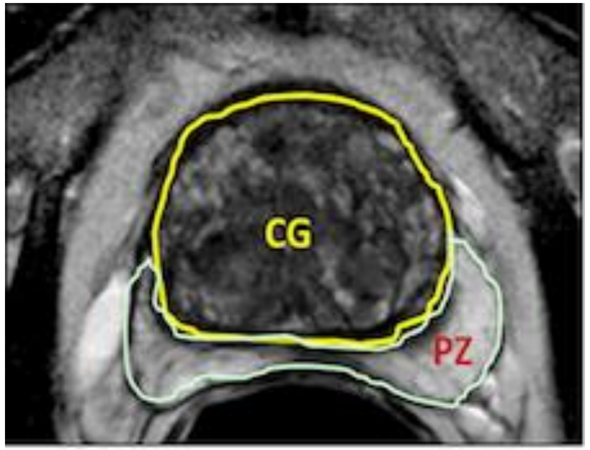


Medical Image Segmentation

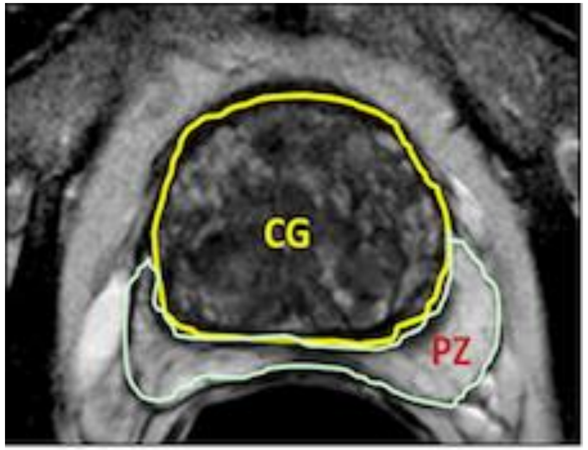
GEORGIA SOUSOURI

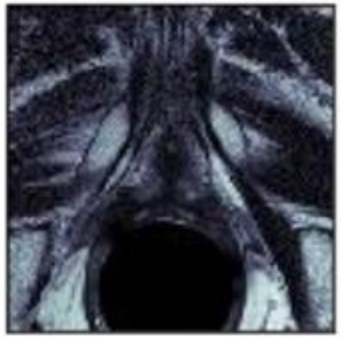
RISHABH SINGH

JANNIK GUT

Problem statement

- MRIs of prostates
- Segment Central Gland (2) and Peripheral Zone (1) from Background (0)
- The MRIs might have different orientations





