Computer Aided Diagnosis
Lung segmentation

Luc Nies (s4136748), Tom van de Poll (s4106512), Harmen Prins (s4132297),
Steven Reitsma (s4132343) & Inez Wijnands (s4149696)
Radboud University Nijmegen
11/05/2016

1 Problem description

The goal of the *LUng Nodule Analysis 2016* (*LUNA16*) challenge is to detect lung nodules in low-dose lung CT images. To be able to detect lung nodule candidates, we first have to segment the lungs from the CT images. The data set consists of 888 low-dose CT images, all with corresponding segmentations that can be used as a ground truth. Each image has slices of 512 × 512 pixels and the amount of slices varies per subject.

We used two different approaches. Firstly, we used a deep learning approach using a fully convolutional network, to classify the pixels of the image as either 'lung' or 'no lung'. Secondly, we took a traditional approach where we use region growing for segmentation. These approaches and our results are explained more detailed in the following sections.

2 Fully convolutional network

Since deep learning for image classification has emerged as an extremely successful method, other applications for deep learning have been explored. One of these applications is image segmentation. Long et al. [1] propose a relatively simple method for using a convolutional neural network for image segmentation. One of the challenges is that the output 'label' of the classification is not a simple scalar label but is of the same dimensions as the input. Another challenge is that the process of pixel-by-pixel classification is too slow for big datasets such as the one we are using. Long et al. [1] overcome both challenges with their method of using fully-convolutional networks for image segmentation.

During training time, we use 64x64 pixel patches that are extracted from slices in the horizontal plane of the image. We assign a single label to the patch, defined by the class of the center pixel of the batch (either 0 for 'background' or 1 for 'lung'). The network is fully-convolutional. This means that no fully-connected layers are used, only convolutional and pooling layers. This is done so that images of various sizes can be used as input of the network. If an image is inputed that has a size larger than 64 × 64 pixels, the result from the final layer in the network will be multidimensional in the spatial dimensions instead of a single label prediction. This means that we can use our complete to-be-segmented image as the input of the trained network and the result will be a downsampled segmentation mask. Of course, we need a normal-sized segmentation mask instead of a downsampled one. To overcome the downsampling we use the shift-and-stitch method, also proposed by Long et al. [1]. The shift-and-stitch method performs two steps to upsample. First, it generates multiple outputs (by shifting), and then combines them into one larger output (by stitching). The shifting step generates a new output by shifting all pixels in the image one pixel to the right or down. This shift is applied $n \times m$ times where n and m are the ratio of the output to the target width and height respectively. When stitching the outputs together, the outputs are interleaved such that the pixels of the outputs are shifted in the same way as in the shifting step. So the first row of the output looks like this: $output_1.pixel_1$, $output_2.pixel_1$, …, $output_1.pixel_2$, …, $output_n.pixel_N$. This interleaving ensures that the pixels of the output correspond to the original pixels of the image.

Currently we have no results yet for the fully convolutional network as we are still in the development phase. However, the fully convolutional network can also be used for the next phase of the *LUNA16* challenge (nodule candidate detection), by viewing it as a segmentation problem (we are trying to segment the candidates from the rest of the image). We are therefore continuing the development of this method and are confident that we will get it to work soon.

3 Region growing in MeVisLab

We used a second approach for segmenting the lungs from the CT scans which is the more conventional approach of using region growing, in order to compare the results of our FCN with the current ‘standard’. The biggest challenge of this approach is that in order to use region growing for segmentation, seed points need to be automatically selected inside the area that needs to be segmented. Since this is a difficult task, we circumvent this problem by using region growing in a different way. The following algorithm is used for segmentation:

- Threshold the image with threshold value 1000.
- Apply region growing to the image, four times, with a corner pixel of the first slice as seed point.
- Add the resulting four images to the thresholded image.
- Threshold the resulting image with value 1000. An example result can be seen in figure B.1.
- Invert the resulting image. All that is left are the lungs and some other artifacts. An example can be seen in figure B.2a.
- Do a connected components analysis on the resulting image, and select only the largest component from the image (the lungs).
- Since there are still black holes in the lungs (veins), closing is applied to the image with a 10x10x1 spherical kernel. An example of the resulting segmentation mask can be seen in figure B.2b.

