCE denotes the Binary Cross Entropy loss.

4. Dataset

4.1. PROSTATEx Dataset

We use the PROSTATEx dataset of 204 prostate transversal T2w MRI scans, obtained from GitHub. The MRIs were square in height and width, ranging from 320x320 to 640x640, and the depth of the MRIs ranged from 19 slices to 27 slices. The data also included annotated PZ and TZ segmentations for each scan, formatted as a binary mask. The dataset also includes lesion segmentations for each lesion in each scan, formatted as a binary mask and given a 0/1 label for clinical significance. Clinical significance, as defined in this dataset, represents lesions with Gleason Score > 6 and Gleason Grade Group/ISUP score > 1, so we interpreted lesions as "malignant" if clinically significant and "benign" if not clinically significant.

For the segmentation task, we used the PZ/TZ segmentations directly from the dataset as our ground truth. For the object detection task, we drew bounding boxes around the provided lesion segmentation masks and used the bounding boxes as our ground truth. A sample MRI image from our dataset, along with the corresponding ground truth PZ/TZ segmentations, lesion segmentations, and extracted bounding boxes, is shown in Figure 2.

4.2. Preprocessing

We used a random 80/10/10 train/val/test split and preprocessed the MRI data into 2D slices for the 2D models and 3D volumes for the 3D models. All data provided by this dataset is formatted using Neuroimaging Informatics Technology Initiative (NIfTI) format. Preprocessed 2D MRI slices and corresponding ground truth annotations were saved in Portable Network Graphics (.png) format, and preprocessed 3D data remained in NIfTI format.

For all models, we normalized the data using the mean and standard deviation of the training dataset. For the 2D segmentation models, we then resized the slices to 256 x 256. For the 3D segmentation models, we resized the height and width dimensions to 320 x 320 and padded the depth dimension to 32. This reshaping ensured we could use batches for both 3D segmentation models and a patch size of 16 for the ViT-based UNETR.

We also utilize masking strategies, which required image preprocessing, to train variations of our object detection model. For our TZ only object detection model, we create a duplicate object detection dataset with the PZ segmentation masked out in the input MRI. We also remove the ground truth bounding boxes in the masked PZ region. For our PZ only model, we create another duplicate object detection dataset with the TZ segmentation masked out in the MRI. We also remove the ground truth bounding boxes in the masked TZ region also removed). Figure 3shows the number of benign and malignant bounding boxes in the full object detection dataset, the TZ only dataset, and the PZ only dataset.

5. Results and Discussion

5.1. Metrics

We report two metrics for our segmentation models: Dice and IoU. Both metrics compute the overlap of the output segmented region with the ground-truth region. These models are helpful for our use case because the majority of each image contains background, and we aim to correctly classify TZ and PZ pixels. We aimed for a Dice score $>= 0.7$ and IoU score $>= 0.6$ to indicate a well-performing model.

Independently from our primary model, we trained a fully convolutional network with three convolutions and three up-samples, obtaining loss shown in Figure 5 and scores shown in Table 1. We used this as a baseline when training the U-Nets.

