The Technical University of Denmark

Deep Learning in Computer Vision

Project 2 - Image Segmentation

Authors:
Tobias Konradsen (s144077)
Helena Hansen (s153178)
Christian Dandanell Glissov (s146996)

Supervisors:
Aasa Feragen
Morten Hannemose

June 25, 2020, Kongens Lyngby

DTU Compute
Department of Applied Mathematics and Computer Science

DTU

1 LIDC Dataset

The dataset consists of medical x-ray images of lungs, and segmentation of lung cancer by 4 annotators. The dataset consists of 12816 lung images with a data split of: 8843 images for the training set, 1993 images for the validation set and 1980 images for the testset. For all of the sets there is 4 manual segmentations for the images. All of the images is of the size 128 by 128.

The 4 segmentations to every image is performed by 12 annotators in total, and it is not noted which annotator is responsible for each segmentation. The segmentations themselves depend largely on the annotator performing it, which results in variations both within each set of segmentation and between the sets themselves.

The segmentation only consist of two classes, $[0, 1]$, 0 being the background and 1 the lung cancer nodule, for which there is a clear imbalance, as the background is much more prevalent.

1.1 Validation measures

In order to validate our segmentation procedures we utilize the following measures: Dice overlap, Intersection over Union (IoU), accuracy, sensitivity, and specificity. The different error scores can be defined according to the numbers of true positive (TP), true negative (TN), false negative (FN), and false positive (FP).

The Dice overlap is a similarity measure and calculated in the following way:

$$Dice = \frac{2TP}{2TP + FP + FN}.$$

Intersection over Union (IoU-score) is the overlap of the 2 segmentations divided with the union of them, this measure is calculated as:

$$IoU = \frac{TP}{TP + FP + FN}.$$

Accuracy, is the proportion of true results, either true positive or true negative:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}.$$

Sensitivity being the proportion of true positives that are correctly identified:

$$Sensitivity = \frac{TP}{TP + FN}.$$

Finally, specificity being the proportion of true negatives:

$$Specificity = \frac{TN}{TN + FP}.$$

As we have a high imbalance between the classes, some of these measures will not be very beneficial to look at. As it is very easy for the model to predict TN's (black background) and there is very few foreground pixels, the accuracy will easily be very high and not a good representation of model performance. The same goes for specificity, we will have a large amount of TN, with very few FP, making the specificity around the 99%. We also have to take the type of problem into account, as we try to find cancer and find the area of the cancer nodule to avoid targeting healthy tissue with relevant treatments. In such a case it is important to find FN, TP, FP targets, e.g. we want to find all of the lesion (maximize TP, minimize FN) and minimize targeting healthy tissue (minimize FP). A high sensitivity will take into account the TP and FN, but not FPs. Dice overlap and IoU-score do take it all into account and seems to be the most relevant metrics for this case. It can be noted that $\frac{Dice}{2} \leq IoU \leq Dice$, where IoU penalizes false classifications more than Dice.

2 A simple segmentation CNN

Our basis for segmentation is a deep fully convolutional neural network architecture for semantic pixel-wise segmentation termed SegNet, Badrinarayanan et al., 2015, which consists of an encoder

network learning the features of the model, and a decoder network which outputs the segmentation. The idea being SegNet having a hierarchy of decoders, one corresponding to each encoder.

Modifications to the SegNet itself may be performed by changing the numbers of channels, stacks of convolution layers, and number of convolution layers in these stacks.

2.1 SegNet architecture

The dataset was fitted onto different SegNets with slightly different architecture. As the encoder network in SegNet is topologically identical to the convolutional layers in VGG the different variations tested was decided with basis in the VGG architecture, Simonyan and Zisserman, 2014.

An illustration of our final implementation can be seen in Figure 1. This network achieves the validation measures seen in Table 1.

