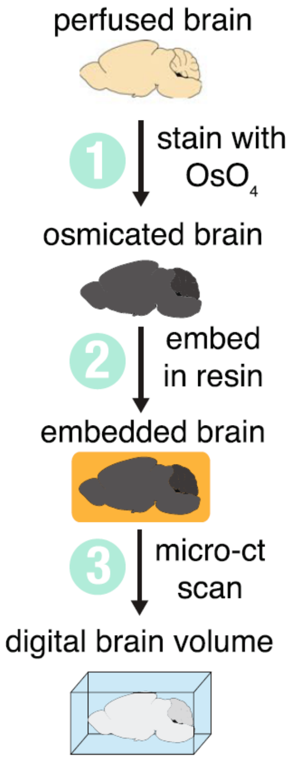
Automated Segmentation of 3-D Brain Imaging Data Using Convolutional Nets

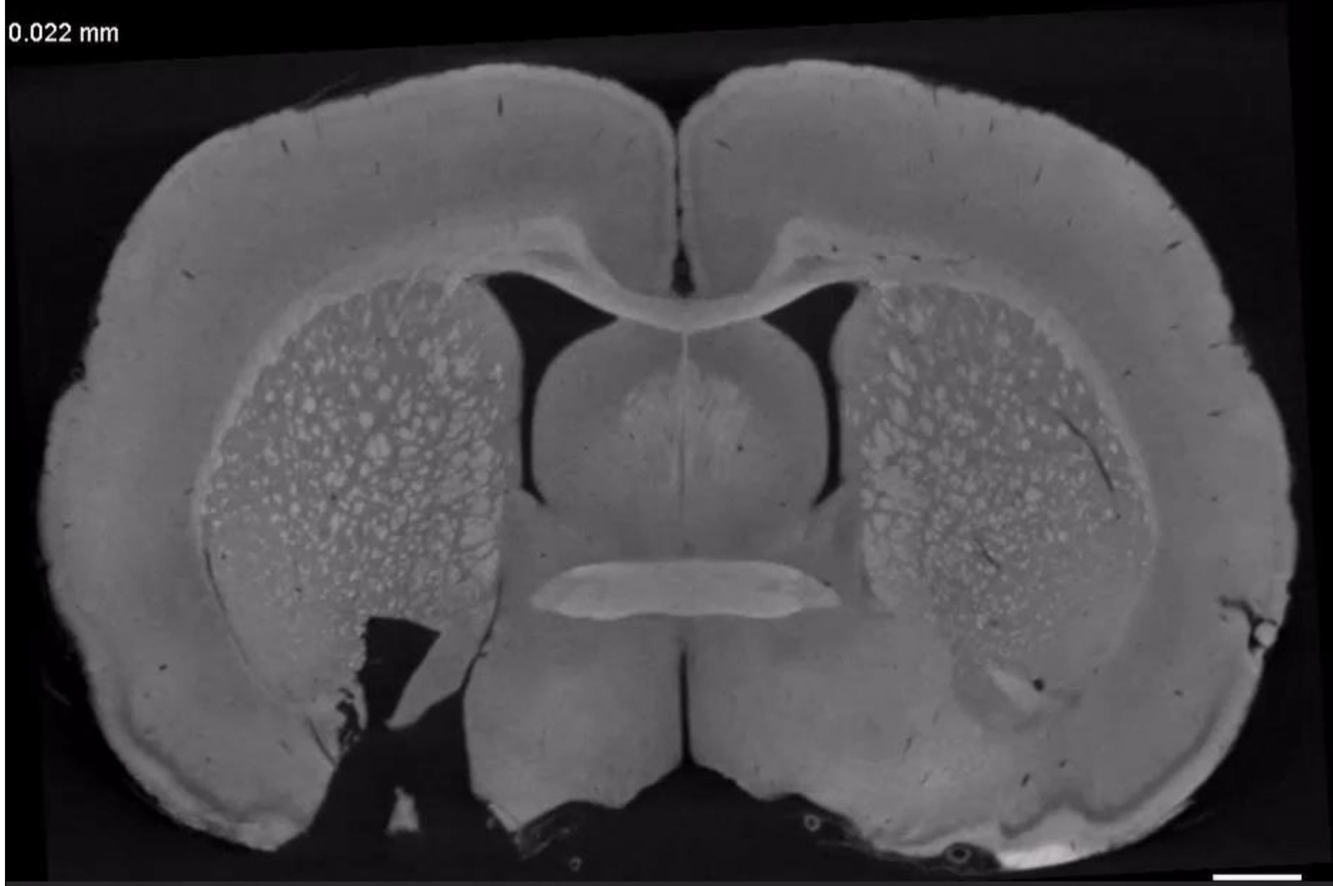
Gerald Pho

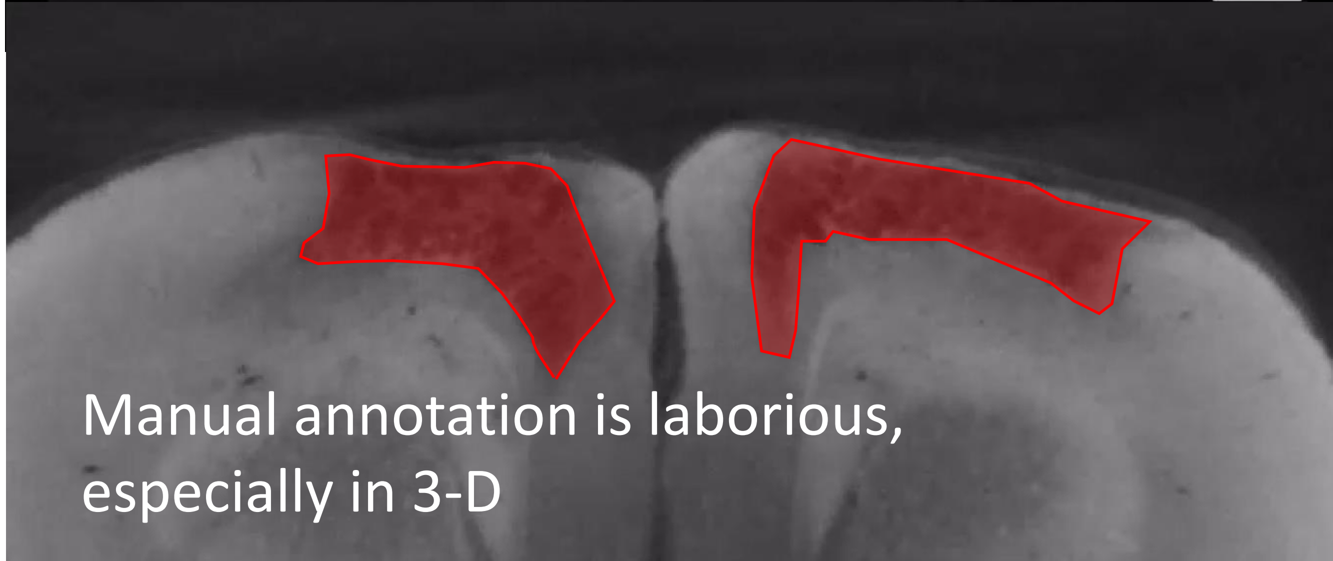
December 20, 2018

Goal: Quantify brain lesions with micro-CT volumetric imaging



Masis et al 2018





Manual annotation is laborious, especially in 3-D