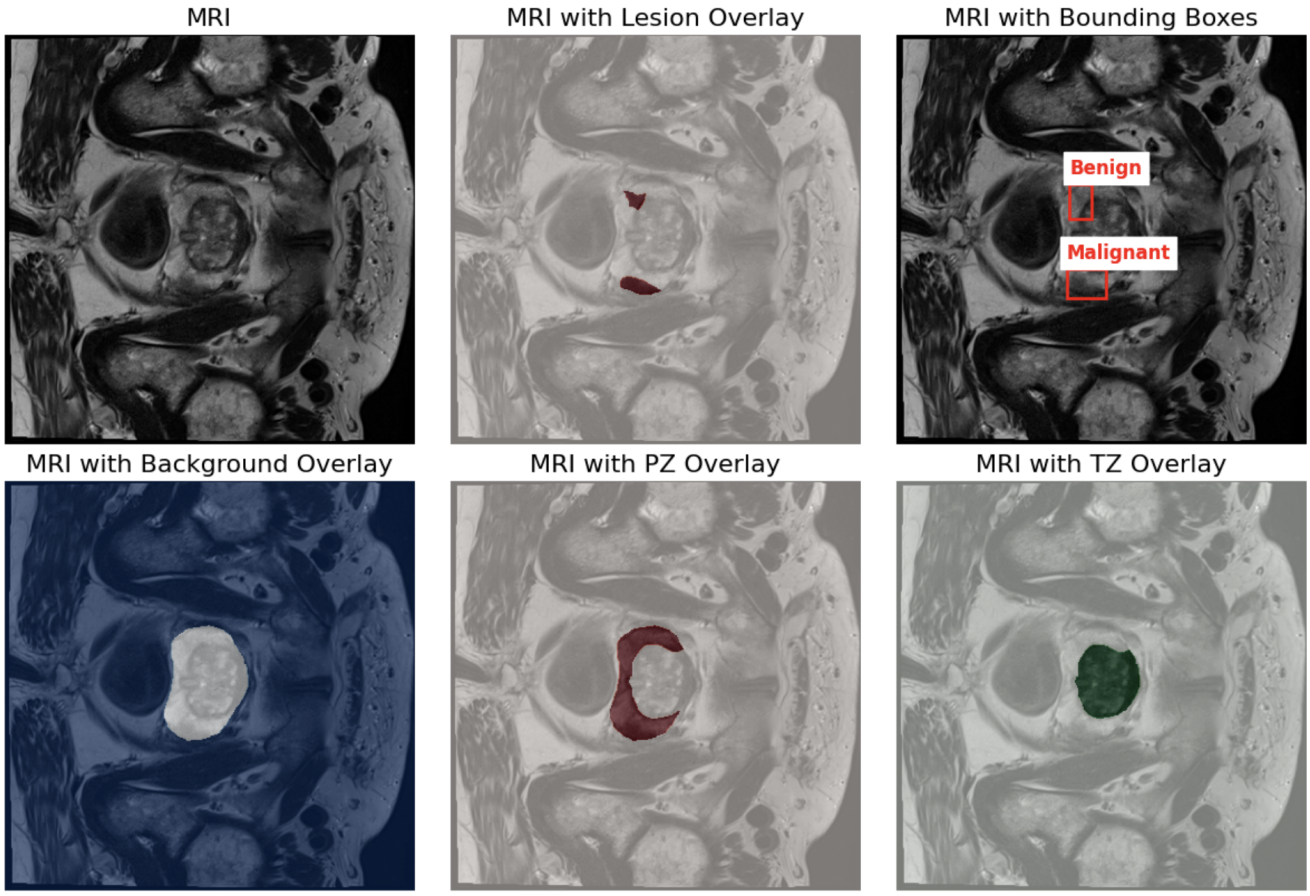

Figure 2. Visualization of MRI with various annotation overlays

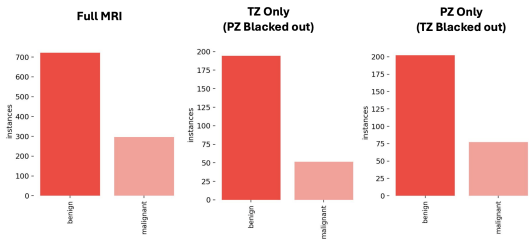

Figure 3. Number of benign and malignant bounding boxes

5.2. Segmentation

5.2.1 2D Segmentation

Our final 2D segmentation model uses AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.95$, $\texttt{weight\_decay} = 0.1$, and and a learning rate of 4.3×10^{-3}. We chose AdamW to combine the fast convergence of Adam with regularization within the optimizer. We lower β_2 from the default value of 0.999 because our dataset is relatively small, allowing more focus on recent examples and faster learning. We increase the weight decay substantially from its default value of 0.01 to mitigate overfitting problems in the model. While the model initially displays overfitting within the first five epochs, it shows no overfitting after 15 epochs after changing weight decay to increase regularization. We increase the learning rate to allow the model to avoid local minima, as it exhibits a plateauing with lower learning rates. We use DiceCE Loss with $\texttt{smooth\_nr} = 0$ and $\texttt{smooth\_dr} = 1 \times 10^{-6}$. We decrease the smoothing parameters to avoid their effects on loss calculations, but leave a small $\texttt{smooth\_dr}$ to avoid division by zero.

