OAR Segmentation

# TO DO

* ~~Transforms not added~~
  + ~~I didn’t add data augmentation to the dataset yet. I added some code, but didn’t use it yet, because we need to add something for fixing the labels after transforming the volume. In the single class version they were simply thresholded, but here the values are class indici, so I guess we need to round them to their nearest existing class index.~~  **DONE**
* I didn’t explicitly deal with the problem of overlapping organs (multiple organs being annotated for the same pixel). When making the label mask volume, it now just goes through the annotations for that volume, and overwrites the class index for pixels that occur multiple times. So the last annotation wins. It would ofcourse be better to have a sensible ordering of which classes should take precedence. But didn’t have time to finish that.
* ~~Evaluation~~
  + ~~I removed the Dice calculation. Not sure what the standard is for evaluating on multiclass segmentation, but maybe implement a multiclass dice score?~~ **DONE**
* Center crop in\_plane
  + Remove unneccessary black pixels on border
  + Possible upsample crop to original size (not sure?)
* 2D UNET
  + Possibly with some depth as channels
* Class weights
* For 16bit training, use memory by increasing depthwise crop size (e.g. 16->32)
  + Or batchsize
* Update training script with more tensorboard output (and use pytorch summarywriter)
  + Images
* Do experiment with with subset of classes and non-missing annotations
* Implement basic missing annotations solution (spatial masking)

# Challenges

Every subject has all the organs, but the dataset has annotations missing. This means that training basic supervised training for multi-organ segmentation would lead to lots of False Negatives.

# Ideas

1. One has to start with Unet (obviously 3D). Let’s see what it does and confirm the effect of the underlying problem.

2. Set up “Backprogate gradients for only confident annotations” along with “curriculum learning” and “self-supervised learning”.

3. Unet with auxilliary losses in the upsampling path

4. Unet with auxilliary predictions fed back to the upsampling path

5. Think of something for active contours in neural networks, conditional random fields, refinement network, boundary loss term

# Dataset size for various combinations of classes