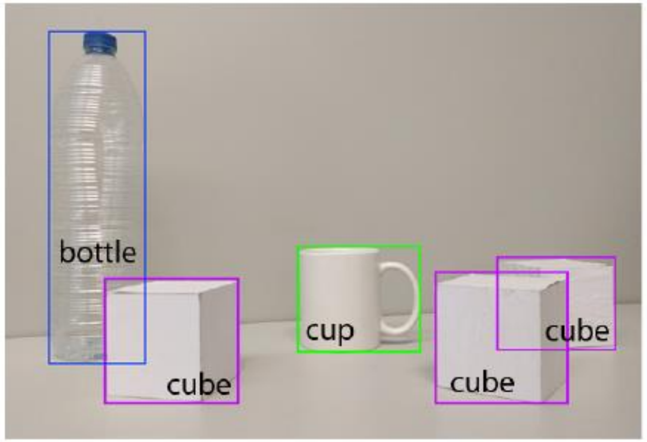
Can we use ML to automatically segment these volumes with sparse training data?

The general ML problem: semantic segmentation

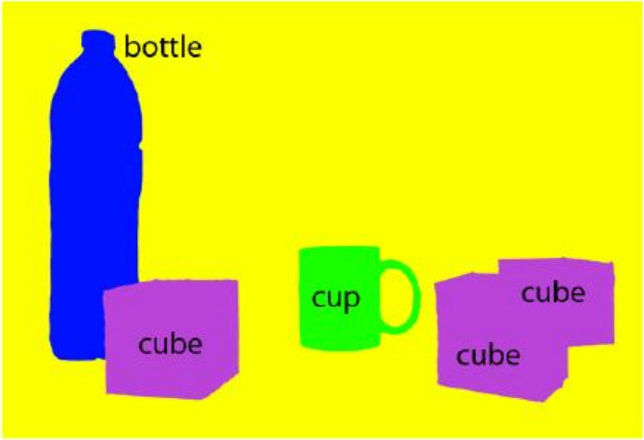
Deep networks are widely successful for **classification** (predict label per image)

How can we use deep networks for **segmentation** (predict label per pixel) ?