We train the model on a T4 GPU, which has 16 GB GPU memory. We set a batch size of 16, which was the maximum batch size that fit in memory on our GPU, in order to maximize stability during training. We trained the model for 15 epochs, at which point the model showed limited improvement each epoch. We used our validation set to choose hyperparameters and identify overfitting. The loss over training and change in validation metrics are shown in Figure 6. After training and hyperparameter tuning, we evaluated our model on our holdout test set, achieving metrics displayed in Table 1.

The 2D U-Net model outperformed our baseline metrics and is our overall best segmentation model. A sample com-

parison of predicted and ground-truth PZ and TZ segmentations is shown in Figure 4.

5.2.2 3D segmentation

The 3D U-Net model was trained on an A100 40GB GPU. We use a batch size of 4, as this is the largest batch size that fits on our machine given the size of 3D volumes. We tuned our learning rate to 4×10^{-3} to allow stable learning, and ran the model for 20 epochs. All other hyperparameters are as described with 2D segmentation. The loss is shown in Figure 7, and test performance in Table 1.

The 3D UNETR model was trained on an A100 40GB GPU. We use a batch size of 4, as this is the largest batch size that fits on our machine given the size of 3D volumes. We tuned the `weight_decay` parameter to 0.05 given that overfitting is less prevalent in transformer models, and ran the model for 5 warmup epochs followed by 100 epochs.We also use a layer-wise learning rate decay following a cosine decay schedule in order to stabilize ViT training. All other hyperparameters are the same as the 2D segmentation training. The tables below show the loss over training and metric performance. The loss is shown in Figure 8, and test performance in Table 1.

Loss curves and metrics over training are shown for all segmentation models. The simple CNN is displayed in figure 5; the 2D U-Net is displayed in figure 6; the 3D U-Net is displayed in figure 7; and the 3D UNETR is displayed in figure 8.

Model	Test MeanIoU	Test MeanDice
2D Simple CNN	0.3390	0.4703
2D U-Net	0.6873	0.7978
3D U-Net	0.5873	0.7325
3D UNETR	0.6097	0.7451

Table 1. Performance of Segmentation Models on Test Set

Although both 3D segmentation models performed well, they did not perform as well as the final 2D segmentation model. This may be due to the relatively small number of training samples, making it difficult to train this large model. Using slices as input in the 2D model provides about 20 times as many training samples as using 3D volumes as input.

5.3. Object Detection

We started with YOLOv5 using pre-trained weights derived from the COCO dataset. We then trained three distinct YOLO models with different inputs. The first model is our baseline and is trained on all MRI images with both the PZ and TZ regions. In order to reduce the number of false positives in the TZ region lesions, we hypothesized that masking out the segmented PZ region in the input MRI will improve the classification accuracy of the lesions in the TZ zone, and vice versa. Therefore, we trained two other models: one with the PZ segmentation masked out in the input MRI (with ground truth bounding boxes in the masked PZ region also removed) and one with the TZ segmentation masked out in the MRI (with ground truth bounding boxes in the masked PZ region also removed). We do not handle cases in which lesions overlap in the PZ and TZ.

We trained these three models on a T4 GPU (16 GB GPU memory). We used a batch size of 32, which was the maximum batch size we could use given our GPU memory in order to maximize learning stability. We used a learning rate of 1e-2 and an Adam optimizer with $\beta_1 = 0.937$ and $\beta_2 = 0.999$. Evaluation on our holdout test set yielded the metrics below.

