

U-Net Models

3 models for the segmentation task:

- **Classic U-Net:** *baseline U-Net model architecture*

U-Net Models

3 models for the segmentation task:

- **Classic U-Net**: *baseline U-Net model architecture*
- **Improved U-Net**: *small improvements, fewer parameters*

U-Net Models

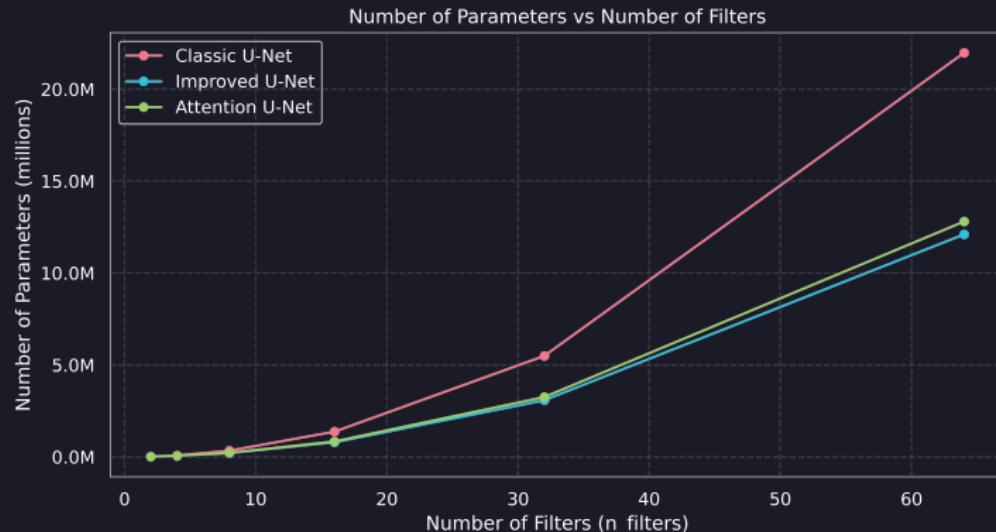
3 models for the segmentation task:

- **Classic U-Net**: *baseline U-Net model architecture*
- **Improved U-Net**: *small improvements, fewer parameters*
- **Attention U-Net**: *attention mechanism added*

U-Net Models

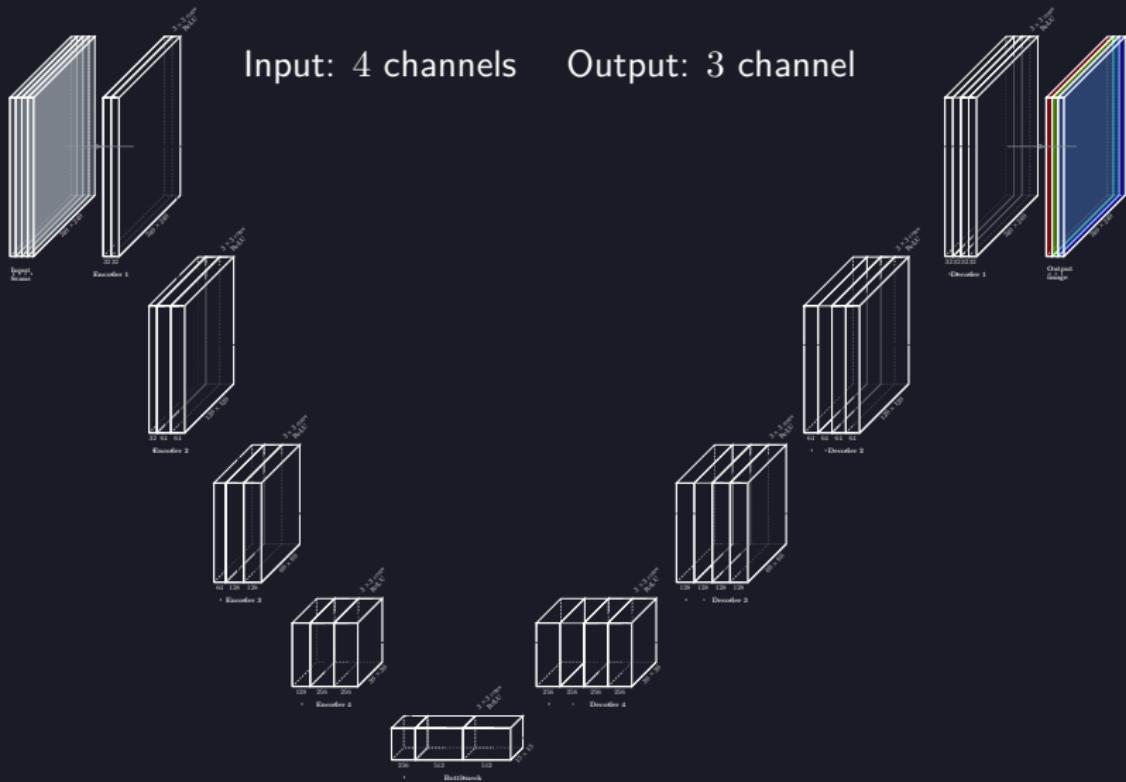
3 models for the segmentation task:

- **Classic U-Net**: *baseline U-Net model architecture*
- **Improved U-Net**: *small improvements, fewer parameters*
- **Attention U-Net**: *attention mechanism added*