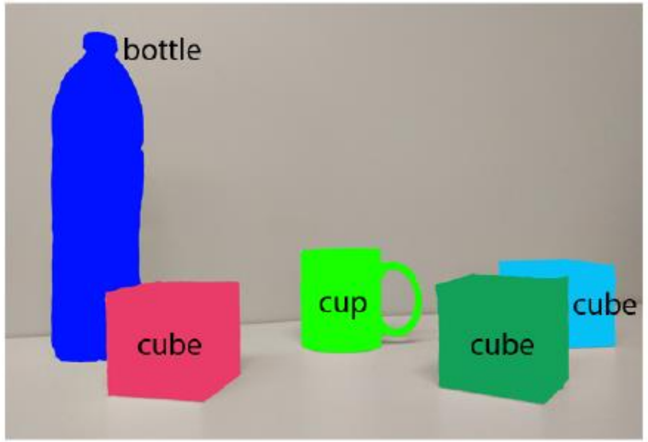
(a) Image classification



(b) Object localization



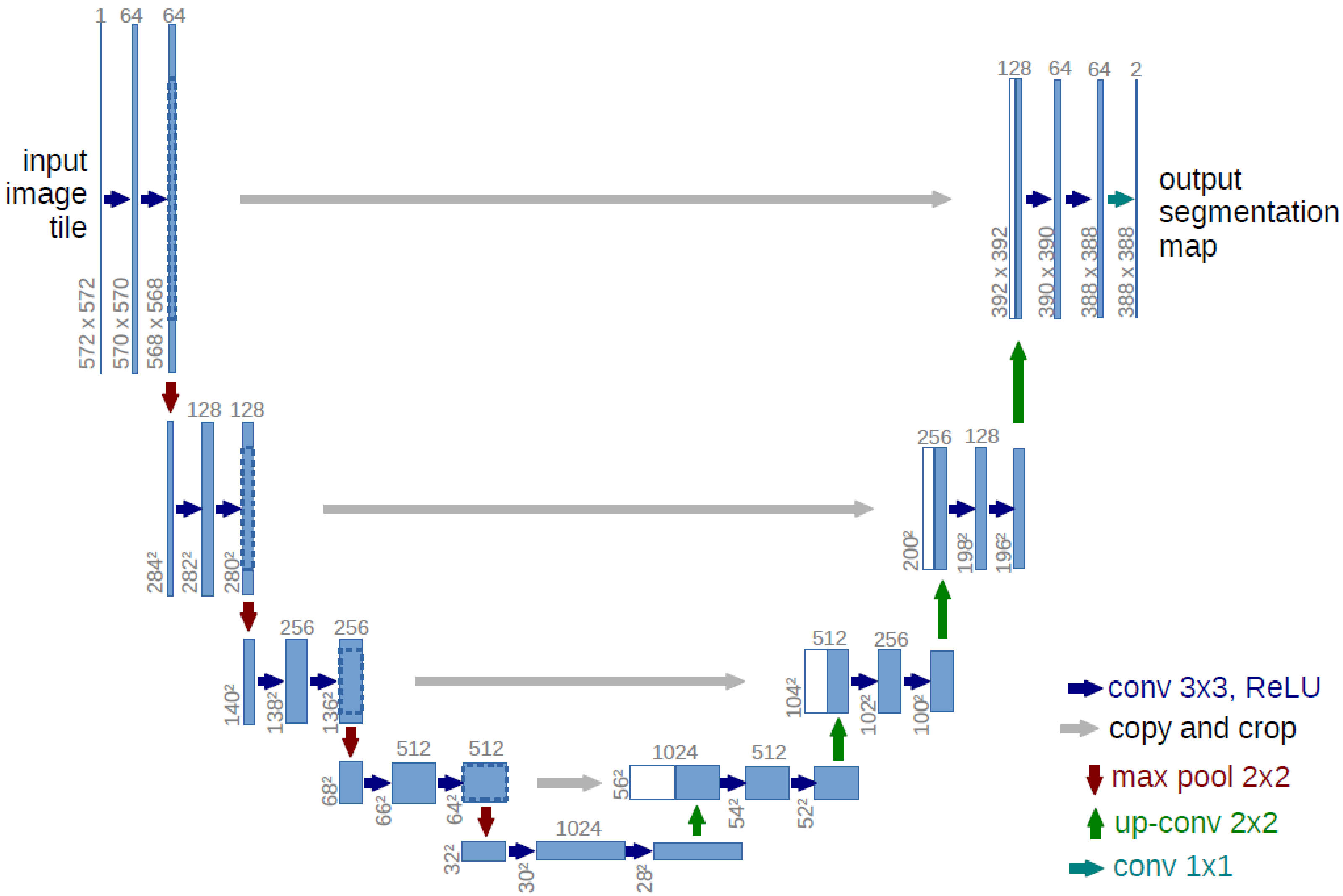
(c) Semantic segmentation



(d) Instance segmentation

Garcia-Garcia et al, 2017

The approach: U-Net…



A type of Fully Convolutional Network (FCN) – pixels to pixels w/ no dense layers

- **Input:** image
- **Output:** another image with each pixel encoding the probability per class

Contracting path (left) = just a CNN!

Expanding path (right) = upsample

Skip connections = concatenate right (deep features for generalization) w/ left (high-res features for localization)

