Project 2

Deep Learning in Computer Vision

Authors

Group 4

Daniel Juhász Vigild - s161749

Lau Johansson - s164512

Frederik Kromann Hansen - s161800

June 18, 2020

1 Image Segmentation

1.1 Short description of the dataset and project purpose

Dataset The dataset contains images of lung CT scans from 1010 individual patients. Each of the scans have been annotated by 4 presumably random radiologists from a group of 12. See appendix Figure 3 for clarification. After a second reading by the annotators, where they had the opportunity to change their own annotation while looking at the others', the images are cropped in 180x180 pixels centered at where at least one annotator has indicated a lesion. For this project the images a pre-downsampled to 128x128 pixels with one input channel. This approach resulted in 8834 images for training, 1993 for validation and 1980 for test (slightly more in the paper by Kohl) [1]. The images of the lung CTs are normalized grey scale images. Regarding annotations, pixel value 1 (white) represents lung lesion and value 0 (black) represents background. Finally, the dataset is highly imbalanced with 99.2 pct. of pixels classified as background by the expert annotators.

Purpose Examining lung lesions caused by e.g cancer is done be analysing lung CTs. Radiologist often disagree on the annotations of lesions on a CT image - regarding both whether the lesion is there or not and on the boundaries of the lesion. This project explores the segmentation of lesions from lung CTs using two different Fully Convolutional Neural Networks for segmentation - the SegNet and U-Net. Finally, we wish to investigate uncertainty both from an empirical and theoretical angle.

1.2 Implementation of SegNet

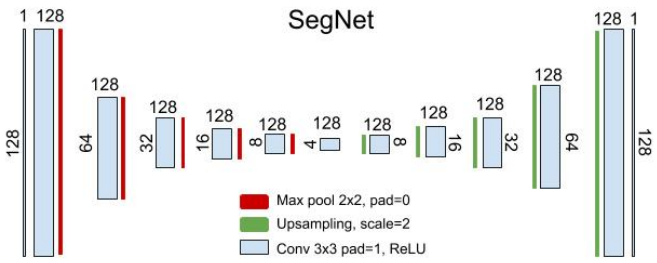

Figure 1: SegNet architecture

Segmentation of the lung lesions can be assessed with the use of semantic segmenta-

tion. A convolutional Encoder-Decoder (Segmentation network) is implemented as shown in Figure 1. Downsampling of the images by convolutional operations captures semantic information of the CT images. The network classifies each pixel into a class (lesion or background) as the output ultimately has the same size as the input.

Training Both of the models take batches of six images. The loss function is the binary cross entropy. The models are optimized using Adam with a learning rate of 0.0001 for 30 epochs.

1.3 Implementation of U-Net

The architecture of the U-Net is similar to the SegNet except for the introduction of skip connections and the number of features in each layer as shown in Figure 2. The final output of the U-Net is a representation of a 128x128 pixel image with one channel.