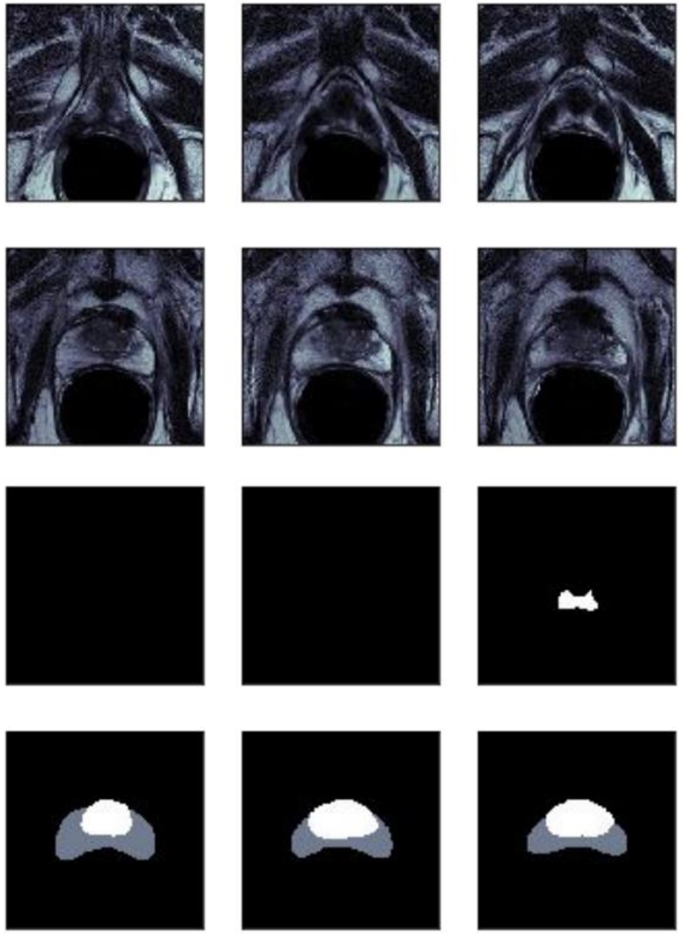
Data

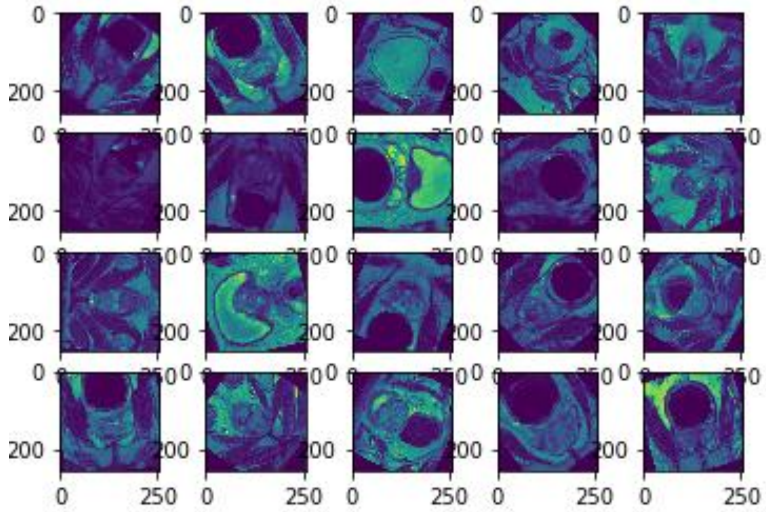
50 (10) subjects

~15 depths each

256x256 black and white images

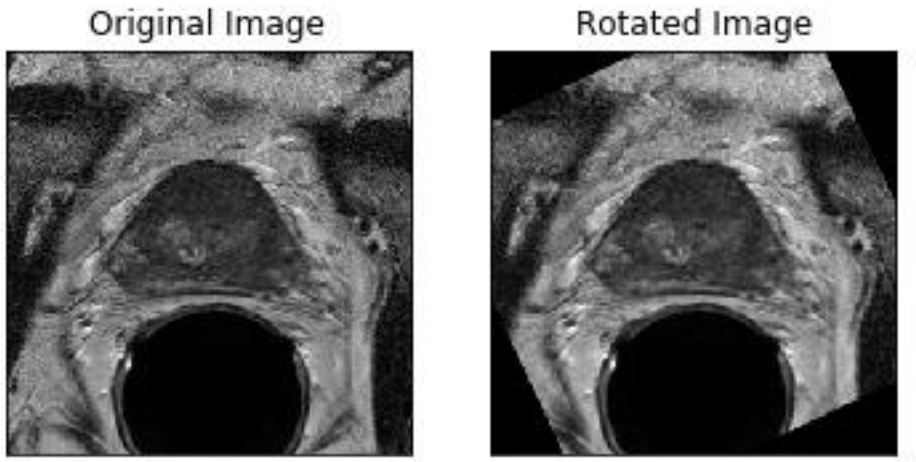
Normalize "lighting" from 0 to 1 for each image