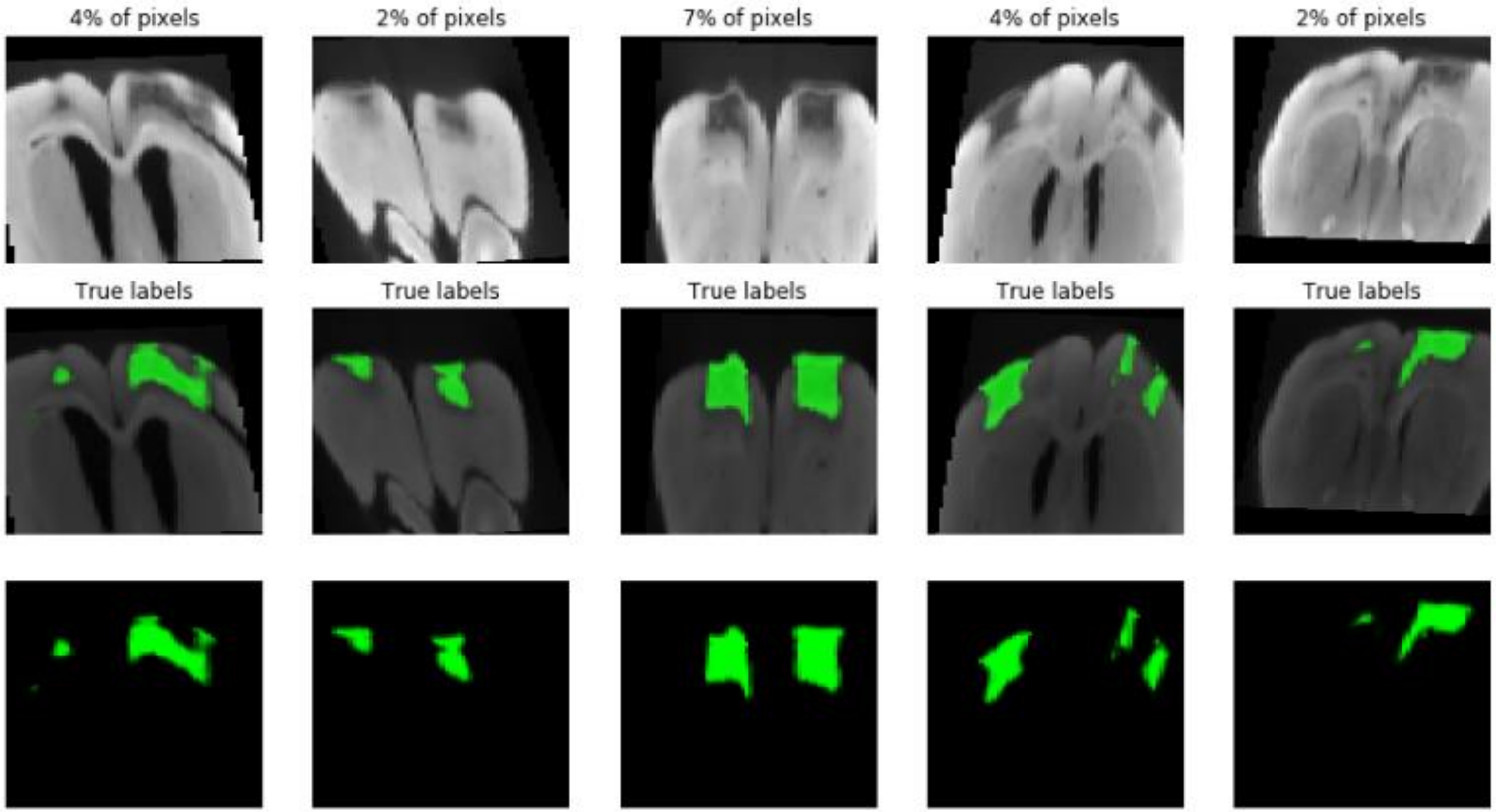
FCN: Long et al 2015

U-Net: Ronneberger et al 2015

… plus heavy data augmentation

Sparse training data → need augmentation!

- 3 rat brain volumes, each about 1000 x 600 x 600 → downsample and crop to 128 x 64 x 64
- Augmentations:
  - Rotation
  - Translation
  - Flip (horizontal)
  - Shear
  - Brightness

