COMPUTER AIDED DIAGNOSIS
LUNA16: FALSE POSITIVE REDUCTION

Luc Nies (s4136748), Tom van de Poll (s4106512), Harmen Prins (s4132297),
Steven Reitsma (s4132343) & Inez Wijnands (s4149696)
Radboud University Nijmegen

1. INTRODUCTION

For the *LUng Nodule Analysis 2016 (LUNA16)* challenge, we want to detect pulmonary nodules in low-dose thoracic CT images as accurately as possible. Initially the challenge is separated into three different phases; lung segmentation, candidate detection and false positive reduction. For the first two phases, we tried two different approaches. Firstly, we used deep learning to create a network that could train on the images and could recognize areas of interest. For the first phase these areas were the lung regions, for the second these areas were possible nodules in the lung regions. This way, we could treat both phases as a segmentation problem and could use the same network. Secondly, we used a more traditional approach using image processing techniques like region growing and blob detection in the images to complete these phases.

After the second phase, we had so far obtained marginally better results for the deep learning approach than the traditional approach. Therefore, we focus solely on our deep learning approach for the third phase.

As mentioned, the third phase of our lung nodule detection challenge is false positive reduction. Heretofore, we have focused on identifying as many possible candidates as possible to include all annotated nodules in our selection. Thus, we only aimed at a high recall, but did not care much for precision. In this phase, we aim to obtain both high recall and high precision. We approached this problem by going back to the initial candidate selection, instead of taking our candidates and making a subselection to reduce the false positives. By tuning our network parameters and trying different settings overall, we tried to achieve better initial candidate selection and thus not needing false positive reduction, merging the second and third phase of the challenge.

Our approach and its results are explained in further detail in the following sections.

2. METHOD

We implemented two different types of networks; a 2D fully connected network and a 3D convolutional network. We describe these in the following.

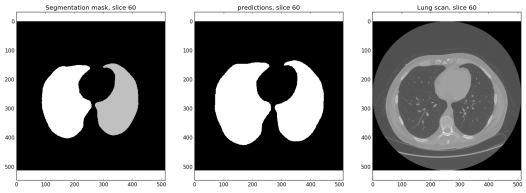

Fig. 1. From left to right: given lung mask, our segmentation, the actual lung scan.

2.1. Fully convolutional networks

2.1.1. Lung segmentation post-processing

We have made use of the segmentations created by the network in earlier phases of the project, in order to more carefully select training patches from the images for training. A small issue with the segmentations created by the network is that there is some noise surrounding the actual lung segmentation. In order to use these segmentations for patch selection, the noise has to be removed from the images. In order to remove the noise, we perform a connected component analysis with a very small 3d-6 neighborhood. This divides the noise next to the segmentations into a large number of small connected components, but still sees the lungs as one large connected component. After this step, only the largest component in the image is selected and used as the real segmentation. This method works for every image in the dataset, including the images on which region growing failed. However, the resulting lung segmentations did have a rather low mean dice score of 0.78 (compared to all of the given segmentations, including the faulty ones). After inspecting our predicted segmentations we found out that the segmentations are slightly larger than the actual lungs (roughly 5 to 10 pixels to each side). This can be seen in figure 1. So, despite the rather low dice score we still chose to use these segmentations as a mask in the actual detection of the nodules, as these segmentations are covering more of the lungs than the given segmentations, which is preferred over losing some parts of the lungs and thus having an increased chance of missing nodules in the nodule detection phase. [1]

2.1.2. *Nodule detection*

We used the same approach for the false positive reduction as for the initial candidate reduction: a fully convolutional network. When handling the nodule detection as a segmentations problem where we try to segment the nodules (the foreground) from the rest of the lungs (background) we are combining both the candidate detection and false-positive reduction in one step. We did use some alternative training schema's to improve both the recall and the accuracy of the network as compared to the previous step. Our main goal was to improve the precision of the results, while maintaining, and preferably even improving, the precision. The problem with the candidate selection fcn was that is classified almost anything that was white as a nodule. So we need to show it more samples of healthy lung tissue. We tried achieving this by using a different class balance during training which contained more healthy samples, so that the network could discriminate better between healthy lung tissue and actual nodules. We also restricted training set by only showing patches that actually contain lung tissue by using the lung segmentations we created as a mask, which resulted in a smaller training set containing more relevant samples (of the actual lung tissue instead of the area outside the lungs). Finally we also increased the training time of the network by a factor of three. The results of these different training schemes are described in section 3.1

2.1.3. *False positive reduction*

The FCN produces many false-positive pixels, thus producing a low precision. However, those pixels are almost never clustered together. Consequently, it is possible to do a connected components analysis on the output of the network and remove all components with a volume below a threshold. This threshold can be set manually or optimized using a small part of the data set. The second option is a good idea for future work.

2.1.4. *Analysing results with FROC*

To transform raw pixel data to coordinates as required by the FROC analysis, the same connected components analysis as for the false positive reduction is applied, and the center of every component is located. Then, the FROC analysis is performed with the nodule diameter as the distance threshold. This means that every prediction that falls within a nodule is counted as a true positive and otherwise as a false positive.