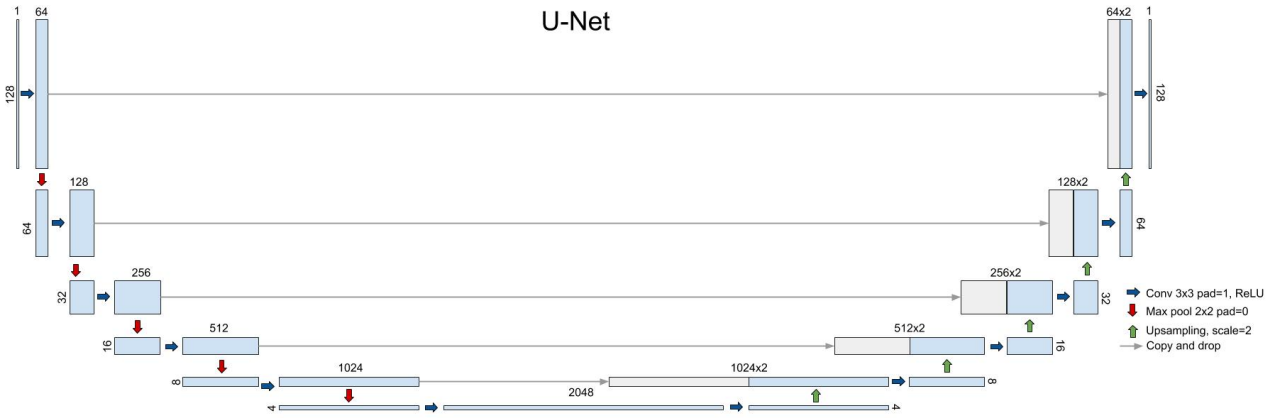

Figure 2: U-Net architecture

Performance measures To show the nuances of the lesion classifications the pixel values in the classification output is between 0 and 1 indicating the probability of a lesion. By inspecting Figure 4 (appendix) it seems that the model classifies the first three lesions well. In the fourth and sixth output, the model classifies lesions even though no lesions are annotated. This reflects the ambiguity in the ground truth annotated by the different radiologist. We know that at least one of the four annotations for each CT scan classifies it as containing a lesion, so outputting a vague segmentation of a lesion might reflect the truth quite well.

When evaluating model performance a threshold for the output pixels has been applied. The thresholds are respectively 0.1, 0.3, 0.5, 0.7 and 0.9. Threshold 0.5 means that if an output pixel value is below 0.5, it is set to 0 and 1 otherwise.

Mode and threshold	**Accuracy**	**Sensitivity**	**Specificity**	**IoU**	**Dice Overlap**
SegNet	0.9956	0.4269	0.9984	0.2459	0.4917
U-net	0.9954	0.5349	0.9984	0.3031	0.6061

Table 1: Average Performances of SegNet and U-Net on the test set.

Table 1 shows the average values calculated on the test set across thresholds and performance measures. On average, the U-Net outperforms the SegNet across performance measures. Looking at the sensitivity and specificity the U-Net is better at identifying the sick people without identifying more healthy people as sick. Also, the IoU is better for the U-Net, again proving that it is better at predicting the true position and extend of the lesions. In general the dice overlap performance is higher in the U-Net than the SegNet. Indicating that the classifications of the U-Net model are closer to the "true" annotations than the SegNet classifications. Looking at the performance for the U-Net on training data, the model performs with a dice overlap on average 0.86. With an average validation dice overlap of 0.65, the model seems to overfit. The networks perform better in sensitivity, when lowering the threshold to 0.1. Though, both dice overlap and IoU are almost unchanged. Finally, all accuracy values are higher than the dummy-baseline of 0.992 (expected performance by a majority class classifier). See appendix Table A and B for full description of performance measures. As for both of the models, simpler architectures were tested before memory constraints became bounding. In both cases, the most complex model gave the best overall performance.

1.4 Model improvement and performance metric discussion

Data augmentation Since we were challenged on available memory, we decided that we would rather fill the GPU with extra real images than augmented copies of images already in the training set. We expect an entirely new image to represent the distribution in the val/test sets better than an augmented image.

Weight on positive class Due to class imbalance in the dataset, 99.2 pct. of all pixels are annotated as background, the network is improved by penalising

false negatives resulting in an increase in the sensitivity. It is implemented by using "BCEWithLogitsLoss" with weights on the positive class. E.g. in the SegNet, when using weight 99, the sensitivity (threshold 0.5) on the test data has been improved from 0.52 to 0.80, but the dice overlap dropped from 0.58 to 0.47. The new loss function gave the U-Net a significantly more stable initialization.

Accuracy Since crudely predicting the dominant class can yield a high accuracy in an imbalanced setting (accuracy paradox), the metric should generally be used with caution. In our imbalanced setting, simply classifying all pixels as background, would result in a (deceivingly) decent performance of 99.2 pct.

Sensitivity and specificity In many cases, steps taken to improve either sensitivity or specificity worsens the other. Furthermore it is difficult to prioritize one over the other, and to do so requires a clear expectation regarding the consequences of a false positive and false negative. In a constructed example, where having an untreated lesion is dangerous and removing healthy tissue is almost risk-free, then it would be advisable to prioritize sensitivity over specificity, that is removing as many true lesions as possible at the expense of removing a lot of healthy tissue.

Intersection over Union Only as long as the predicted and actual segmentations overlap, IoU yields a value greater than 0, so no matter how far the segmentations drift apart the value will be 0 if they don't overlap. So, a segmentation in the near vicinity of the truth has the same performance as a segmentation much further away. This type of nuance can be useful in some problems, but is lost when using IoU.