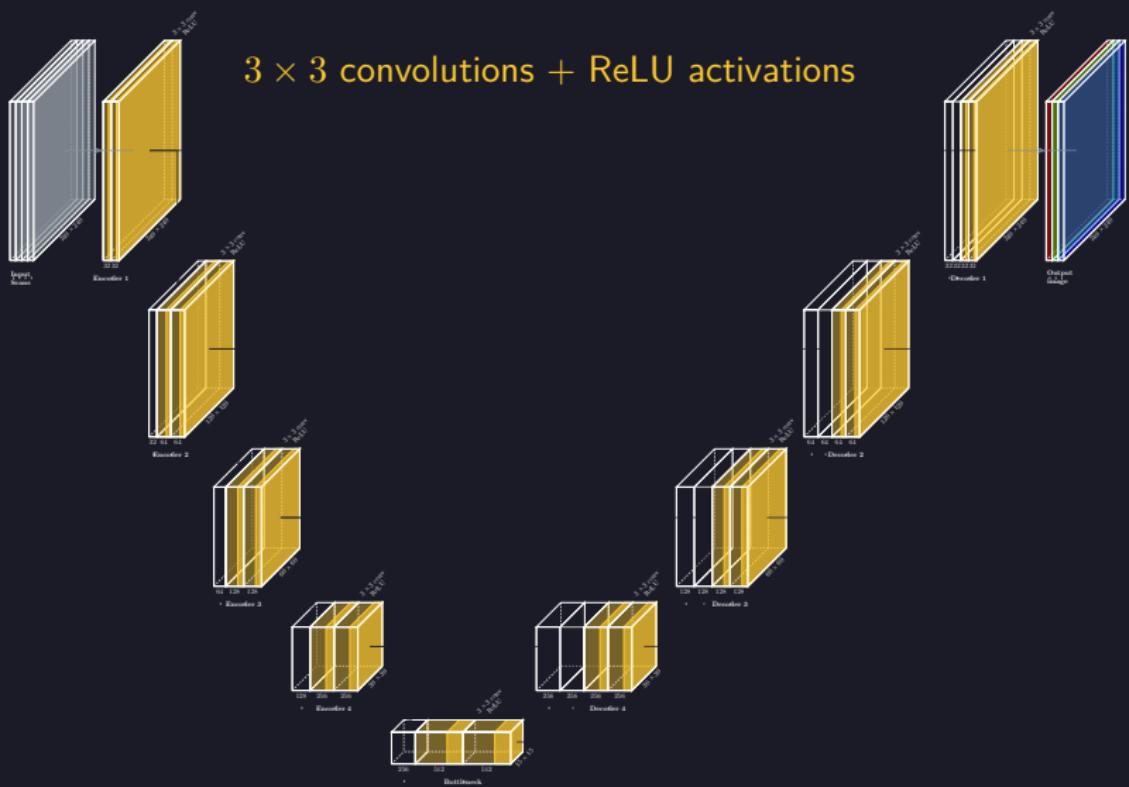
Segmentation

Classic U-Net



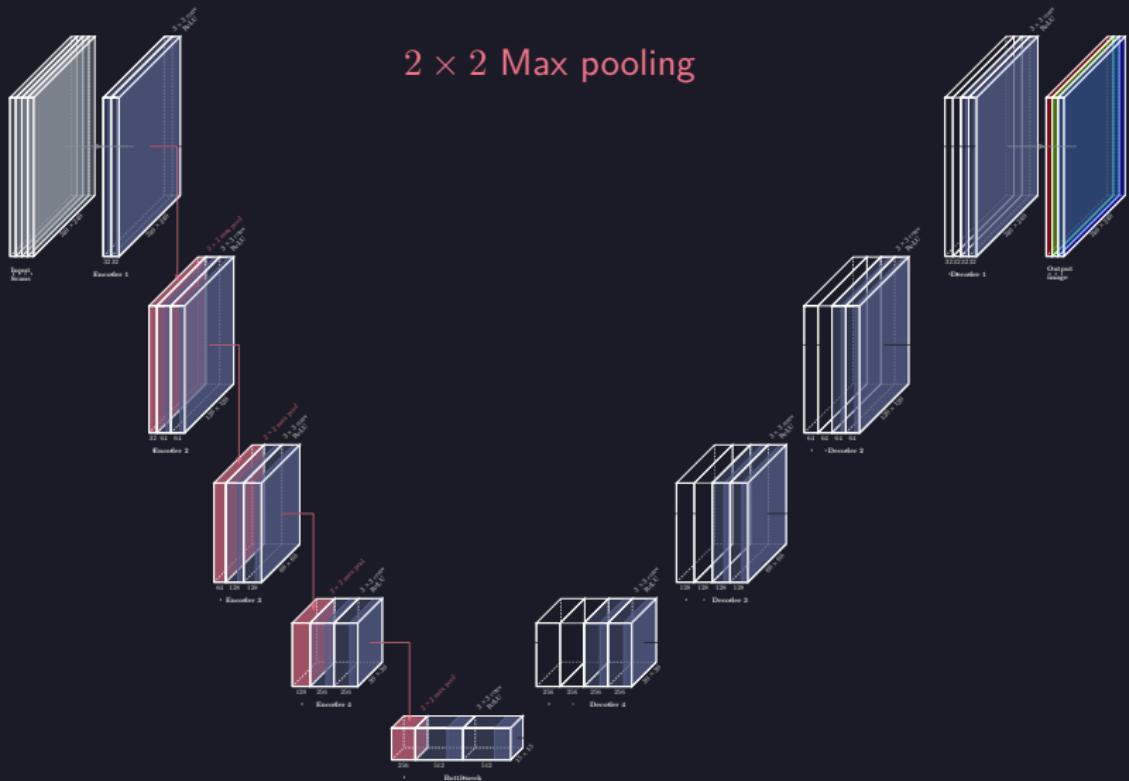
Segmentation

Classic U-Net



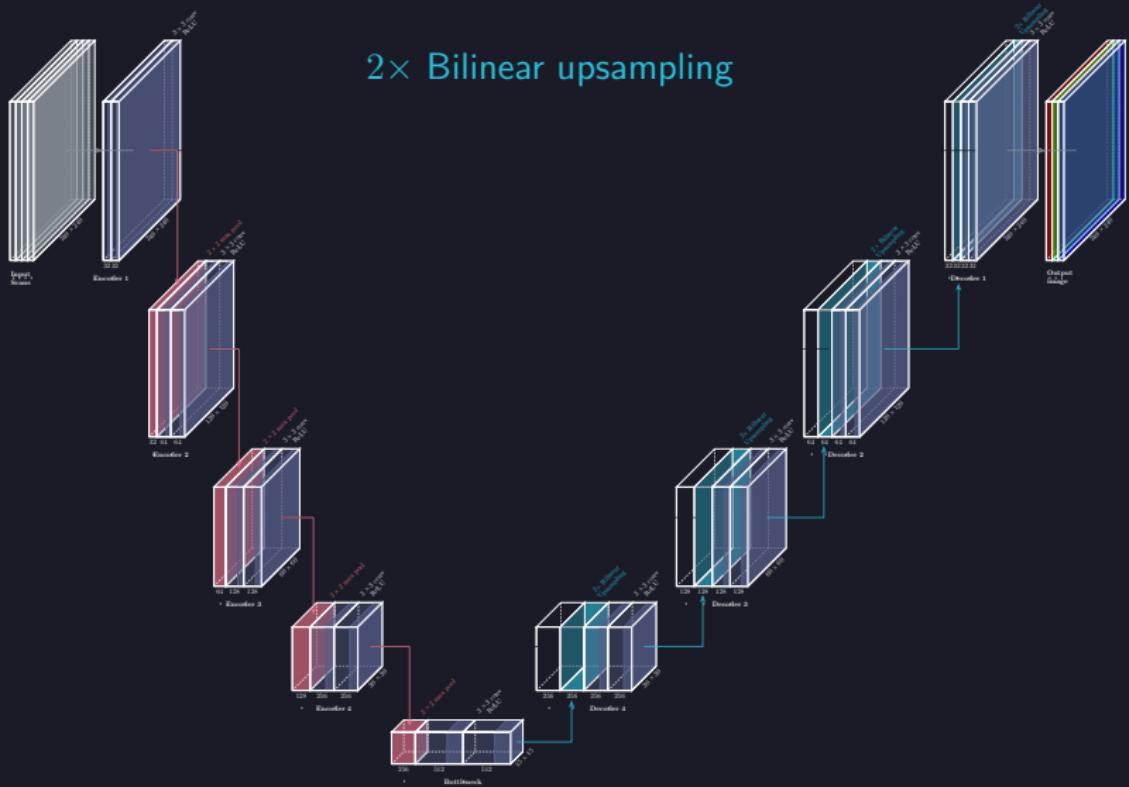