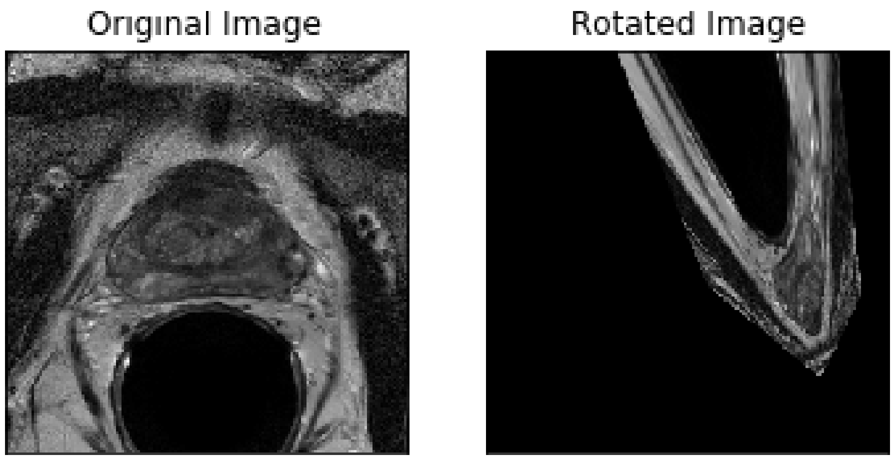




Computer Vision

- ORB features (similar to SIFT)
- Bruteforce Hamming Descriptor Matcher
- Take homography (ROTATION and translation) which is closest to training data