Dice Overlap (F-metric) The Dice Overlap alleviates some of the issues shown for accuracy, which is that it works better than accuracy in imbalanced settings. It does so by taking both precision and recall equally into account. The weakness of the Dice Overlap however, is that it only focuses on one class (typically the positives, thereby ignoring true negatives) and also that is is not symmetric regarding labels, so flipping the labels would substantially change the performance value.

1.5 Uncertainty investigation

Ensemble To model the segmentation uncertainty an ensemble of four different U-Net models are implemented. The models have respectively been trained on each of the four annotations e.g U-Net model 1, is trained with target labels from annotation data set 1. The models use all training data in 10 epochs. The ensemble model takes an average weighting of each of the models predictions. The ensemble model is tested on annotations from dataset 1 (the one used all through this assignment). In Figure 5 (appendix) the four U-Net models segment four different regions as being a lesion, which reflects the underlying ambiguity. 3 out of 4 models classifies some area of the "true" lesion, which shows that the ensemble method works well. The best test set performance of the ensemble model (threshold 0.5) outperforms the simple U-Net on sensitivity, IoU and Dice Overlap (appendix, Table C). The source of uncertainty could stem from three areas: disagreement in whether there exist a lesion in the image, boundaries of the lesions and the annotation style of the radiologists. The level of detail in the annotations vary - some are neat circles others with more detailed boundaries (see Figure 3 in the appendix).

Reflection on theoretical approaches A way of quantifying uncertainty is by Monte Carlo Dropout (MCD) as proposed by Wickstrøm, et al [2]. This method trains the model with dropout and calculates the standard deviation of the softmaxed outputs from multiple runs in order to explain how uncertain pixels in the predictions are. A different strategy is a Bayesian approach [3], for example BCNNs or the probabilistic U-net, where variational inference is used to learn the posterior distributions of weights in the model. Distributions are learned by minimizing the KL-divergence between the prior and posterior distribution thereby approximating the true unknown distribution of the data. The Bayesian approach ultimately offers a probability distribution for each classification (for example lesion or not), that quantifies the uncertainty of the class probability. In contrast to the MCD, priors of any tractable family of distributions can be used in sampling predictions thereby enabling domain knowledge to influence the modelling. A way to validate both methods of uncertainty quantification could be to assess the standard deviations of the probability point estimates over multiple runs, and see if they coincide with those images that are annotated with high variance by the expert annotators.

2 References

[1] Kohl et al. 2020. *A Probabilistic U-Net for Segmentation of Ambiguous Images*

[2] K. Wickstrøm, M. Kampffmeyer, R. Jenssen. Nov. 20, 2019. *Uncertainty and interpretability in convolutional neural networks for semantic segmentation of colorectal polyps.* Published by Elsevier B.V. https://www.sciencedirect.com/science/article/pii/S1361841519301574, (Visited Jun. 18, 2020).

[3] T. LaBonte. 2020 *Deep Learning Segmentation with Uncertainty via 3D Bayesian Convolutional Neural Networks* found on: https://towardsdatascience.com/deep-learning-segmentation-with-uncertainty-via-3d-bayesian-convolutional-neural-networks-6b1c7277b078, (Visited Jun. 18, 2020).

Appendix

Diary for project 2

Lau prepared the data in a meaningful way for the training of models, while Daniel and Frederik prepared the initial SegNet and U-Net for training. Lau implemented weight on positive class. Daniel dove into a discussion of the different performance metrics. Frederik trained a well performing U-Net network. Lau was able to use the code and train four different models. Subsequently Lau combined them in an ensemble model and made nice visualisation of the ambiguities. Frederik and Daniel reflected on theoretical approaches for quantifying uncertainty.