Model	Class	Prec.	Recall	mAP50
Full MRI, all labels	A	.002	.479	.0219
Full MRI, all labels	B	.003	.625	.0378
Full MRI, all labels	M	.001	.333	.006
Full MRI, PZ labels	A	.706	.140	.0139
Full MRI, PZ labels	B	1	0	.0141
Full MRI, PZ labels	M	.413	.279	.264
Full MRI, TZ labels	A	.001	.0385	.0188
Full MRI, TZ labels	B	.002	.077	.0376
Full MRI, TZ labels	M	0	0	0
TZ-only MRI	A	.003	.229	.0024
TZ-only MRI	B	.004	.292	.0037
TZ-only MRI	M	.002	.167	.0011
PZ-only MRI	A	.001	.152	.0008
PZ-only MRI	B	.002	.161	.0013
PZ-only MRI	M	.001	.143	.0004

Table 2. Performance Metric for YOLO Lesion Detection Models. Class labels are given by A: all, B: benign, M: malignant.

A sample object detection prediction from our best performing object detection model "Full MRI, PZ labels" is shown in Figure 9

Overall the YOLO model performs poorly on the 2D MRI images for the detection of benign and malignant lesions. The extremely low precision, recall, and mAP50 indicate the model fails to detect both benign and malignant cases from the prostate MRI. However, we do notice some differences based on what object labels we input. Specifically, providing PZ bounding box labels only improves precision and therefore reduces false positives but also misses true positives. In addition, when we mask out either region, PZ, or TZ, the model performs worse, contradictory to our hypothesis.

One possible explanation for why models trained with either PZ, or TZ, masked out performed poorly could be due to the removal of contextual information that may help distinguish between malignant and benign. By removing such regions, we fail to acknowledge that lesions may be dis-

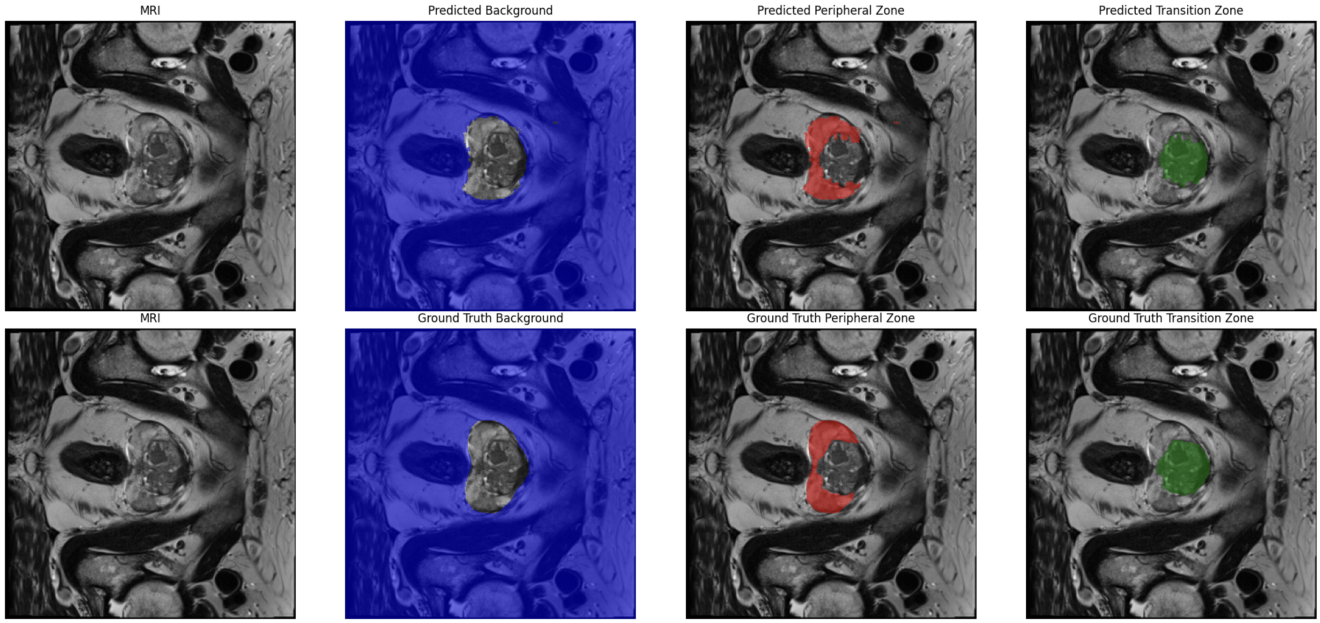

Figure 4. 2D U-Net Predicted (first row) vs. Ground Truth (second row) Segmentations. Left: MRI, Middle Left: Background, Middle Right: Peripheral Zone, Right: Transition Zone