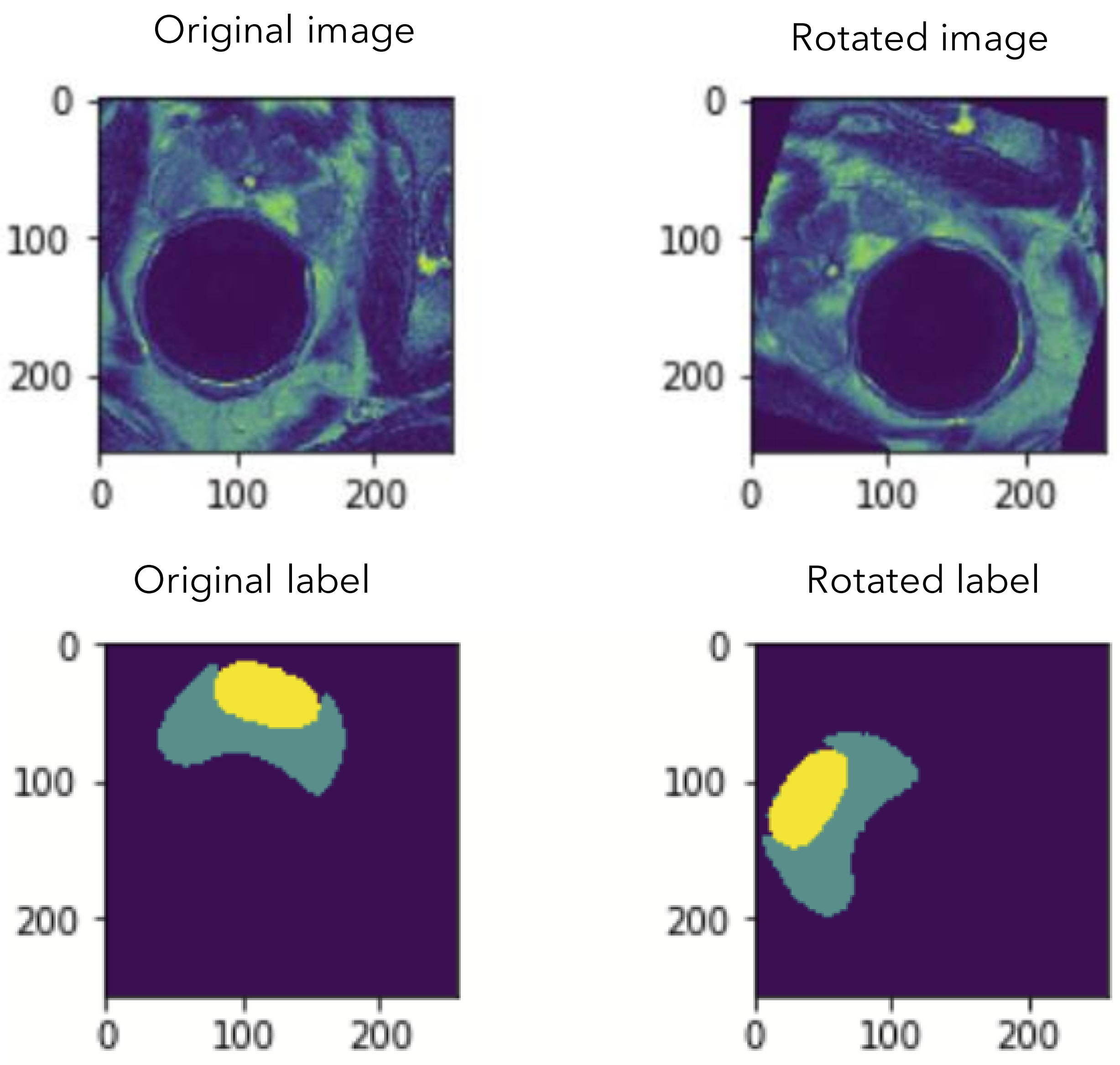
Data augmentation

Rotation invariant model

- ✓ Random rotation: 0°-360°
- ✓ Zero-padded edges
- ✓ Normalize image values (0-1)

20 rotations/image (842x20=16840 images)

80/20 split to train and test set



U-NET

ENCODER:

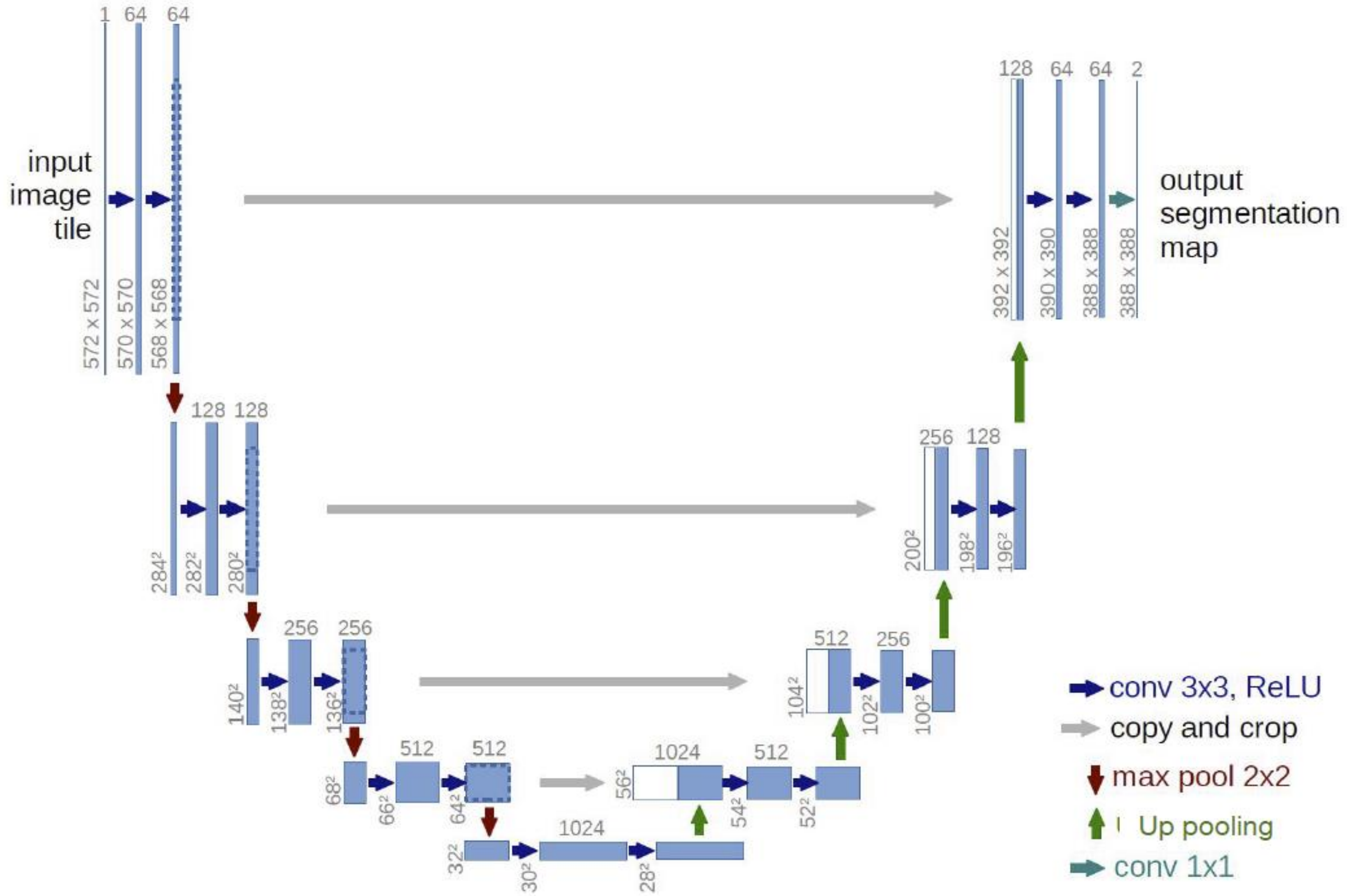
- ✓ Downsampling
- ✓ Convolutional and max pooling operations
- ✓ Captures the context in the image
- ✓ HR image --> LR image