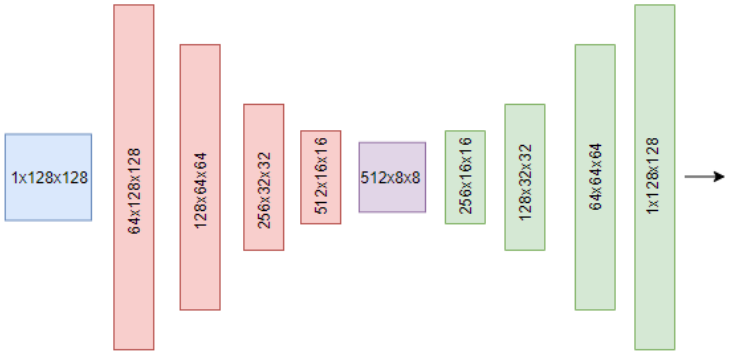

Figure 1: The segmentation network. Red indicates a down-sampling using max-pooling. Green indicates up-sampling. Purple is the bottleneck. Each square is a convolutional layer, except for the input image (blue)

Testing of different architectures in this form, a final model was implemented, with 4 max-pooling stacks with one convolutional layer, so that the bottleneck becomes a size of $(512 \times 8 \times 8)$. Increasing the number of channels when downsampling is done, is to make sure the network have the necessary neural capacity to capture sufficient information of the images. As we have a binary classification problem the output will be fed to a sigmoid function that maps the output to probability space.

2.2 SegNet Results

Our best results for the SegNet after 80 epochs can be seen in Table 1. We see fairly similar results for the validation and test set, while the scores are much higher for the training set. This discrepancy between the scores of the dataset is a sign of overfitting, and a show of the SegNet's insufficient capability of generalizing the dataset.

%	**Accuracy**	**Sensitivity**	**Specificity**	**DICE**	**IOU**
Training set	99.79	74.20	99.93	79.70	66.57
Validation set	99.65	56.24	99.86	59.88	43.95
Test set	99.59	54.35	99.89	57.68	42.25

Table 1: Measures of the SegNet

From the sensitivity scores it can be seen that we classify 54.35% of the cancer pixels according to the annotation, while still having a high number of FNs according to the IoU. As there are multiple sources of uncertainties; model, image and annotations themselves, it is hard to comment about the models performance, and this is the reason why a later section will look into the uncertainty concerning utilizing the annotations as ground truth.

3 U-net

U-nets utilize the same architecture as SegNets, while adding a number of skip-connections between the stacks in the encoder and decoder in order to help reconstruction, as information may have been lost. This loss may both have been as a result of the up and down-sampling, but may also be a gradient information that have been lost going through multiple layers leading to vanishing gradients. Our U-net follows the same architecture as our SegNet, except we use a stack of two convolutions before and after each skip connection to resemble the U-net structure more and to exploit the skip connections in hope of capturing more features by the extra layers.

3.1 Improving the segmentation

As with the SegNet several versions of the architecture was tested, where changes to the number of channels, stack and convolution layers was looked at.

For data argumentation we tested flip, rotation and two methods of cropping: center and random. Both cropping methods resulted in lower train and test scores, while flip and rotation only affected the training statistics. As rotation is only a more encompassing version of flip, and edges are a non issue as foreground objects tend to be centrally placed in the images, 180 degrees rotation made the network significant less prune to over-fitting and improved the regularization of the network.

As mentioned in section "LIDC Dataset" the data has a large imbalance between the foreground and background pixels, with the latter being the most frequent. Depending on the loss function this may be problematic, as the neural networks will classify any image as only consisting of background and achieve a high accuracy. To mitigate this behaviour a positive weighting is assigned to the loss functions, $\alpha \in [0, 1]$. The weighing increases the loss more for the white class for $\alpha > 0.5$, assigning white pixels a higher influence.