Model outputs and tables

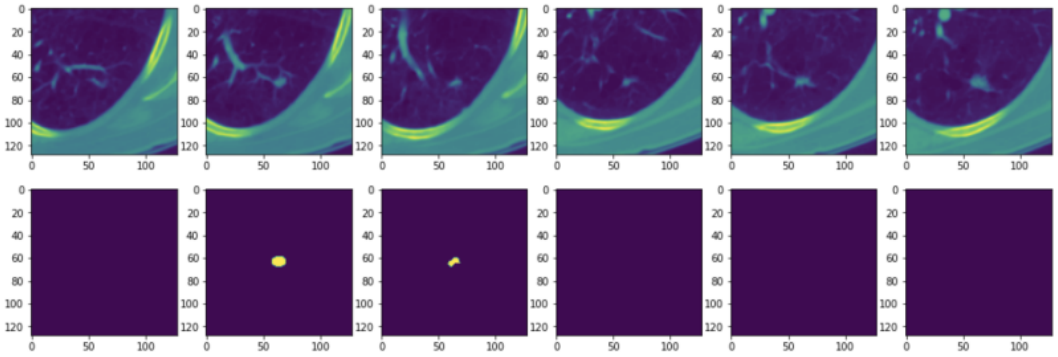

Figure 3: Annotations from "group one" in the dataset showing slices of a lung from an individual patient. Note the difference in annotation style, indicating that the annotations of each slice has not been conducted by the same annotator. On the leftmost image, there is no annotation. Secondly, there is a large circular one. Next, a very detailed smaller one followed by none again. It makes sense to shuffle the annotators, to prevent systematic bias.

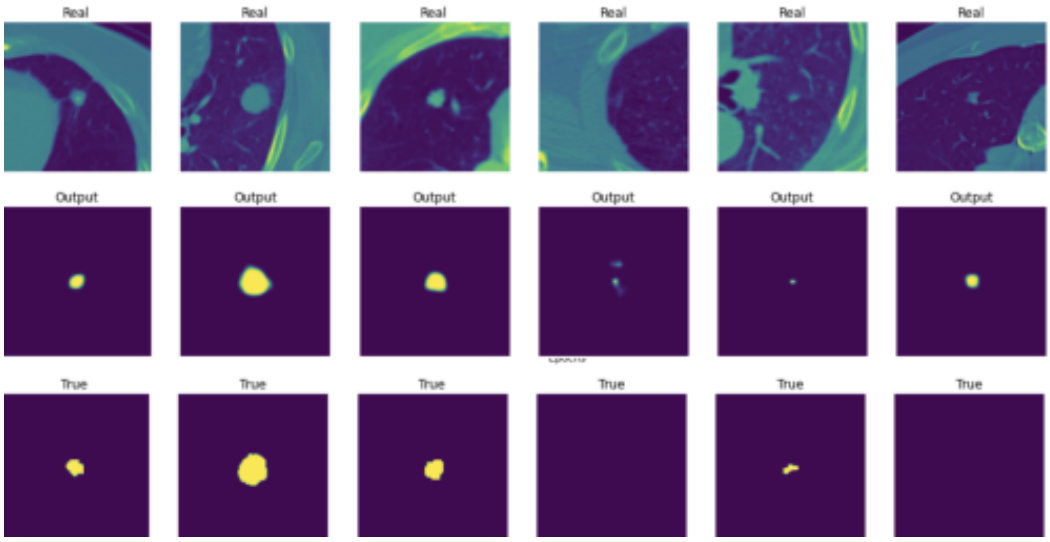

Figure 4: SegNet classification of lung lesions

SegNet performance

Mode and threshold	Accuracy	Sensitivity	Specificity	IoU	Dice Overlap
Training (0.1)	0.9981	0.9588	0.9983	0.4084	0.8168
Training (0.3)	0.9985	0.9482	0.9987	0.4110	0.8220
Training (0.5)	0.9984	0.8664	0.9995	0.4475	0.8950
Training (0.7)	0.9980	0.7451	0.9998	0.4208	0.8416
Training (0.9)	0.9985	0.6128	1.0000	0.3770	0.7540
Validation (0.1)	0.9969	0.5851	0.9982	0.2696	0.5392
Validation (0.3)	0.9931	0.4543	0.9974	0.2548	0.5095
Validation (0.5)	0.9958	0.5901	0.9976	0.2751	0.5503
Validation (0.7)	0.9977	0.4775	0.9990	0.2563	0.5125
Validation (0.9)	0.9959	0.4083	0.9995	0.2740	0.5480
Test (0.1)	0.9931	0.5969	0.9963	0.2910	0.5819
Test (0.3)	0.9951	0.5097	0.9980	0.2750	0.5501
Test (0.5)	0.9976	0.5205	0.9988	0.2945	0.5889
Test (0.7)	0.9957	0.3927	0.9991	0.2529	0.5058
Test (0.9)	0.9966	0.1449	0.9996	0.1160	0.2320

Table A: Performance measures across modes and thresholds for SegNet.