DECODER:

- ✓ Upsampling
- ✓ Transposed convolution
- ✓ Skip connections
- ✓ LR image --> HR image

ENCODER
Contraction path

DECODER
Expansion path



Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015.

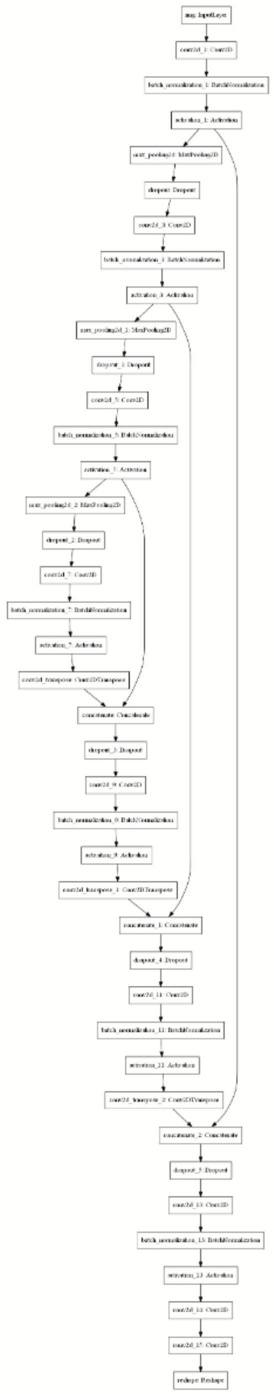
Decision function and training

- Variating hyperparameters:
  1) network depth (3, 5, 7)
  2) number of filters (16, 32)
     (training time limitations)

- Modified decision function to align with 3-class classification task using *softmax:*
  Two 2D convolutional layers

- Vectorized output by reshaping (256x256, 3)

- Training:
  1) Adam optimizer
  2) Sparse categorical crossentropy
  3) Weighted classes based on the occurrence of pixels in each class



Loss Functions

1. Categorical Cross Entropy loss
   - True class represented as a one-hot encoded vector, and the closer the model's outputs are to that vector, the lower the loss.
2. Focal loss
   - Down-weight the contribution of easy examples so that the CNN focuses more on hard examples
3. Jaccard loss
   - Similar to IoU loss, useful for unbalanced number of pixels in each class for semantic segmentation.
4. Dice loss
   - Measure of overlap between sets, considers loss both locally and globally