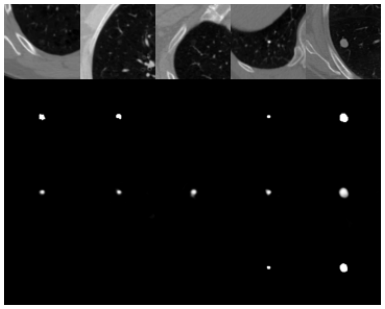

Figure 2: 1st row is the image. 2nd row is the annotation. 3rd row is the model prediction from the sigmoid and 4th row is the thresholded output.

The loss functions we tested for the models were the binary cross entropy (with and without α weighting) and the focal loss. The focal loss is defined as:

$$\mathrm{FL}\,(p) = -\alpha\,(1 - p_t)^{\gamma}\log\,(p_t)$$

Where $CE = -log(p_t)$ is the cross entropy, for the binary case this is defined as $BCE = -[y\log(p) + (1 - y)\log(1 - p)]$. Focal loss can put more focus on hard, misclassified examples by adding a factor $(1-p)^{\gamma}$, Lin et al., 2017. As we are interested in the white pixels, this will be added to the first term in the BCE. It leaves us with the following focal loss:

$$\mathrm{FL}\,(p, y) = -\alpha\,(1 - p)^{\gamma}\,y\log(p) - (1 - \alpha)(1 - y)\log(1 - p)].$$

Despite the class imbalance our predictions did include both classes before weight adjustment for the BCE. In the FL we had to adjust the weighting to see results, we found $\alpha = 0.7$ to be optimal, weighing the white pixels more. We noticed a worse performance when having a too high α, as this forces the model to have white pixels, giving a too high bias. From the paper, Lin et al., 2017, $\gamma = 2$ is found optimal, we did not thoroughly test different values of γ and therefore used the same value.

We set the threshold of the model output to be 0.5, this means that a confidence > 0.5 will be assigned a pixel value of 1 and confidence ≤ 0.5, 0. One of the issues of setting a fixed threshold is that it can make the model performance worse. Looking at Figure 2, we notice that before the threshold the model predicts with low probability that there are lesions. Two of the predictions are correct, while one of them is wrong. The threshold correctly removes two of the correct predictions, resulting in more FNs, but also incorrectly for one of them resulting in more TNs. The threshold was later optimized based on sensitivity, dice and IoU score for the validation set in one of U-net models, the value was found to be 0.4, but it was not used in the report, see appendix in section 6.

3.2 U-Net results

Our best results for the U-net can be seen in Table 2. It can be noted that our results for the test set is better than that for the validation set, as opposed to the SegNet where this was the opposite.

%	Accuracy	Sensitivity	Specificity	DICE	IOU
Training set	99.68	59.59	99.89	63.89	48.86
Validation set	99.65	55.47	99.87	54.34	39.81
Test set	99.62	59.70	99.89	62.55	47.16

Table 2: Measures for the U-net

Comparing our results to that of our SegNet, it can be seen that the U-net performs better on the test set, while worse on training and validation set. This may be due to better generalizing from the inclusion of two convolutions in each stack layer, and better reconstruction due to the inclusion of skip-connections as in comparison, but it may also just have something to do with the images themselves in the set.

4 Investigation of segmentation uncertainty

We have until now only considered the segmentations from a single set, but as there are 4 different included in the images, there is some natural uncertainty included in the dataset of whether one pixel belong to one class.

One way to investigate the uncertainty is to make an ensemble model, where we train our model for each annotated segmentation resulting in four different predictions for each image. We choose to continue with the U-Net, as it gave better results. While outputs produced by ensembles are consistent, they do not have much diversity and ensembles are typically not able to learn rare variants as each are trained independently, Kohl et al., 2018. An example of the result of one image from our ensemble model can be seen in Figure 3 and Figure 4.