Segmentation

Classic U-Net



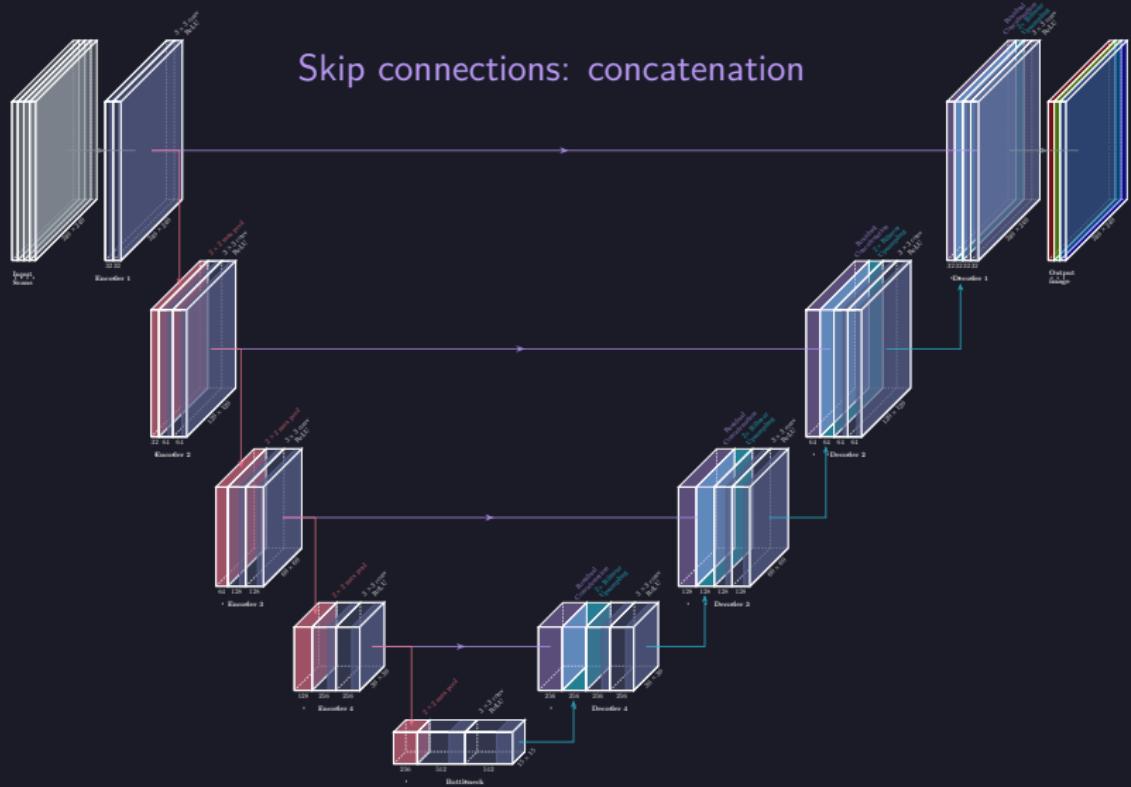
Segmentation

Classic U-Net



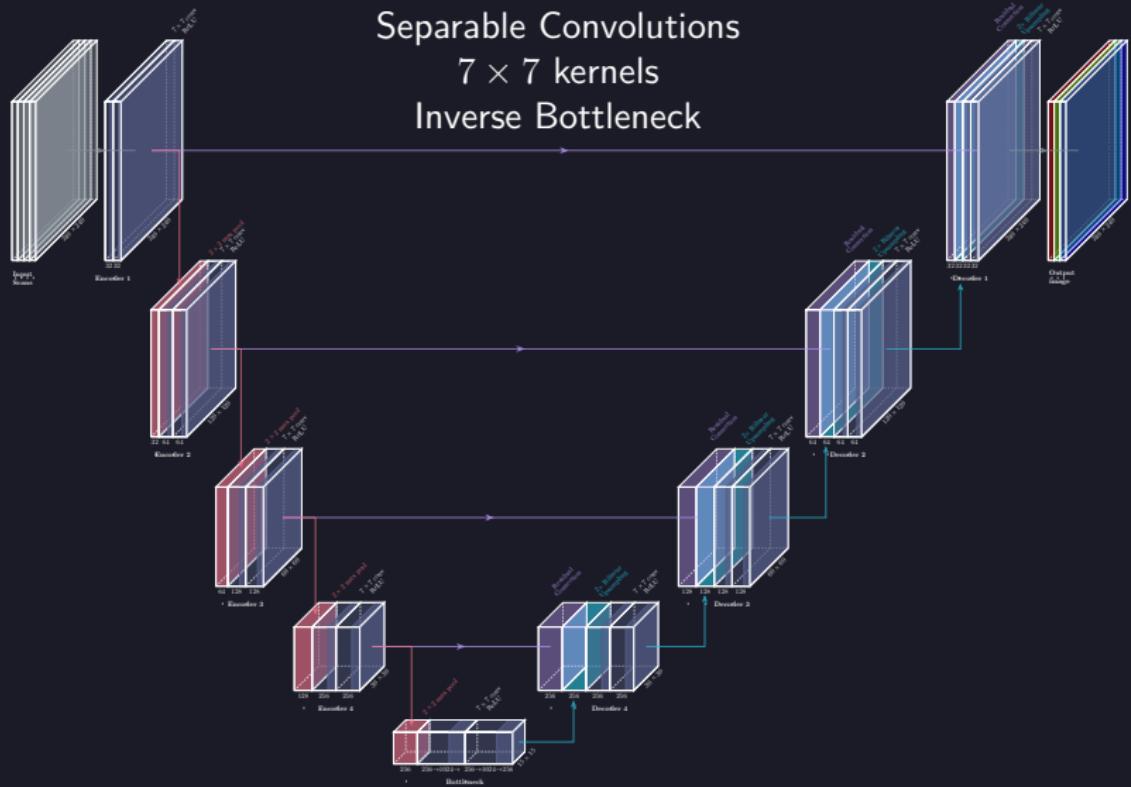
Segmentation

Classic U-Net