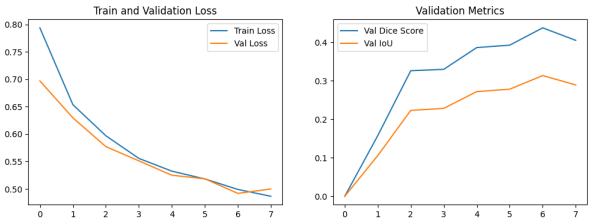

Figure 5. 2D Simple CNN

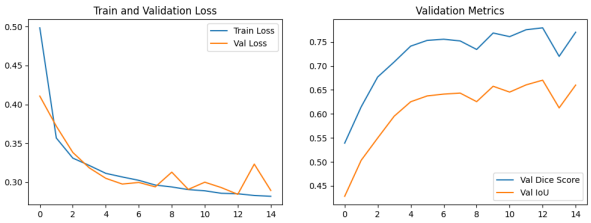

Figure 6. 2D U-Net

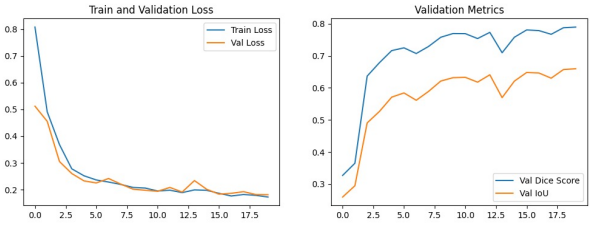

Figure 7. 3D U-Net

tinguished by their presence near structures that we mask out. In addition, when we mask out regions, we also mask out bounding boxes for lesions that may have inside the regions. Since we begin with approximately 900 images with bounding boxes for training, further decreasing the number of positive samples could lead to a biased model that mainly learns that most MRIs don't have lesions.

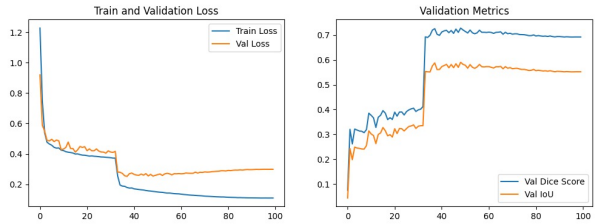

Figure 8. 3D UNETR

6. Conclusions

In this project, we aimed to segment the prostate into zones, and use these segmentations to identify lesions and classify each as benign or malignant. Because of the clinical need for more accurate lesion classification in the TZ, we emphasized accurate classification in that zone. We performed segmentation using a 2D U-Net and 3D U-Net and UNETR. The 2D U-Net performed best on this task, achieving a final Dice score of about 0.8. This is likely due to the relatively small training set, which made it difficult for this large model to learn the task.

We maintain the importance of training models specifically for TZ lesion classification. As next steps, we would obtain a larger dataset containing TZ lesions, and train it