U-Net performance

Mode and threshold	Accuracy	Sensitivity	Specificity	IoU	Dice Overlap
Training (0.1)	0.9984	0.9858	0.9984	0.4047	0.8093
Training (0.3)	0.9991	0.9513	0.9992	0.4360	0.8720
Training (0.5)	0.9991	0.8610	0.9997	0.4436	0.8873
Training (0.7)	0.9990	0.8481	0.9998	0.4507	0.9014
Training (0.9)	0.9985	0.7425	0.9996	0.4244	0.8488
Validation (0.1)	0.9931	0.7533	0.9942	0.2592	0.5183
Validation (0.3)	0.9929	0.5624	0.9959	0.2619	0.5237
Validation (0.5)	0.9974	0.8933	0.9982	0.4214	0.8429
Validation (0.7)	0.9972	0.7059	0.9992	0.3877	0.7753
Validation (0.9)	0.9974	0.5075	0.9991	0.2900	0.5800
Test (0.1)	0.9947	0.6091	0.9974	0.3108	0.6217
Test (0.3)	0.9955	0.6035	0.9982	0.3273	0.6546
Test (0.5)	0.9953	0.4587	0.9983	0.2605	0.5209
Test (0.7)	0.9967	0.5156	0.9992	0.3102	0.6203
Test (0.9)	0.9948	0.4877	0.9991	0.3066	0.6132

Table B: Performance measures across modes and thresholds for U-Net.

Segmentation uncertainty

Mode and threshold	Accuracy	Sensitivity	Specificity	IoU	Dice Overlap
Test (0.1)	0.9948	0.7111	0.9964	0.3047	0.6093
Test (0.3)	0.9959	0.5801	0.9981	0.3074	0.6149
Test (0.5)	0.9969	0.6539	0.9985	0.3336	0.6671
Test (0.7)	0.9961	0.4664	0.9992	0.2926	0.5853
Test (0.9)	0.9958	0.3518	0.9996	0.2486	0.4972

Table C: Performance measures across modes and thresholds for Ensemble model.

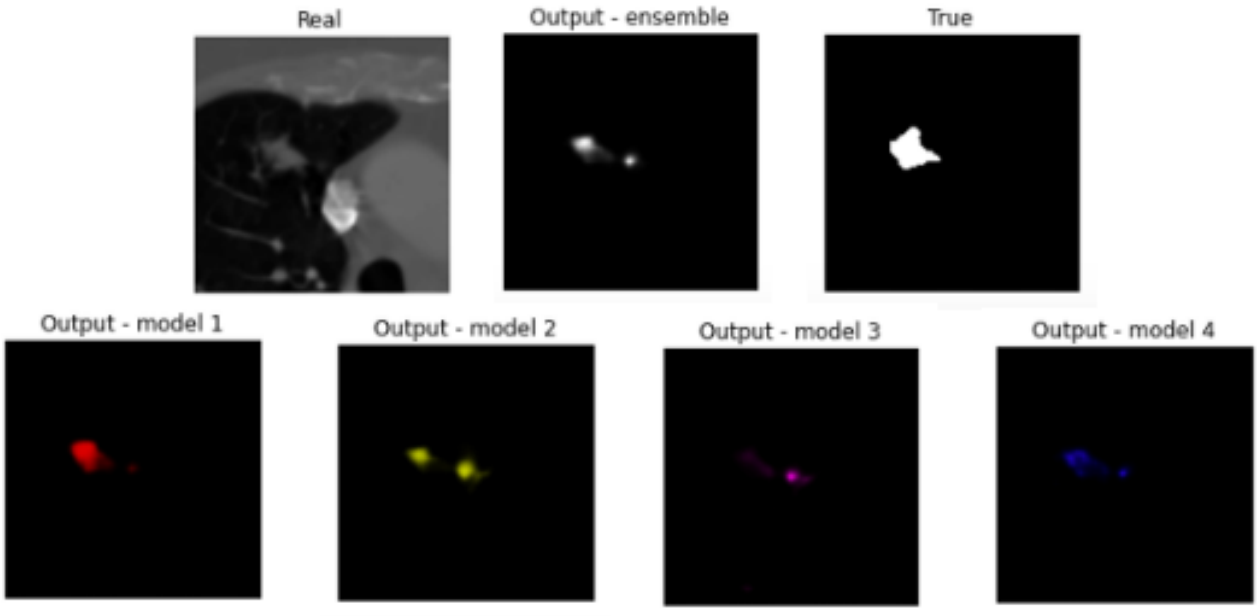

Figure 5: Predictions from ensemble model and the individual models