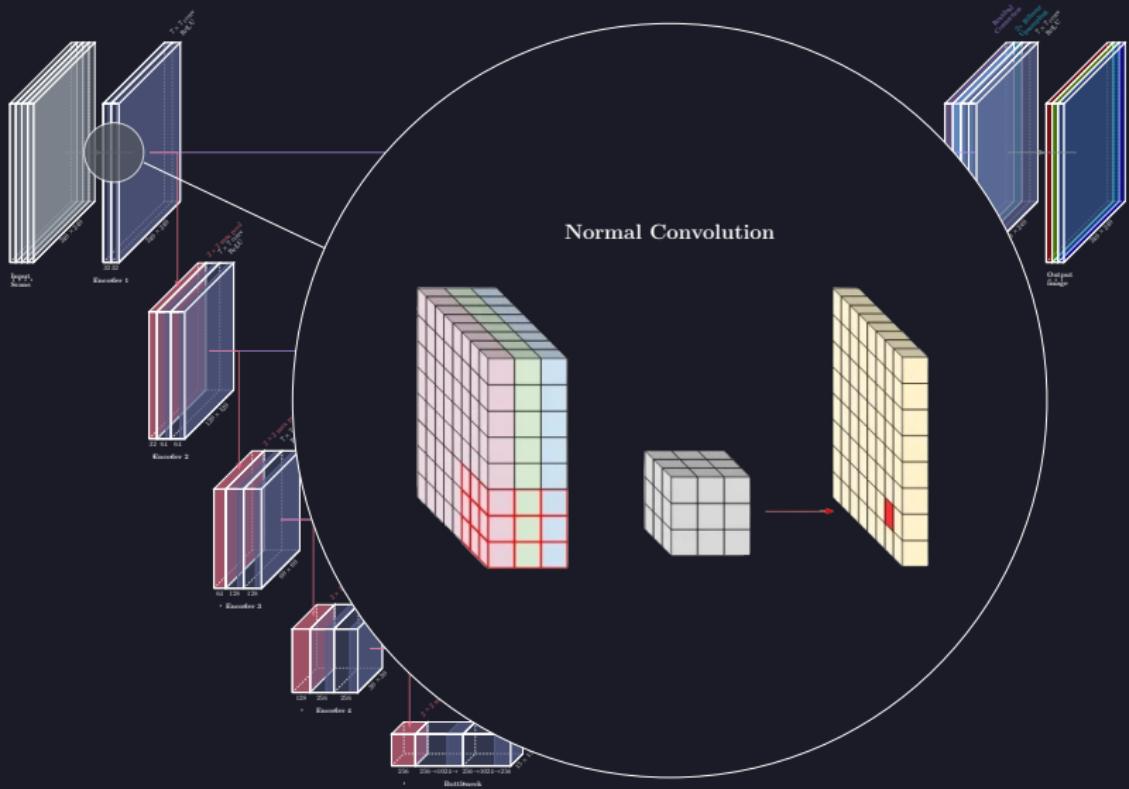
Segmentation

Improved U-Net



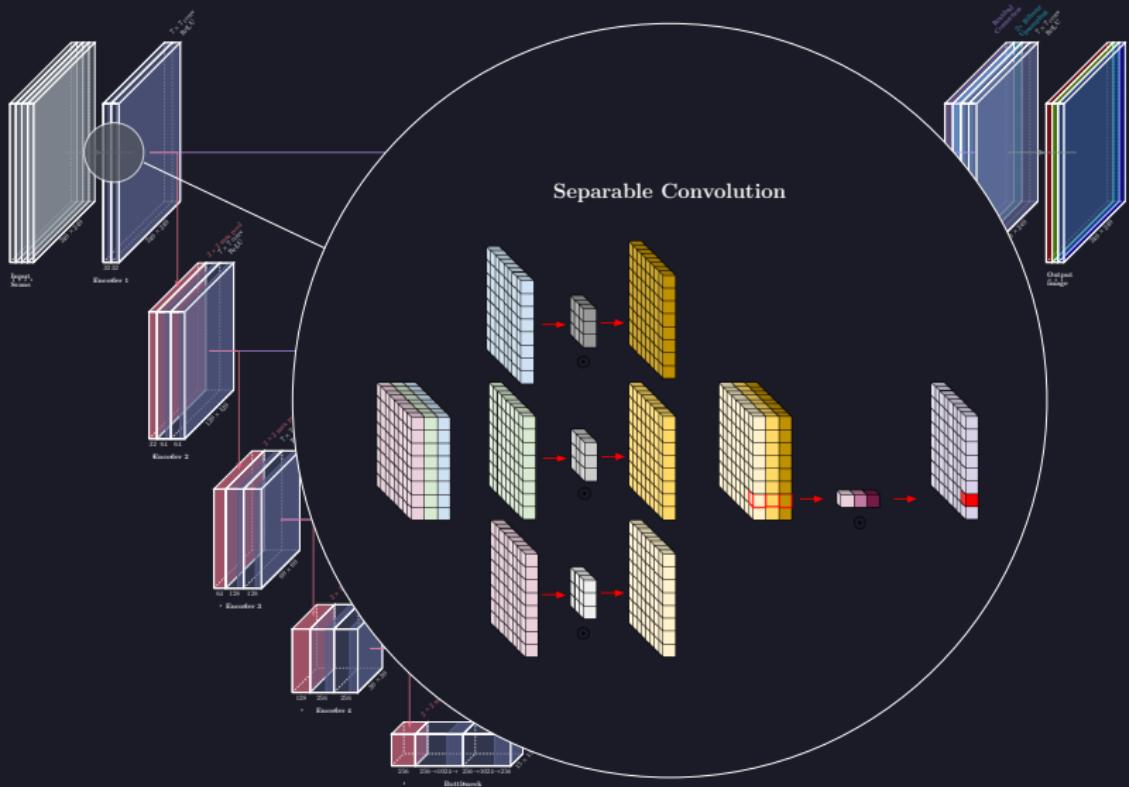
Segmentation

Improved U-Net



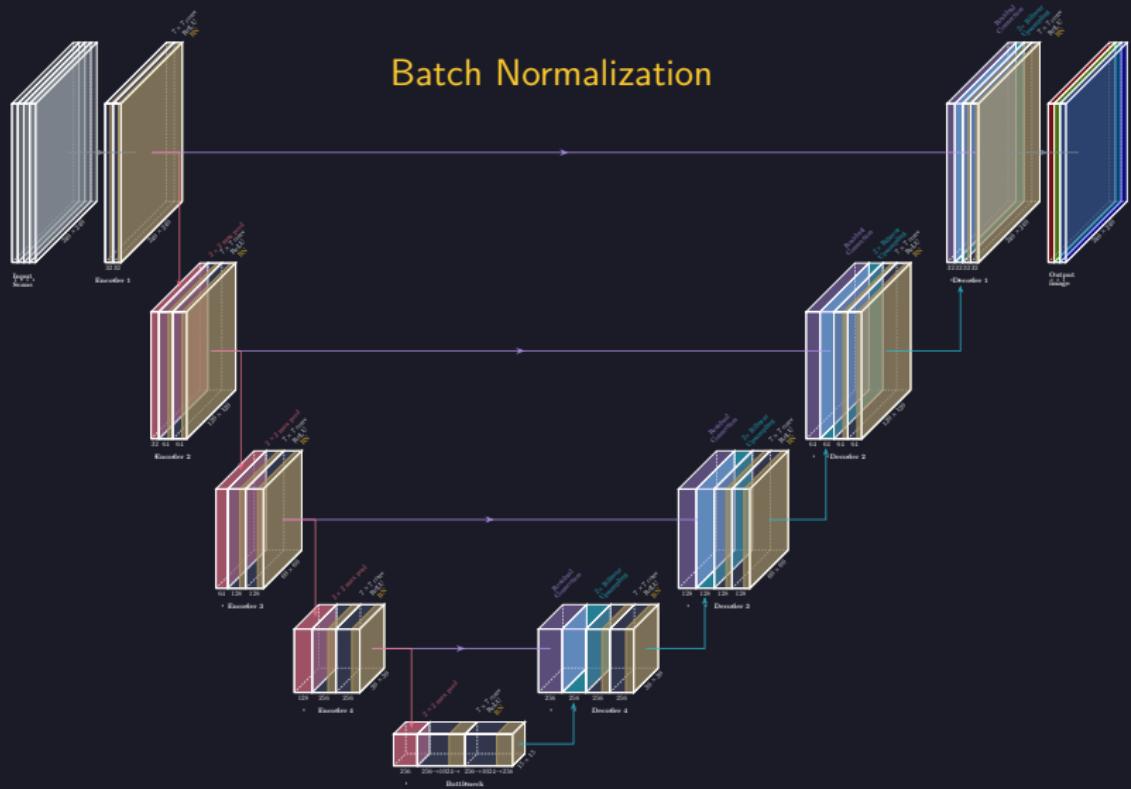