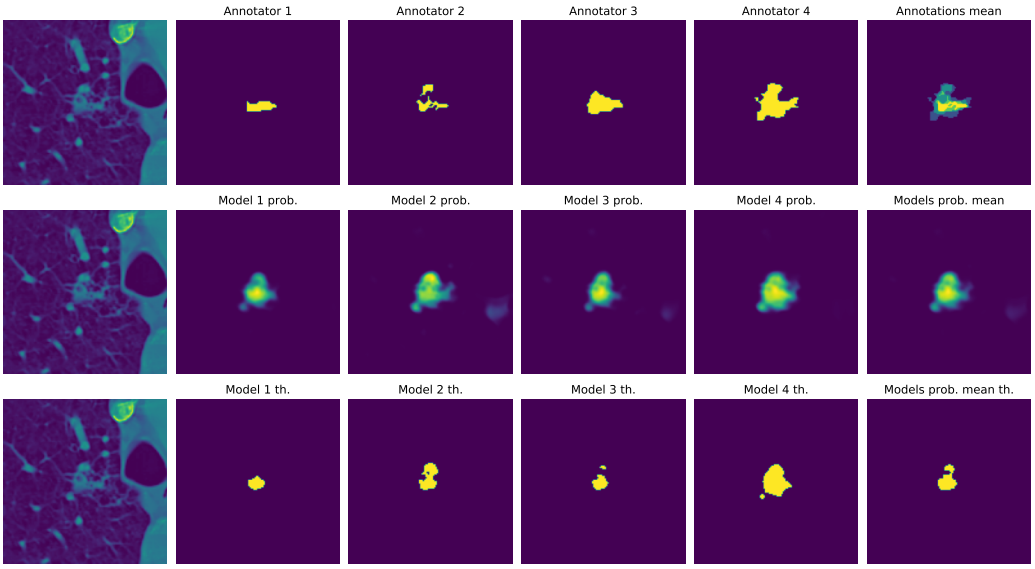

Figure 3: To the left-most the images can be observed. To the right-most the mean of the respective measures can be seen and in the middle each of the annotations. The 1st row contains the ground truth annotations, the 2nd row the corresponding model confidence output and in the 3rd row we threshold the confidences.

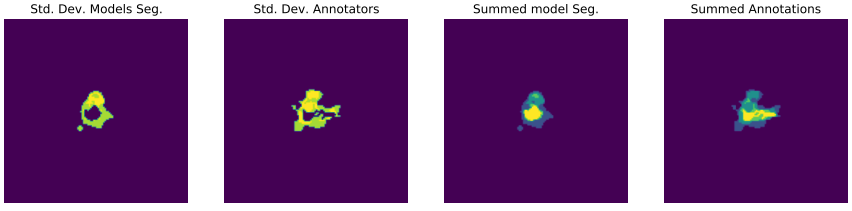

Figure 4: From left the standard deviation of the thresholded samples from the networks, the standard deviation of the annotations, and the summed segmentation of the model and the annotators.

In Figure 3 we see the difference of the annotators and the resulting segmentation by the 4 models. The "Annotations mean" image are the mean of all annotators segmentations, the "Models prob. mean" image are the mean of the probabilities for the four models, and the "Models prob. mean th" image is the segmentation gained from thresholding the combined probability image of the 4 models.

The four models all have slight differences in their probability distribution this is mainly a consequence of the annotators difference in segmenting the cancer lesions. So as when there is applied a threshold to segment the pictures on these probability distribution, the resulting segmentation will differ quite dramatically.

Figure 4 summarize the difference of the 4 ensemble models in 4 images, the first two show the standard deviation of the models and of the annotators. These images show the largest differences are in the boundary of the segmentations, as this is where the uncertainty between the 4 models are mainly located. As each of the models tries to approximate the distribution of the annotators, they will inherit the local uncertainty of each annotation. An ensemble model will therefore enable the possibility of an uncertainty estimate.

As we do not have a ground truth of the lesion area, but only approximations of the lesions from different annotators, this deviance in segmentation is another source of uncertainty for the model. A more consistent system of annotations within one segmentation set, will result in better results, but as of now the models capture the uncertainty of the dataset.