$$L(gt, pr) = -gt \cdot \log(pr) \tag{1}$$

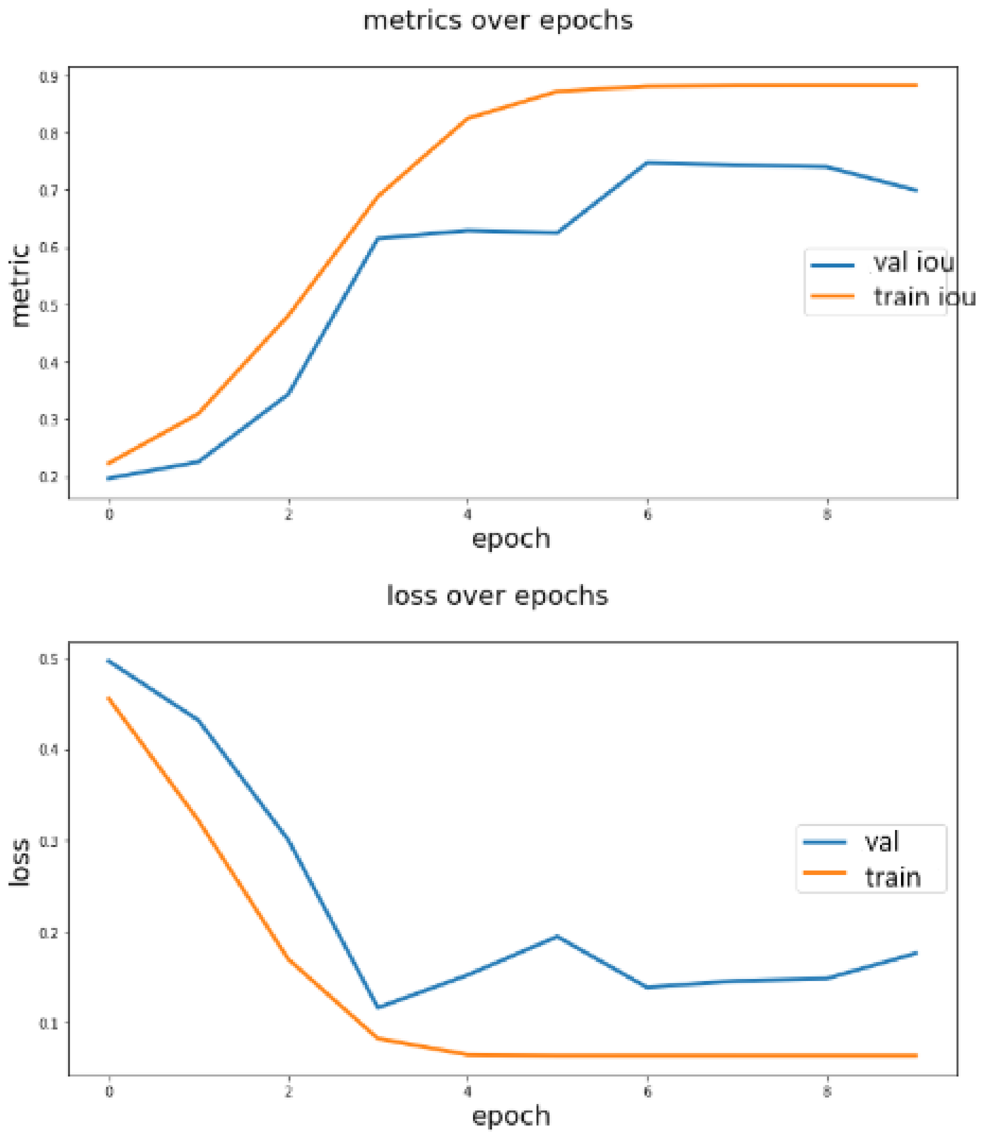
$$L(gt, pr) = -gt \cdot \alpha \cdot (1 - pr)^{\gamma} \cdot \log(pr) \tag{2}$$

$$L(gt, pr) = 1 - \frac{gt \cap pr}{gt \cup pr} \tag{3}$$

$$L(precision, recall) = 1 - (1 + \beta^2) \frac{precision \cdot recall}{\beta^2 \cdot precision + recall} \tag{4}$$

Validation

- Analyzed the trend of training and validation loss with epochs
- Curve flattens quickly after 5 epochs
- Early stopping reduces over-fitting



Validation Results

- Grid search for tuning hyper-parameters
- Depth of network & Number of filters varied, rest kept constant
- Adding class-specific weights led to an improved performance
- Best results for a depth of 7 and number of filters 32

TABLE I: Evaluation metrics from grid search on test data from augmented dataset. The names in the first column indicate the depth of the U-Net (3, 5 or 7) and the number of filters (16 or 32). The w/nw indices refer to training implemented with class weights (w) or not (nw). In the parenthesis there is the last part of the name of the notebook you may find the respective iteration (first part is "Train_Evaluate_Unet_").