Segmentation

Improved U-Net



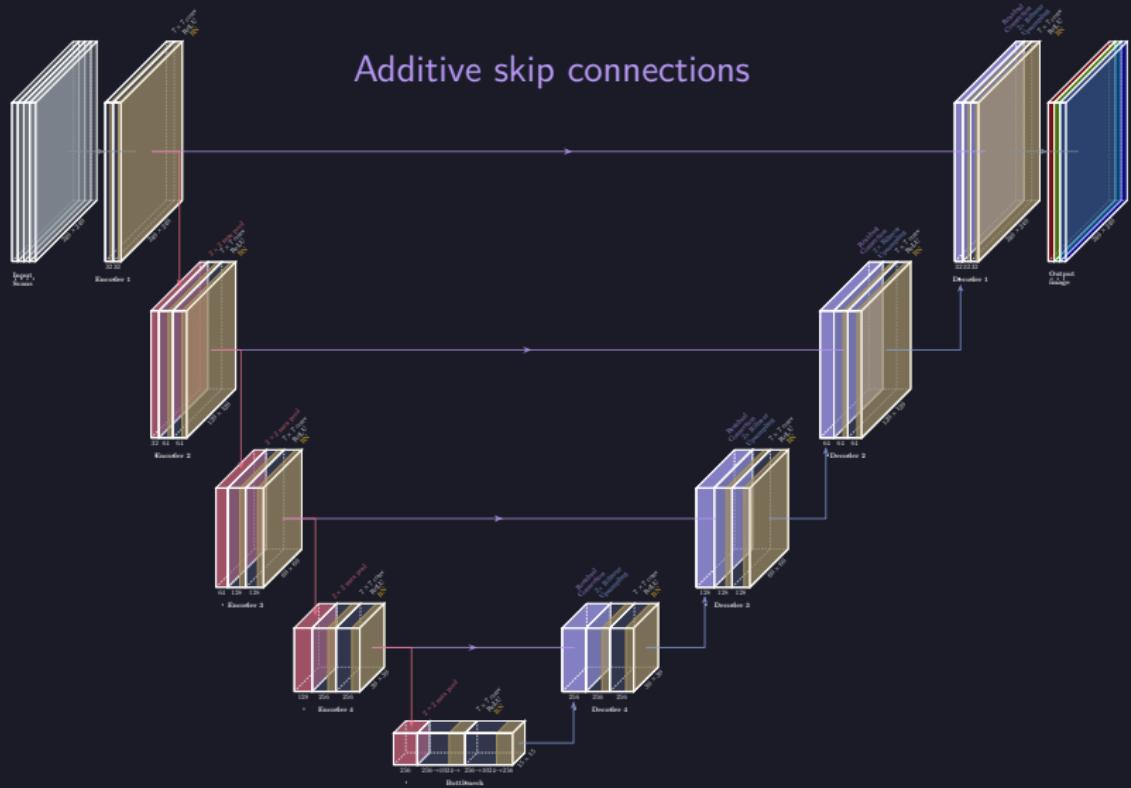
Segmentation

Improved U-Net



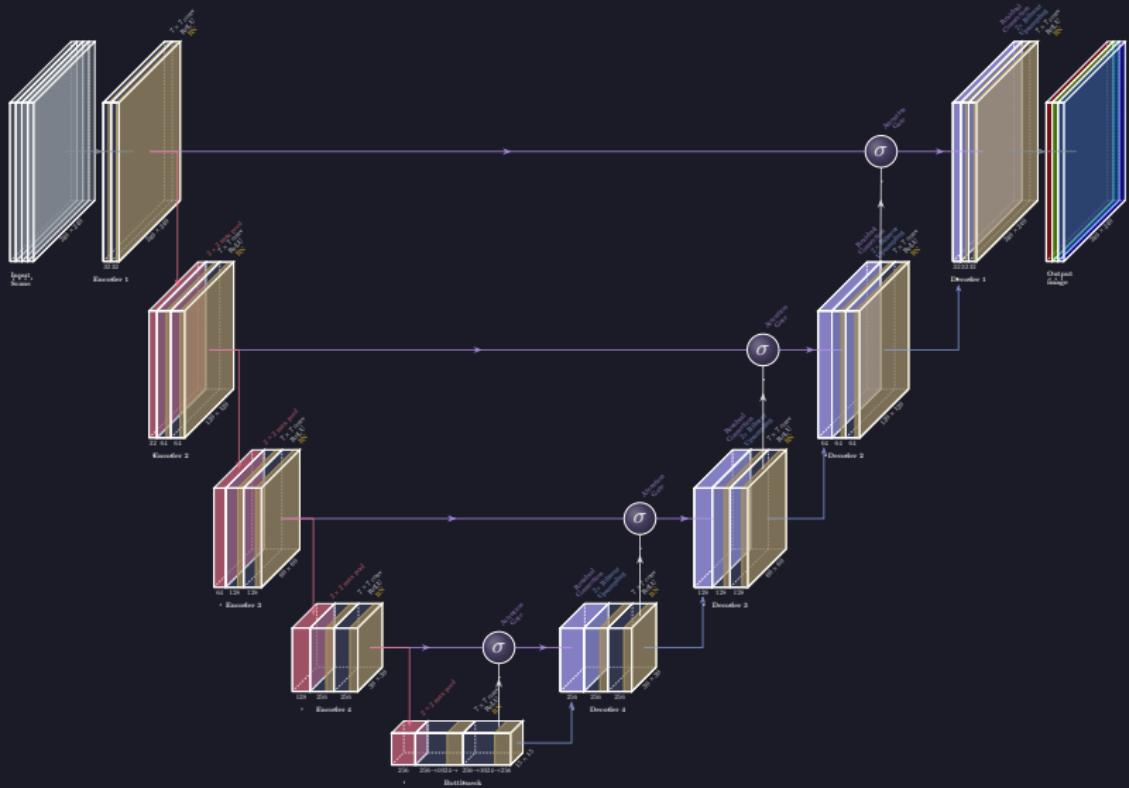
Segmentation