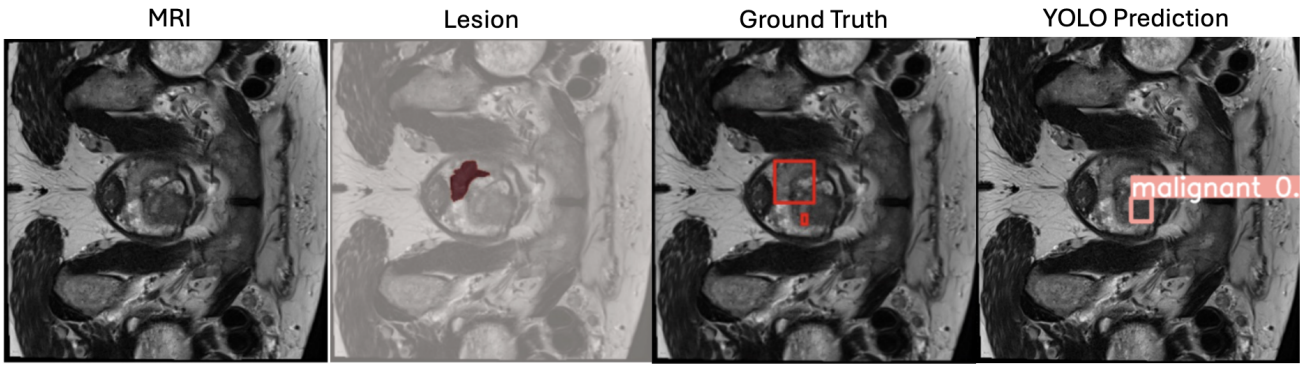

Figure 9. Best YOLOV5 Model ("Full MRI, PZ labels") Prediction

using YOLOV5. With more data, we expect to see these models outperform models trained on TZ and PZ lesions.

In addition, our 2D object detection experiment revealed that contextual information is important for lesion detection in MRIs. Given this, a 3D object detection model may better capture the complexity of lesions in the PZ and TZ regions. By incorporating volumetric information and capturing spatial relationships between MRI slices, a 3D model could effectively detect lesions and take into account lesion morphology.

7. Contributions & Acknowledgements

All group members contributed equally to project formulation, model selection, and project direction. All group members contributed equally to dataset preprocessing. Hannah worked primarily on setting up the 2D segmentation model; Abhi worked primarily on setting up the 3D segmentation model; Sam worked primarily on setting up the object detection model. All groups members contributed equally to debugging and project writeup.

We collaborated with Dr. Richard Fan in the Department of Urology at Stanford Medicine, who advised project direction and clinical needs in the field.

We made use of multiple GitHub repositories during the course of our project, all of which are cited within the appropriate section of this report.

References

[1] J. O. Barentsz, J. C. Weinreb, S. Verma, H. C. Thoeny, C. M. Tempany, F. Shtern, A. R. Padhani, D. Margolis, K. J. Macura, M. A. Haider, F. Cornud, and P. L. Choyke. Synopsis of the pi-rads v2 guidelines for multiparametric prostate magnetic resonance imaging and recommendations for use. *European Urology*, 69(1):41–49, January 2016. Epub 2015 Sep 8.

[2] R. Cao, X. Zhong, S. Shakeri, A. M. Bajgiran, S. A. Mirak, D. Enzmann, S. S. Raman, and K. Sung. Prostate cancer detection and segmentation in multi-parametric mri via cnn and conditional random field. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pages 1900–1904, 2019.

[3] Q. Chen, S. Hu, P. Long, F. Lu, Y. Shi, and Y. Li. A transfer learning approach for malignant prostate lesion detection on multiparametric mri. *Technology in Cancer Research & Treatment*, 18:1533033819858363, 2019.

[4] T. Clark, J. Yuan, J. Wong, M. A. Haider, and D. L. Langer. Fully automated segmentation of prostate whole gland and transition zone in diffusion-weighted mri using convolutional neural networks. *Journal of Medical Imaging (Bellingham, Wash.)*, 4(4):041307, 2017.

[5] CVLab Stony Brook. Selfmedmae. `https://github.com/cvlab-stonybrook/SelfMedMAE`. Accessed: June 4, 2024.

[6] A. Duran, G. Dussert, O. Rouvière, T. Jaouen, P.-M. Jodoin, and C. Lartizien. Prostattention-net: A deep attention model for prostate cancer segmentation by aggressiveness in mri scans. *Medical Image Analysis*, 77:102347, April 2022.