The 3rd image from the left, Figure 4, shows the summed segmentations for the 4 models, this is a way to visualize the differences between the models output. Areas where the 4 models agree will have a larger value (more yellow in the image). It shows that the models mostly agree for the center of the lesions, which is expected, as it is mainly found true for the annotators, as seen in the image of the summed annotations.

If the ensemble model is utilized for predictions, then there may be a risk of losing details or segmenting too much, depending on whether a mean or union is taken. The lack of fidelity can also be seen, as combinations of the annotations must also to some extent be valid cancer segmentation and these possible combinations cannot be provided from an ensemble as it models them independently. The ensemble model can be compared to the probabilistic U-net in order to illustrate its weaknesses, as the probabilistic U-net is capable of producing more unique variants by taking all annotations into account, though both models suffer from low fidelity, Kohl et al., 2019.

In Table 3 the mean of the 4 models error-values are calculated for the annotations they are trained on, as to give an estimation of the uncertainty of the ensemble model, and to find a confidence interval for each of the measures. The confidence interval is found by assuming a t-distribution and calculating standard error of the mean of the error measures. These results makes the uncertainty associated with the difference in annotations between sets clear, as the images and model is the same for each set, the difference in results must mostly be contributed by the annotations. The difference is largest for sensitivity, followed by Dice and IoU score, but they all range wildly depending on the segmentation set. This large difference in the error measures could be attributed to the fact, that some of the annotators are annotating within different confidence intervals on the lung tissue.

%	**Accuracy**	**Sensitivity**	**Specificity**	**DICE**	**IOU**
Training set	99.68 ± 0.02	67.79 ± 3.65	99.85 ± 0.07	66.61 ± 3.47	51.84 ± 3.72
Validation set	99.60 ± 0.11	65.17 ± 13.41	99.79 ± 0.11	58.47 ± 4.63	43.87 ± 4.42
Test set	99.58 ± 0.07	61.98 ± 11.73	99.81 ± 0.12	58.34 ± 6.11	43.46 ± 5.16

Table 3: Standard Error of the mean and the corresponding 95% confidence interval of measures for the Ensemble Net

Lastly, a source of uncertainty we cannot investigate or change are the images themselves, as they may contain noise that influence our results. The model itself have to take these into account.

5 References

Badrinarayanan, V., Kendall, A., & Cipolla, R. (2015). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *CoRR*, *abs/1511.00561*arXiv 1511.00561. http://arxiv.org/abs/1511.00561

Kohl, S. A. A., Romera-Paredes, B., Maier-Hein, K. H., Rezende, D. J., Eslami, S. M. A., Kohli, P., Zisserman, A., & Ronneberger, O. (2019). A hierarchical probabilistic u-net for modeling multi-scale ambiguities. *CoRR*, *abs/1905.13077*arXiv 1905.13077. http://arxiv.org/abs/1905.13077

Kohl, S. A. A., Romera-Paredes, B., Meyer, C., Fauw, J. D., Ledsam, J. R., Maier-Hein, K. H., Eslami, S. M. A., Rezende, D. J., & Ronneberger, O. (2018). A probabilistic u-net for segmentation of ambiguous images. *CoRR*, *abs/1806.05034*arXiv 1806.05034. http://arxiv.org/abs/1806.05034

Lin, T., Goyal, P., Girshick, R. B., He, K., & Dollár, P. (2017). Focal loss for dense object detection. *CoRR*, *abs/1708.02002*arXiv 1708.02002. http://arxiv.org/abs/1708.02002

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition.

Diary

15-06-2020

Goal for today was to complete question 1-3. Christian worked on the data loader and implemented validation performance measures into the model. Tobias and Helena worked on the basic CNN and the UNet. Everyone worked on the validation performance measures and the report.