2.2. 3D convolutional networks

Next to the fully convolutional network approach, we also tested a three-dimensional convolutional network approach. This method is more simple than the fully convolutional network, but needs to have nodule candidates as its input. As such, it is not able to propose candidates itself, but needs some other system to do that. The idea of the network is simple: its input consists of a $40\times40\times26$ voxel neighborhood where the center is the nodule candidate coordinate. The output of the network is a softmax distribution over the labels: in this case 0 for no nodule or 1 for a true nodule. The network uses three-dimensional convolution and pooling layers to accommodate for the three dimensions in the data. Alternatively, one could use multiple slices of the 3D cube in different angles and combine them. However, we found that a three-dimensional approach performed better. An input size of $40\times40\times26$ was chosen by looking at the nodule sizes. Prior work has already analyzed this [3], and in Figure 2 you can see the result.

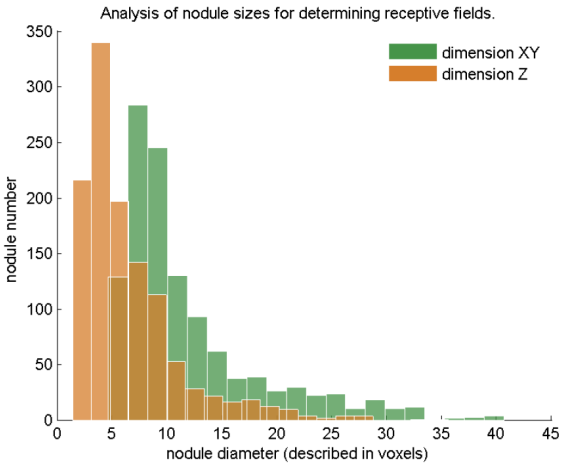

Fig. 2. Analysis of the nodule sizes. Taken from [3].

Just like in [3], we choose a smaller neighborhood in the Z dimension since its scaling is different from the other dimensions. The network structure consists of four three-dimensional convolutional layers with a kernel size of $5\times5\times3$. After each convolutional layer, pooling is performed with a size and stride of $2\times2\times2$. Finally, two dense layers are added with 512 and 2 units respectively. A softmax layer is applied to normalize the results and to obtain a probability distribution. All layers have batch normalization and dropout is used in the first dense layer ($p = 0.25$). The network is trained using backpropagation with stochastic gradient descent and Nesterov momentum (0.9). The learning rate is set to 0.001 and decays with 0.0001 per gradient step. Binary cross-entropy loss was used and leaky ReLUs were used for every non-linearity in the network. During training, each batch was balanced to contain approximately 50% negative and 50% positive classes. Since there are a lot more negative examples, each epoch different negative examples are chosen and negative samples are never shown twice. The network was implemented using Keras and Theano. Proposed candidates are all taken from the extended false positive reduction csv file (candidates_V2.csv) and evaluation was done using precision and recall metrics. The network's results are shown

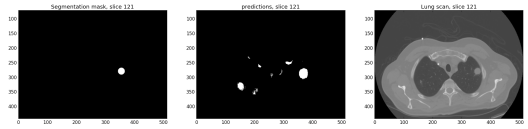

Fig. 3. The result of using the full lung slice image (size 512x512) as input in the network.

in the results section.

3. RESULTS

3.1. Fully convolutional network

This time again, the network showed to be very promising during training: it achieved both a recall and precision of 0.93 on the validation set when testing patch-wise. However, this performance dropped drastically after shifting and stitching the images, achieving a recall of 0.32 and a precision of 0.0008 when classifying pixel wise. So while we did manage to increase the precision, we lost a lot of accuracy.

We did manage do drastically decrease the amount of false positives, as can be seen in figure 3 which only shows a few incorrectly labeled areas. It can also be noted that the found nodule is a bit larger than the actual nodule, just as with the lung segmentation's.

We can probably improve this performance a little bit with some post-processing: do a connected component analysis on the segmentation en remove the small structures. First applying opening and then closing might also connect some disconnected structures which might reduce the amount of detected structures, hopefully improving the precision. We might even try to apply LoG on the resulting image to remove all structures which are clearly note nodule shaped.

3.2. 3D convolutional network

The results for the 3D convolutional network are shown in Table 1.

Table 1. Results for the 3D convolutional network.

Precision	Recall	F_1
0.87	0.92	0.89

These results were computed on a separate validation set (15% of data) without balancing. While the scores look good, with more time we would have liked to perform 10-fold cross-validation and a FROC analysis to enable comparison to other *LUNA2016* participants. We also would have liked to add more data augmentation and to optimize the network structure.

A. CONTRIBUTIONS

Luc Nies: Additional training of the fully convolutional network and wrote the corresponding section. Evaluation of the results of the FCN. Setting up data and software on cluster.

Tom van de Poll: Contributed to segmentation post-processing and wrote the report section on this topic

Harmen Prins: Built connected components filter and FROC analysis. Wrote the corresponding sections.

Steven Reitsma: Built the 3D convolutional network and wrote the report sections on this topic.

Inez Wijnands: Access cluster. Wrote introduction section and editorial revisions.

B. REFERENCES

[1] J. Long, E. Shelhamer & T. Darrell (2015). Fully convolutional networks for semantic segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*: 3431–3440.

[2] K. Simonyan & A. Zisserman (2014). Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*

[3] Q. Dou, H. Chen & P.A. Heng. Lung Nodule Classiffcation with Multi-level Contextual 3D Convolutional Neural Networks. *Method description for LUNA2016*