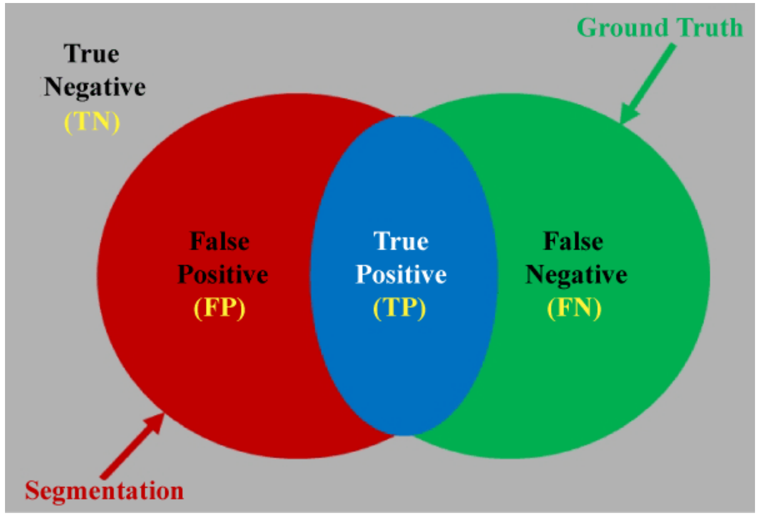


- Future work: random elastic deformations

Evaluation metric and loss function

Dice coefficient for evaluation

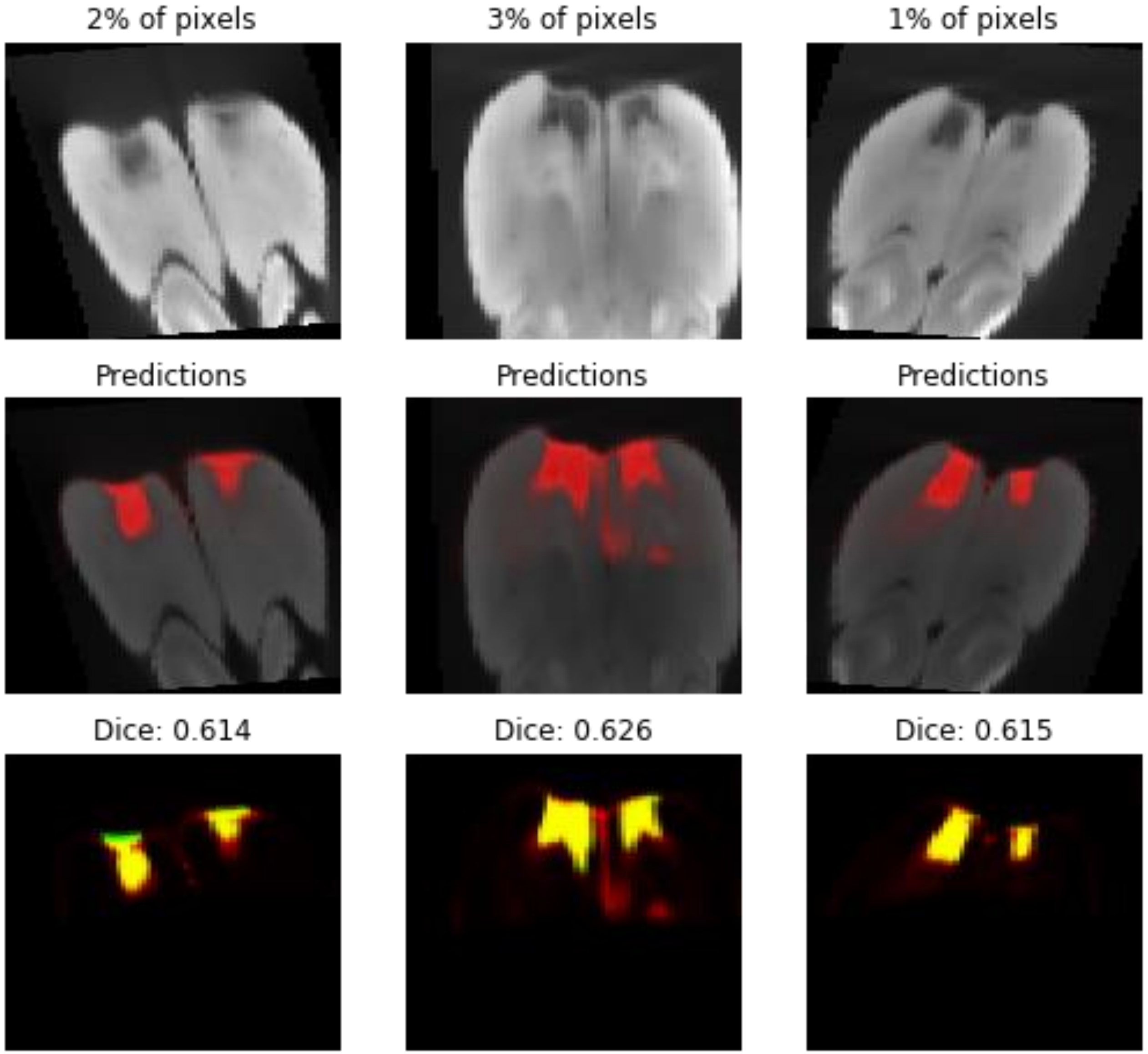
$$DSC = \frac{2TP}{2TP + FP + FN}$$



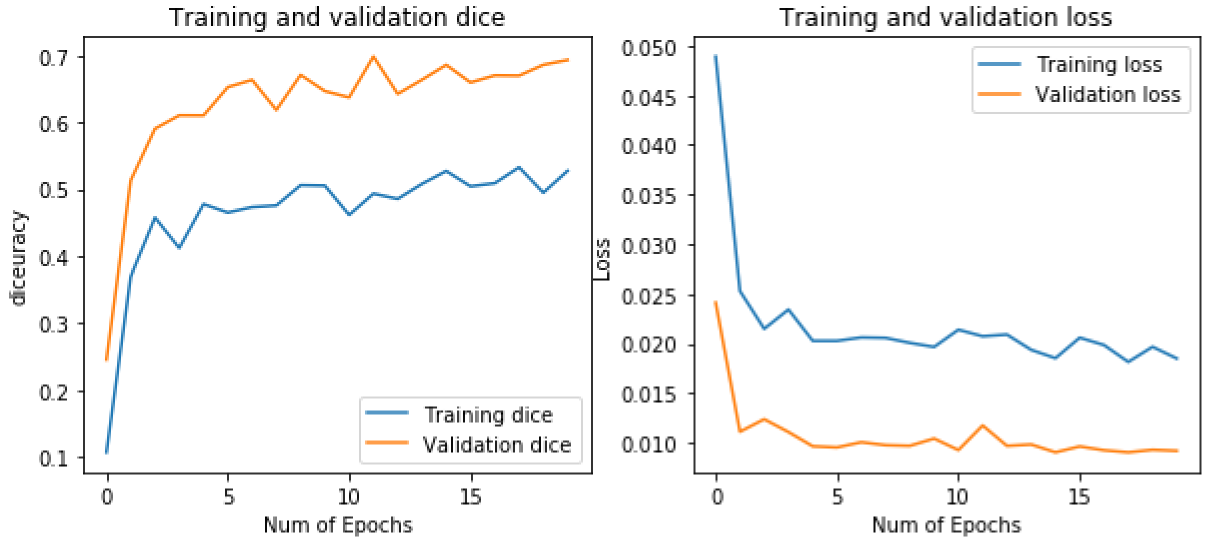
Weighted cross-entropy for loss function