Depth, Filters, w/nw (file)	Overall precision	Per-class precision	IoU
3, 16, w (Iter_1_w)	0.95	0.72	0.49
3, 16, nw (Iter_1_nw)	0.95	0.69	0.49
3, 32, nw (Iter_3_nw)	0.96	0.75	0.60
5, 16, nw (Iter_2_nw)	0.93	0.75	0.50
5, 32, w (Iter_4_w)	0.97	**0.86**	0.67
5, 32, nw (Iter_4_nw)	0.96	0.57	0.50
7, 32, nw (Iter_5_nw)	**0.98**	**0.86**	**0.71**

Test Results

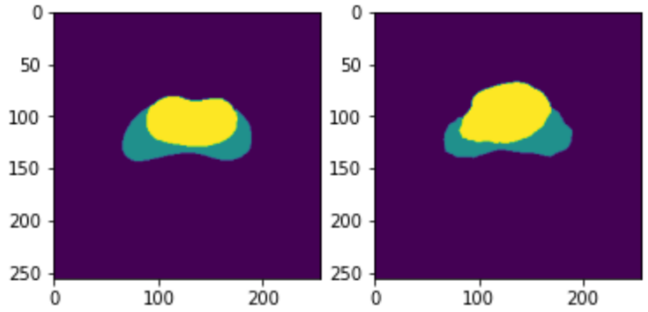
Best model obtained through validation

Performance on aligned and rotated sets similar

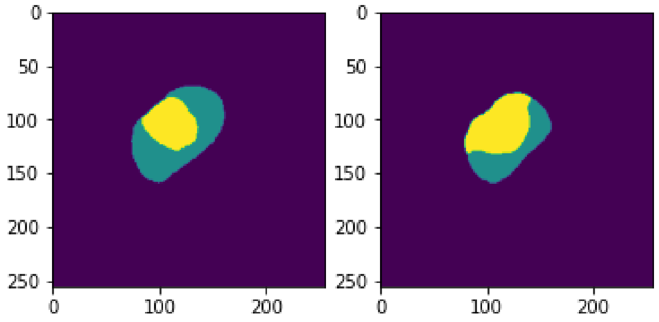
Model robust to rotation

TABLE II: Evaluation metrics from final (7,32, nw) U-Net on provided test set

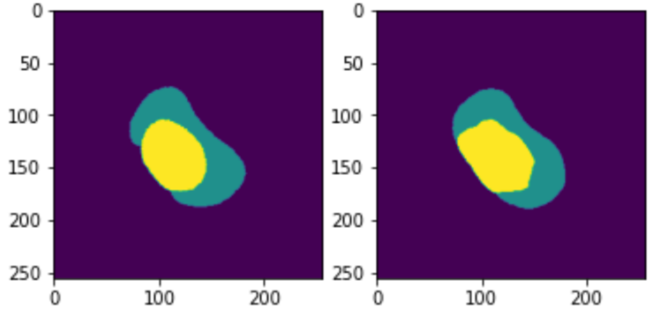
	Aligned data	Randomly rotated data
Overall precision	0.95	**0.96**
Per-class precision	**0.82**	0.80
IoU	0.54	**0.57**