This approach circumvents the need to find specific region growing seed points, since we take the same seed points for each image, namely the four corner pixels of the first slice. The results gathered with this method seem fairly promising. More on the results of using this segmentation method is discussed in the section 4.

4 Results

Since our deep learning method is unfinished, the following results are obtained using the region growing method. We used subset 9 of the data set as our test set. This subset contains 88 subjects in total. We achieved a mean Dice score of 0.955 with a standard deviation of 0.150. A closer inspection of the results revealed that two subjects were very badly segmented, resulting in a Dice score of 0.000. These bad segmentations were the result of a very noisy image, as shown in figure B.3. The third lowest image had a Dice score of 0.889, the rest of the images had a Dice score of 0.970 or higher.

5 Conclusion

Unfortunately we did not manage to get our fully convolutional network to work in time, but we still achieved good results with the conventional region growing approach and showed that this method is a nice standard to compare other approaches with. Most of the segmentations using region growing were close to perfect (with a Dice score higher than 0.97), however an obvious step of improvement would be noise removal since the algorithm fails completely on noisy images. We could definitely improve our performance. If we calculate our mean and variance of the Dice scores on our test set without the noisy images, we would achieve a mean of 0.977 and a standard deviation of 0.011. Thus if we could correctly segment the noisy images, this could be a promising improvement on our current scores.

A Contributions

Luc Nies: Mostly worked on the fully convolutional and made the basic pipeline. Ran the region growing algorithm on the test set and wrote the results and conclusion section.

Steven Reitsma: Looked into implementing U-net. Afterwards assisted in getting the fully convolutional network to train. Wrote report section on fully convolutional networks.

Harmen Prins: Looked into implementing U-net. Implemented shift-and-stitch. Wrote report section on shift-and-stitch.

Inez Wijnands: Contributed to the initial implementation of the fully convolutional network. Contributed to creating the MeVisLab segmentation method. Implemented the evaluation method for the segmentations (Dice score). Created template and structure of report, wrote introduction section and made editorial revisions.

Tom van de Poll: Contributed to the initial implementation of the fully convolutional network. Contributed to creating the MeVisLab segmentation method and responsible for troubleshooting this method. Implemented the method for looping the MeVisLab segmentation method over the data. Wrote report section on region growing.