- Class imbalance: give high weight to lesion pixels

Performance on training data

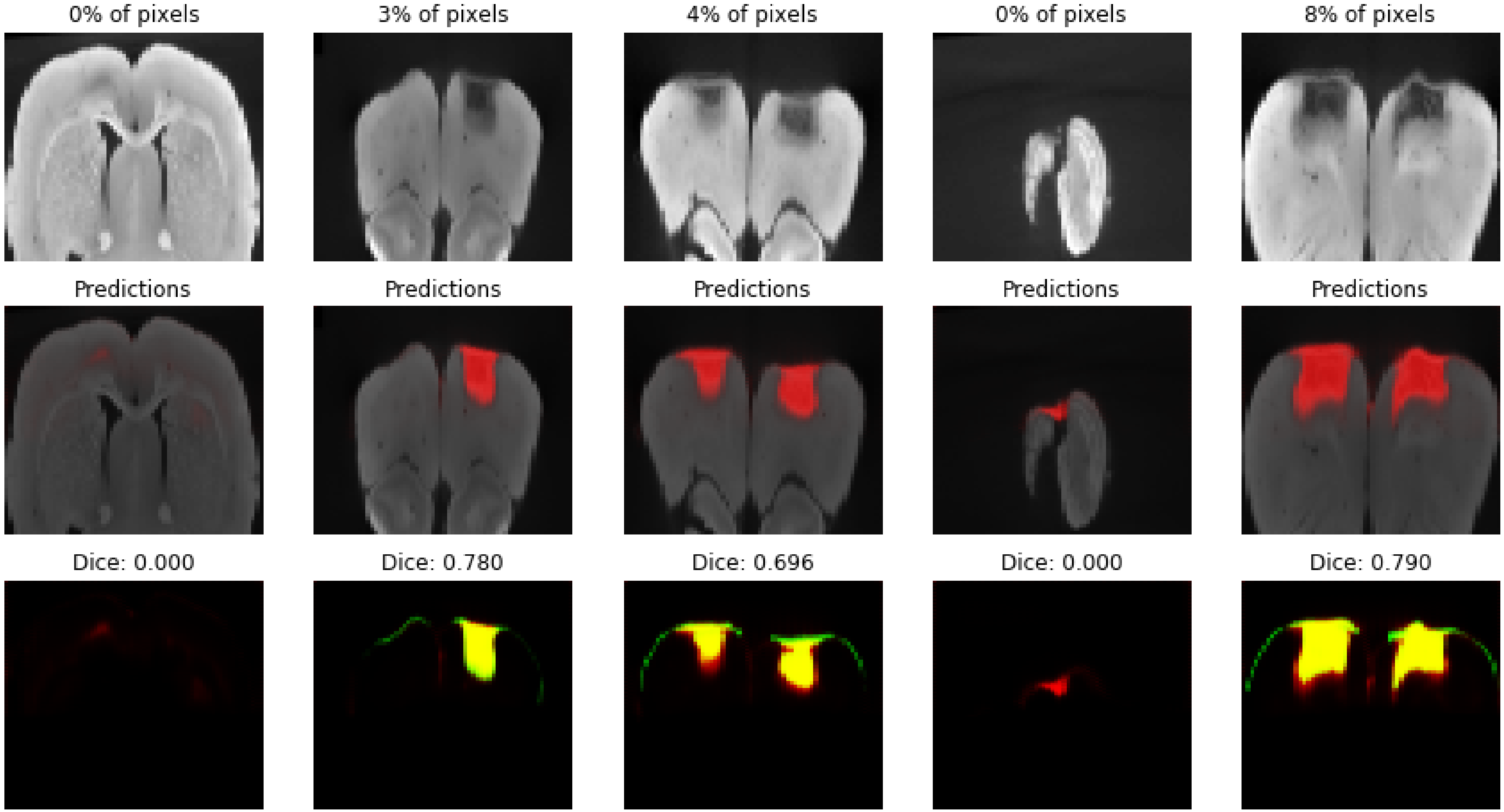


Training and validation loss

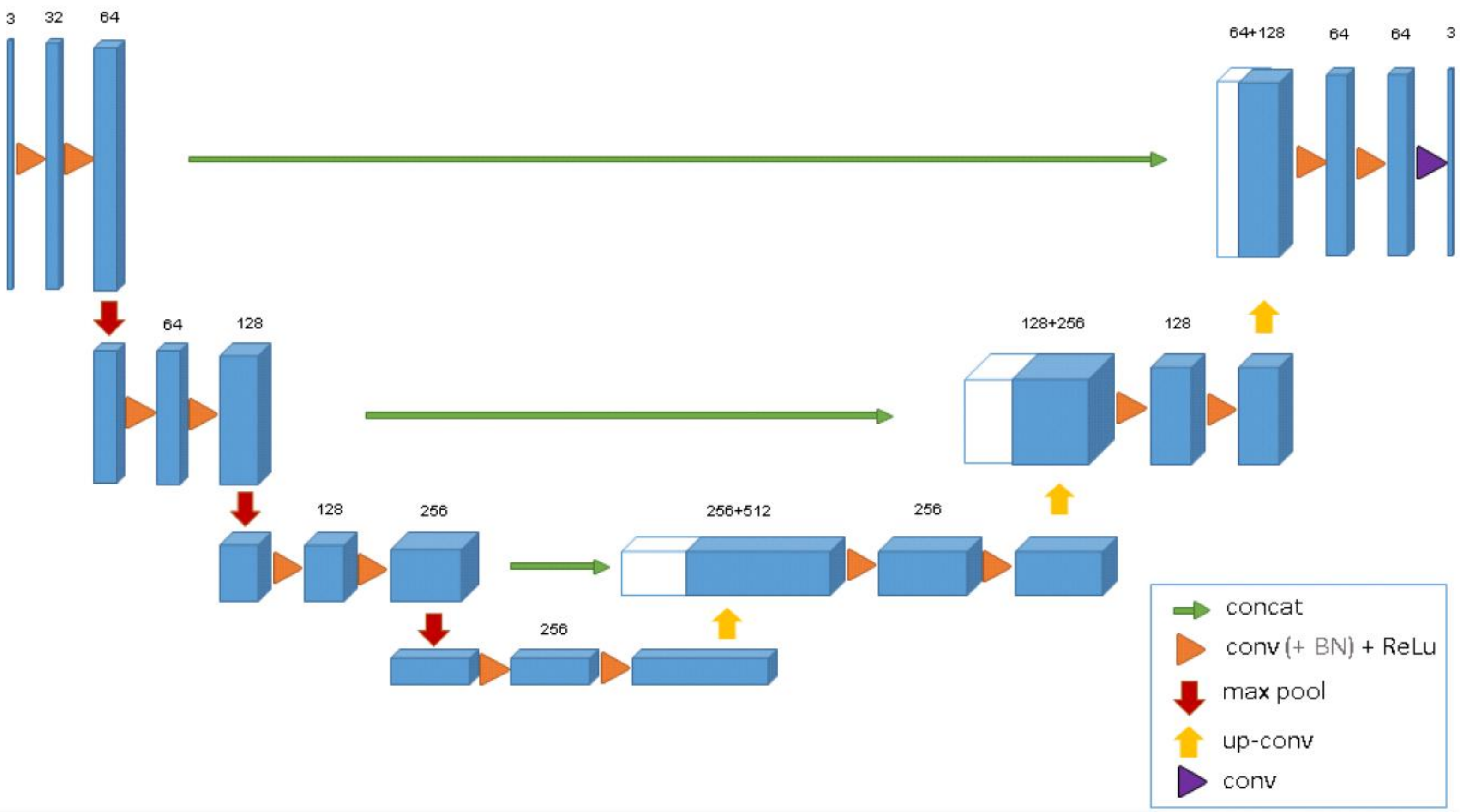


Trained for 20 epochs of 100 augmented images (~15 minutes)

Performance on test data