Improved U-Net



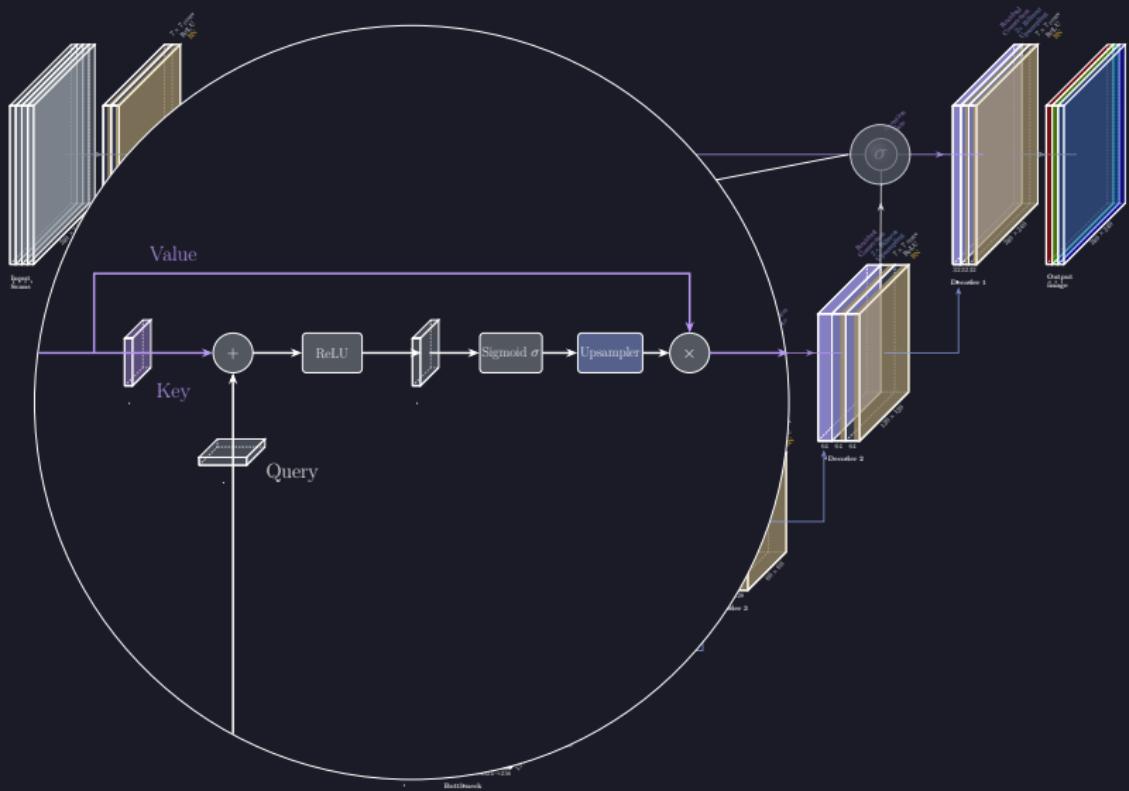
Segmentation

Attention U-Net



Segmentation

Attention U-Net



Training Details

U-Net Models training parameters:

- **epochs:** 20
- **Optimizer:** Adam (with weight decay (1×10^{-2}))
- **Scheduler:** Exponential Decay ($\gamma = 0.9$)
- **Loss Function:** BCE with Logits Loss:

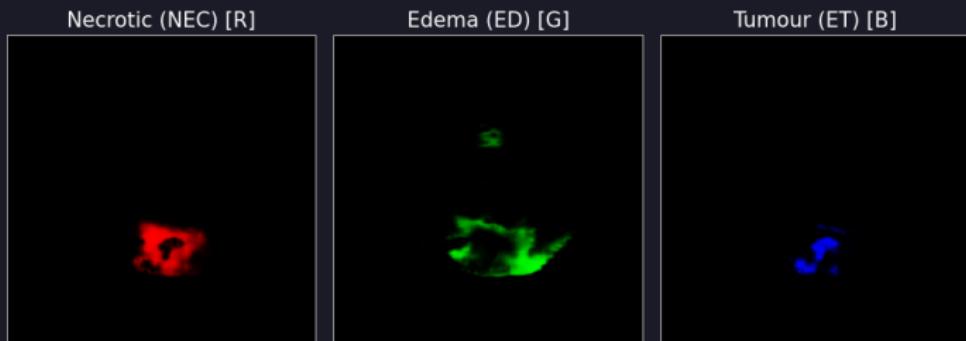
$$\ell(y, \hat{y}) = -[y \log(\sigma(\hat{y})) + (1 - y) \log(1 - \sigma(\hat{y}))]$$

- **learning rate:** 2×10^{-3}
- **batch size:** 32 (both training and validation)
- **image size:** 240×240
- **first encoder filters:** 32

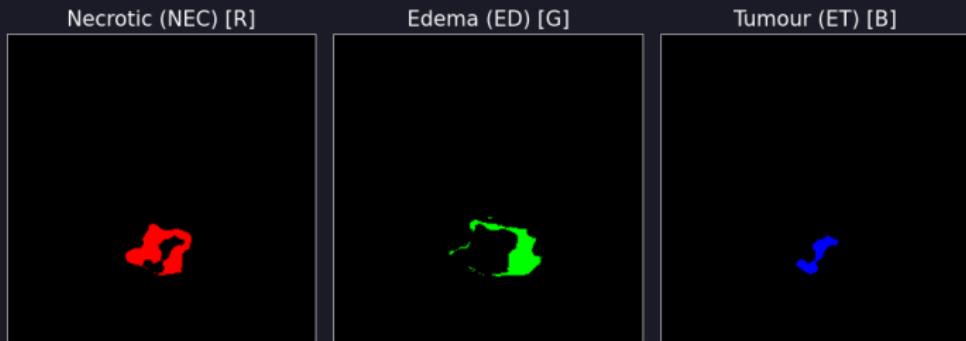
Segmentation

Visualizing a prediction

Predicted Mask Channels [RGB]



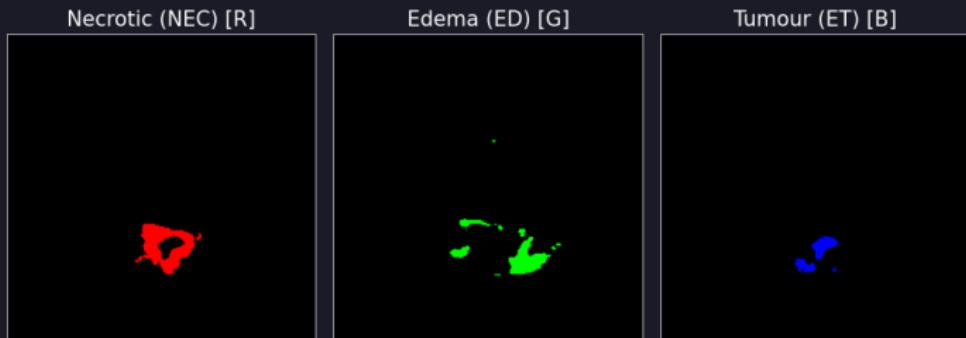
Ground Truth Mask Channels [RGB]



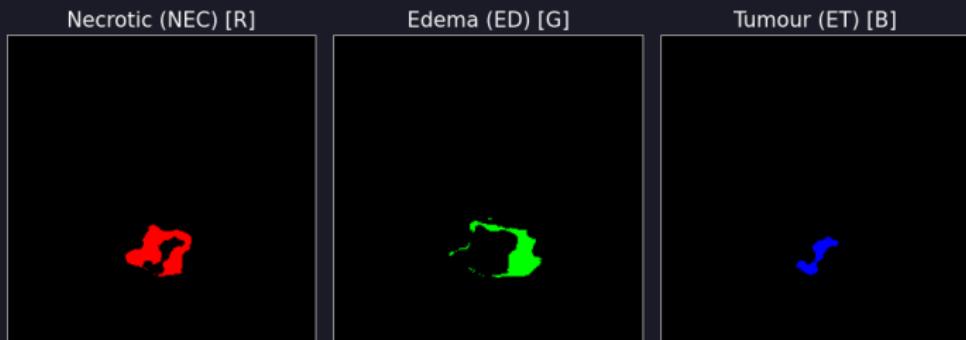
Segmentation

Visualizing a prediction

Binarized Predicted Mask Channels [RGB]



Ground Truth Mask Channels [RGB]