[7] C. Gaudiano, L. Bianchi, B. Corcioni, et al. Evaluating the performance of clinical and radiological data in predicting prostate cancer in prostate imaging reporting and data system version 2.1 category 3 lesions of the peripheral and the transition zones. *International Urology and Nephrology*, 54:263–271, 2022.

[8] gphdotam. Unet3d, 2021. Accessed: 2024-05-13.

[9] C. A. Hamm, G. L. Baumgärtner, F. Biessmann, N. L. Beetz, A. Hartenstein, L. J. Savic, K. Froböse, F. Dräger, S. Schallenberg, M. Rudolph, A. D. J. Baur, B. Hamm, M. Haas, S. Hofbauer, H. Cash, and T. Penzkofer. Interactive explainable deep learning model informs prostate cancer diagnosis at mri. *Radiology*, 307(4):e222276, May 2023.

[10] M. R. Hassan, M. F. Islam, M. Z. Uddin, G. Ghoshal, M. M. Hassan, S. Huda, and G. Fortino. Prostate cancer classification from ultrasound and mri images using deep learning based explainable artificial intelligence. *Future Generation Computer Systems*, 127:462–472, 2022.

[11] A. Hatamizadeh, Y. Tang, V. Nath, D. Yang, A. Myronenko, B. Landman, H. Roth, and D. Xu. Unetr: Transformers for 3d medical image segmentation, 2021.

[12] Y. Li, J. Wang, M. Hu, P. Patel, H. Mao, T. Liu, and X. Yang. Prostate gleason score prediction via mri using capsule net-

work. In *Medical Imaging 2023: Computer-Aided Diagnosis*, volume 12465, page 1246523, April 7 2023.

[13] S. Mehralivand, D. Yang, S. A. Harmon, D. Xu, Z. Xu, H. Roth, S. Masoudi, D. Kesani, N. Lay, M. J. Merino, B. J. Wood, P. A. Pinto, P. L. Choyke, and B. Turkbey. Deep learning-based artificial intelligence for prostate cancer detection at biparametric mri. *Abdominal Radiology (NY)*, 47(4):1425–1434, April 2022. Epub 2022 Jan 31.

[14] milesial. Pytorch-unet. `https://github.com/milesial/Pytorch-UNet/tree/master`. Accessed: June 3, 2024.

[15] O. J. Pellicer-Valero, J. L. Marenco Jiménez, V. Gonzalez-Perez, J. L. Casanova Ramón-Borja, I. Martín García, M. Barrios Benito, P. Pelechano Gómez, J. Rubio-Briones, M. J. Rupérez, and J. D. Martín-Guerrero. Deep learning for fully automatic detection, segmentation, and gleason grade estimation of prostate cancer in multiparametric magnetic resonance images. *Scientific Reports*, 12(1):2975, February 22 2022.

[16] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection, 2016.

[17] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[18] J. Sanyal, I. Banerjee, L. Hahn, and D. Rubin. An automated two-step pipeline for aggressive prostate lesion detection from multi-parametric mr sequence. *AMIA Joint Summits on Translational Science Proceedings*, 2020:552–560, May 30 2020.

[19] J. Simeth, J. Jiang, A. Nosov, A. Wibmer, M. Zelefsky, N. Tyagi, and H. Veeraraghavan. Deep learning-based dominant index lesion segmentation for mr-guided radiation therapy of prostate cancer. *Medical Physics*, 50(8):4854–4870, August 2023.

[20] Ultralytics. Yolov5. `https://github.com/ultralytics/yolov5/tree/c0380fd85549c8b315bb1fb2f423f9297e9744ec`. June 2, 2024.

[21] C. Vente, P. Vos, M. Hosseinzadeh, J. Pluim, and M. Veta. Deep learning regression for prostate cancer detection and grading in bi-parametric mri. *IEEE Transactions on Biomedical Engineering*, 68(2):374–383, February 2021. Epub 2021 Jan 20.

[22] X. Yu and et al. Deep attentive panoptic model for prostate cancer detection using biparametric mri scans. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, volume 12264 of *Lecture Notes in Computer Science*. Springer, 2020.