3D U-Net



Goal: Go from a sparsely labeled volume to a fully labeled volume

Labels have 3 classes: Lesion, Non-lesion, and Unlabeled

Output has 2 classes: Lesion, Non-lesion

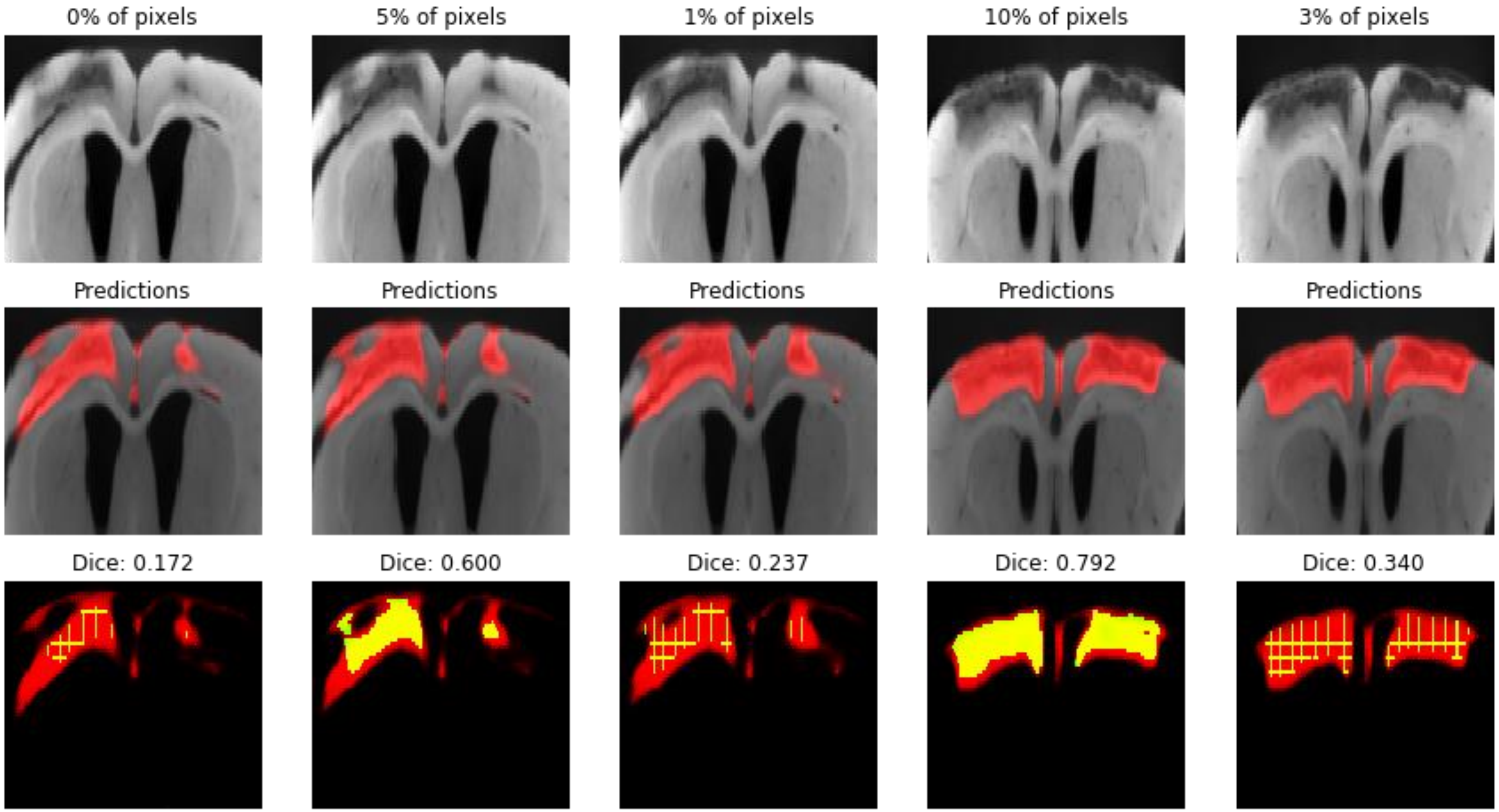
→Force network to make predictions on unlabeled data