16-06-2020

Goal for today was to complete question 4 and look into 5. Christian worked on the ensemble net and setting up the loss functions, including architecture and positive weights. Helena looked into improvements for the U-net, extra layers, data augmentation and dropout. Tobias looked into the probabilistic U-net and considered the empirical and theoretical investigation of uncertainty, he was also assigned as a designated debugger.

17-06-2020

Goal for today is to complete 5. Christian worked on loss functions, debugged the error measures and looked into whether Segnet or U-net had the best performance. Helena and Tobias wrote, looked into empirical investigation and trained the ensemble model. Tobias also created plots for the ensemble model.

18-06-2020

Goal for today is polishing and writing. All of the members helped for all parts of report. Helena worked as the main editor. Tobias worked mostly on writing the uncertainty section, and Christian worked on the section about improving the segmentation and also editing.

6 Appendix

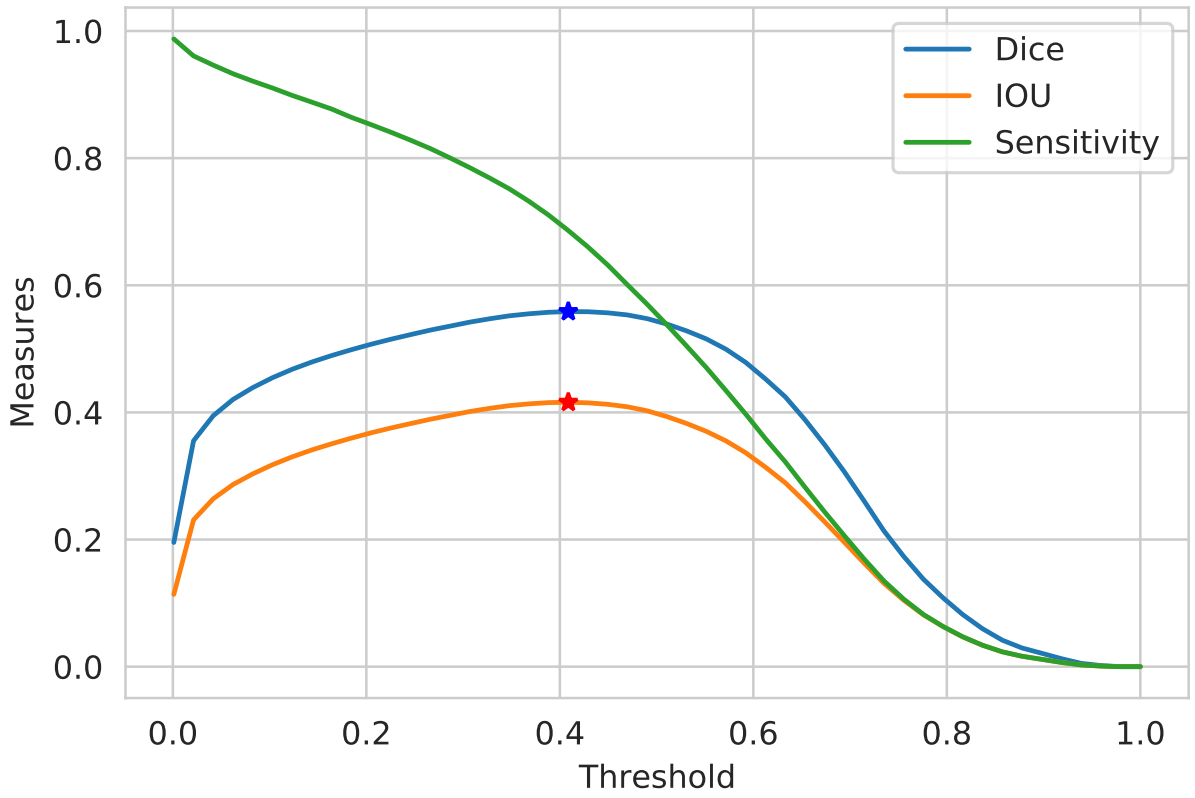

Figure 5: Best threshold based on the Dice, IoU and sensitivity to be 0.4.