Aligned Test Set



Rotated Test Set

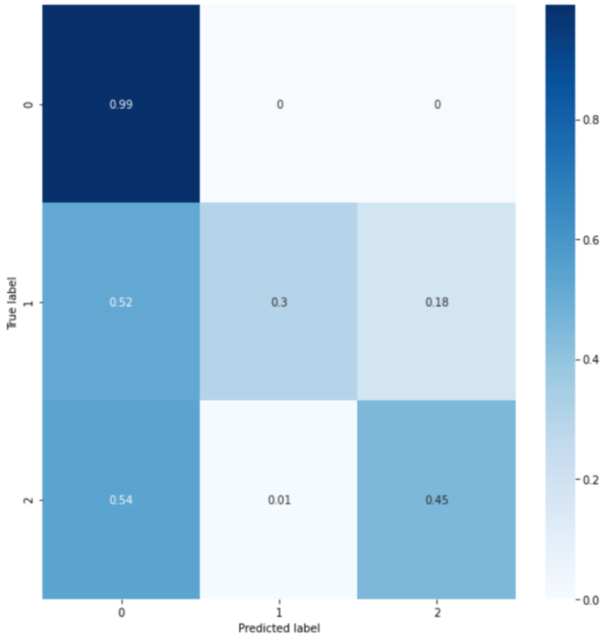


Augmented Validation Set

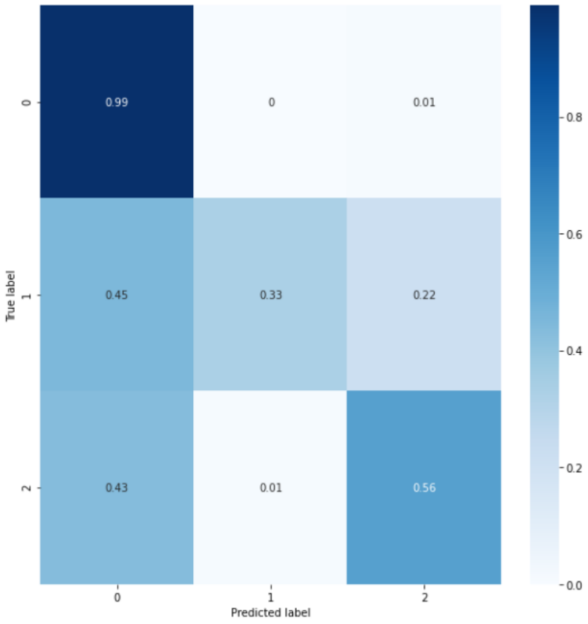
Results

Pairs of ground truth (left) and predicted (right) labels for the aligned test, rotated test and augmented validation sets. The model has consistent performance across three sets we used for evaluation.

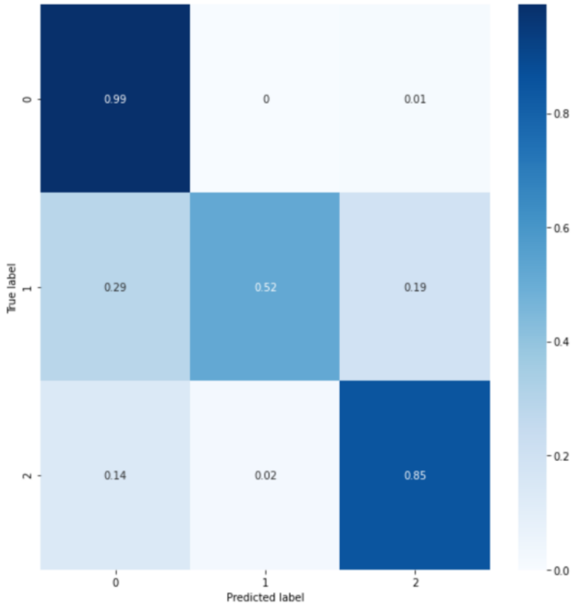
Results: Confusion Matrix



Aligned Test Set



Rotated Test Set



Augmented Validation Set

Discussion

Labels in sparse categorical format performed better than one-hot encoded

Rotation invariance through augmenting training set with random rotations

Given problem statement has high class imbalance

Weights to classes during training improves overall performance

Increasing depth U-Net improved per class precision

More filters improved performance

Class 1 often misclassified at discontinuities, calls for multi-modal segmentation

References

Data

Computer Vision

Losses for Image Segmentation

Dice Loss

Keras Segmentation Models

Project GitHub

Thank you

QUESTION?