Set training weight of unlabeled class to zero → has no effect on training loss!

Cicek et al 2016

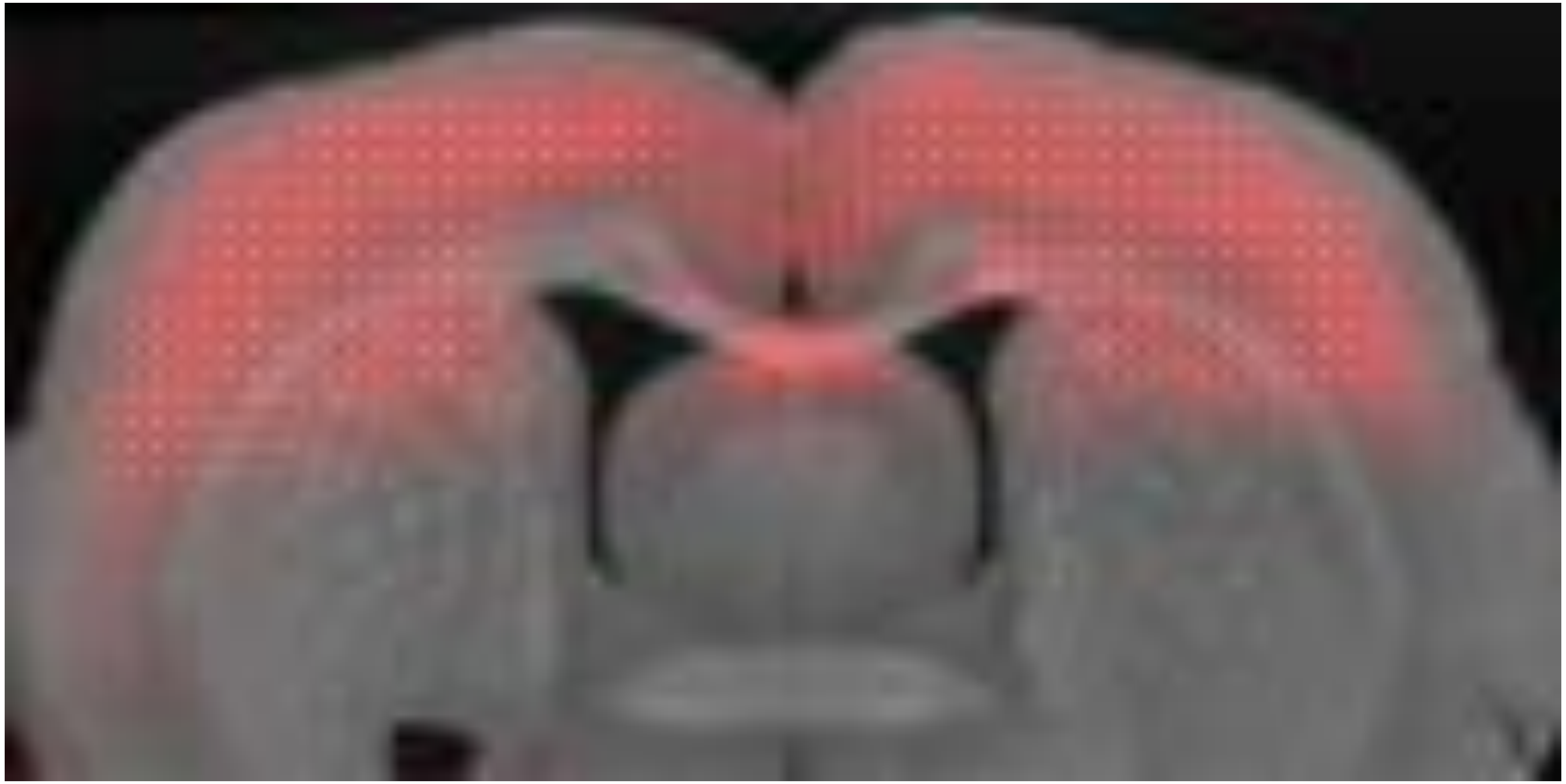
3D U-Net performance

Trained for 20 epochs of 30 augmented volumes (20 minutes)



3D U-Net tends to over-estimate lesion size

- Unclear whether it’s due to suboptimal choice of class weight or something else



Conclusion and references

- U-Net is a powerful approach for semantic segmentation
- Even just a 2D U-Net can work pretty well!
- Data augmentation is very helpful in case of sparse data

- U-Net paper = Ronneberger et al 2015
- 3D U-Net paper = Cicek et al 2016
- Useful starting code for U-Net: https://www.kaggle.com/keegil/keras-u-net-starter-lb-0-277