Segmentation

Performance Assessment

$$\text{Dice} = \frac{2 \times |X \cap Y|}{|X| + |Y|}$$

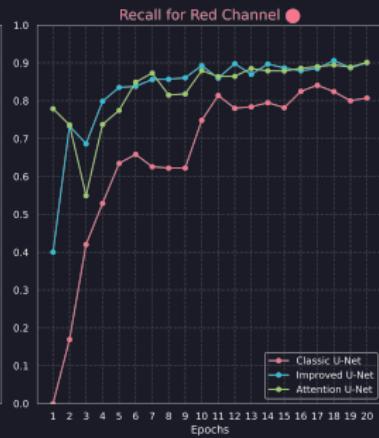
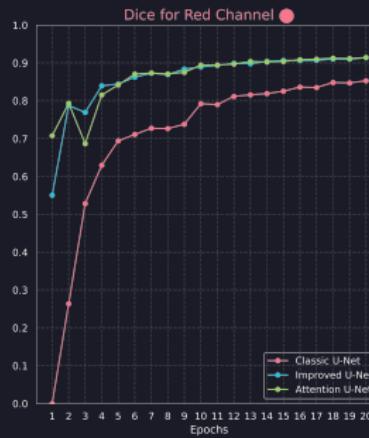
Dice Coefficient
"overlap" metric

$$\text{Precision} = \frac{TP}{TP + FP}$$

Precision
prediction quality

$$\text{Recall} = \frac{TP}{TP + FN}$$

Recall
prediction quantity



Segmentation

Performance Assessment

$$\text{Dice} = \frac{2 \times |X \cap Y|}{|X| + |Y|}$$

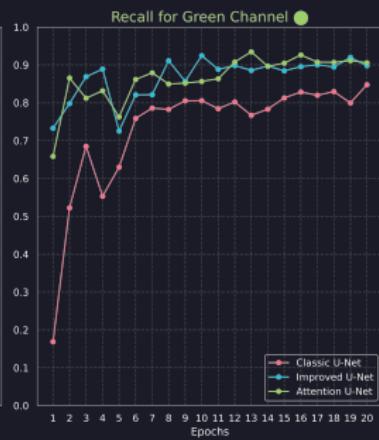
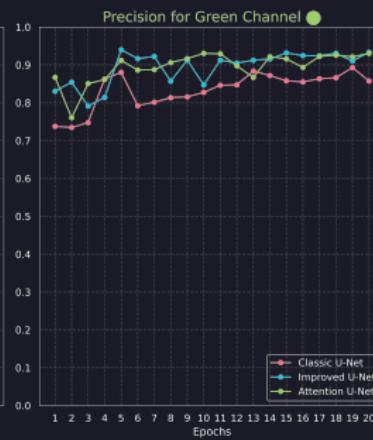
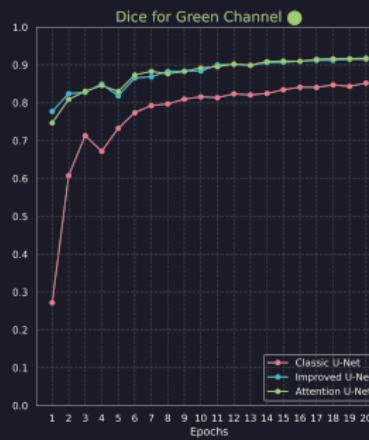
Dice Coefficient
"overlap" metric

$$\text{Precision} = \frac{TP}{TP + FP}$$

Precision
prediction quality

$$\text{Recall} = \frac{TP}{TP + FN}$$

Recall
prediction quantity



Segmentation

Performance Assessment

$$\text{Dice} = \frac{2 \times |X \cap Y|}{|X| + |Y|}$$

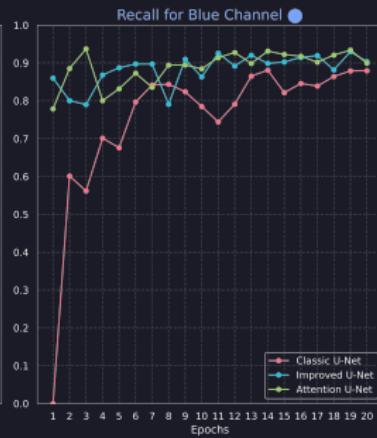
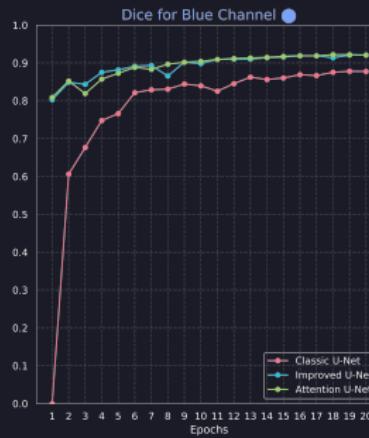
Dice Coefficient
"overlap" metric

$$\text{Precision} = \frac{TP}{TP + FP}$$

Precision
prediction quality

$$\text{Recall} = \frac{TP}{TP + FN}$$

Recall
prediction quantity



Segmentation

Performance Assessment

$$\text{Dice} = \frac{2 \times |X \cap Y|}{|X| + |Y|}$$

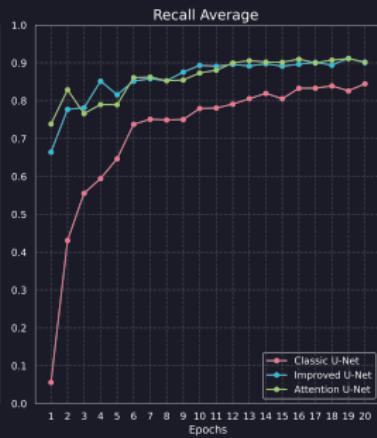
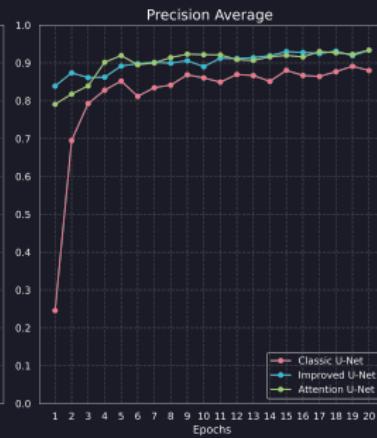
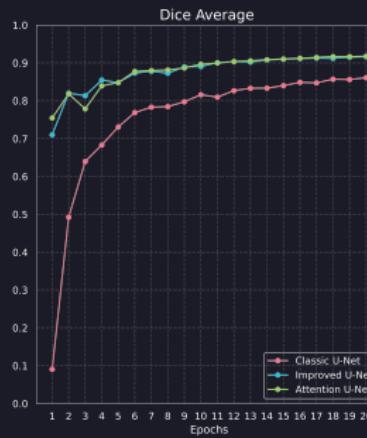
Dice Coefficient
“overlap” metric

$$\text{Precision} = \frac{TP}{TP + FP}$$

Precision
prediction quality

$$\text{Recall} = \frac{TP}{TP + FN}$$

Recall
prediction quantity



Segmentation

Visualizing Attention Maps