B Figures

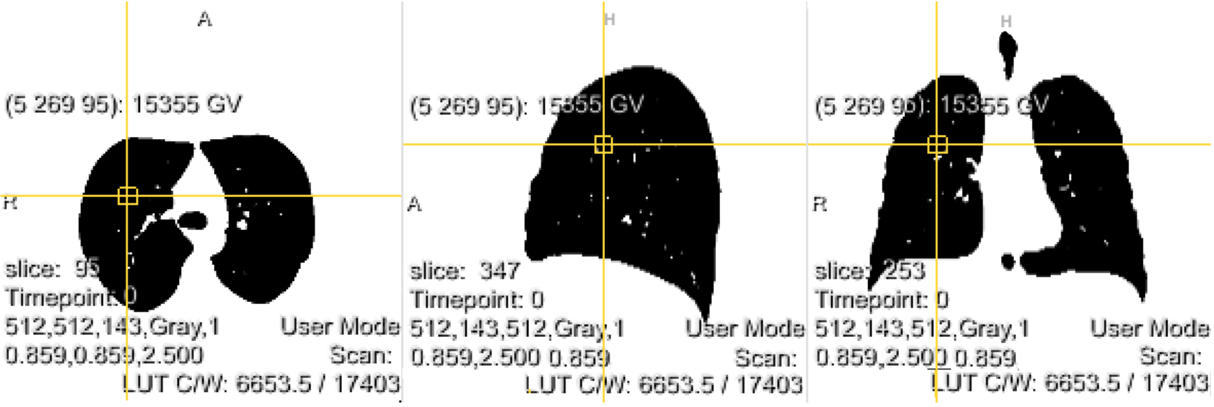
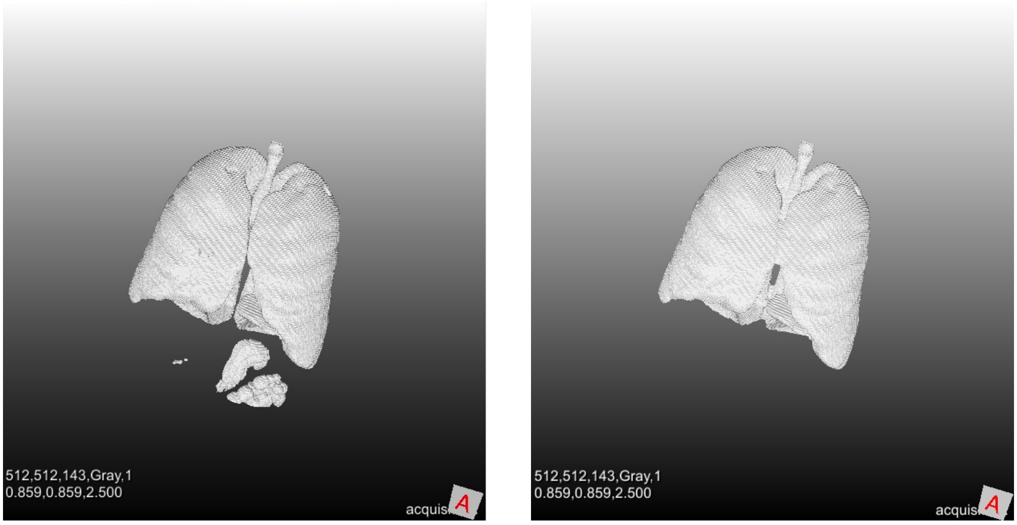


Figure B.1: The thresholded image after region growing.



(a) After the inversion step.

(b) After connected component analysis and closing.

Figure B.2: Results of the segmentation

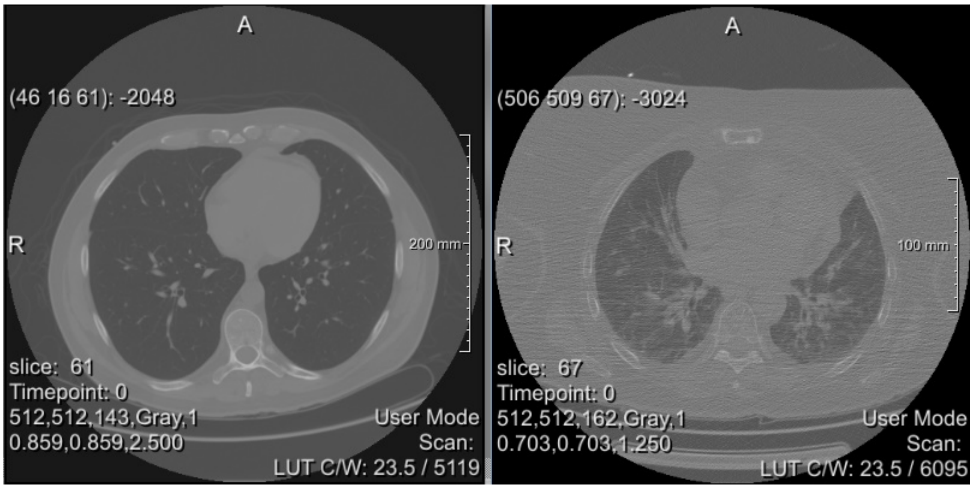


Figure B.3: Clean image on the left, noisy image on the right

References

[1] Jonathan Long, Evan Shelhamer, and Trevor Darrell, *Fully convolutional networks for semantic segmentation*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3431–3440.