Cell nuclei segmentation using deep convolutional neural networks

Proof of concept

Ole-Johan Skrede, Fritz Albregtsen and Håvard E. Danielsen

Department of Informatics — University of Oslo
Institute for Cancer Genetics And Informatics — Oslo University Hospital

Problem description

We look into the problem of segmenting cell nuclei in images of histological sections from colorectal cancer tissue. These images are difficult to segment for a number of reasons:

- We are only interested in non-overlapping, in-focus nuclei.
- The background is very heterogenous and contains disruptive objects.
- The nuclei are often clustered, and have a wide variety of shapes, light intensities and texture.

Method overview

Artificial neural networks have during the last few years been established as the *de facto* standard for solving high-level computer vision problems; segmentation being no exception (Lecun et al., 2015). Broadly speaking, the method consists of a hierarhical classification structure, where feature extractor parameters are computed with mathematical optimization rather than being hand-engineered. This makes the method more data-driven and less *a priori* dependent, which aids the classification of pixels based on subtle, high-level features in the image. The aforementioned properties are highly desirable when segmenting images of colon cancer nuclei.

A common approach to segmentation with neural networks is to extend a full-image classifying network to pixel-wise classification with some kind of upsampling in the deeper layers. Until now, we have e.g. used the well-known *VGG* (Simonyan and Zisserman, 2015) and *ResNet* (He et al., 2015) architectures as base networks, and the results shown here use *atrous* convolution and interpolation in the upsampling (Chen et al., 2016).

The neural network produces probability maps that represent the probability that a pixel belongs to a nucleus. The probability results are binarized using mathematical morphology and random walks.

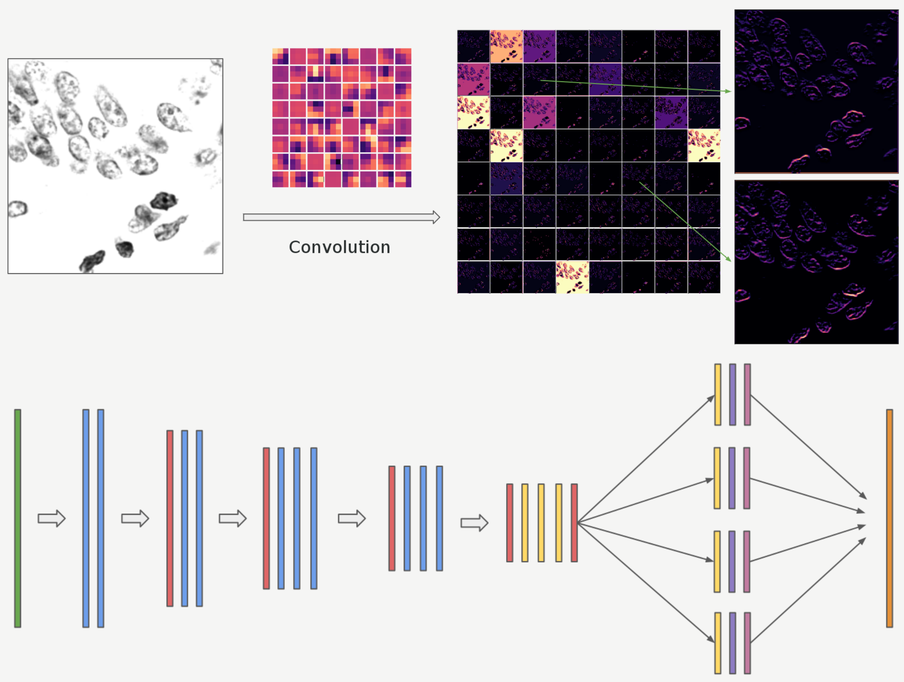


Figure 1: Illustration of convolution (top), and network architecture (bottom).

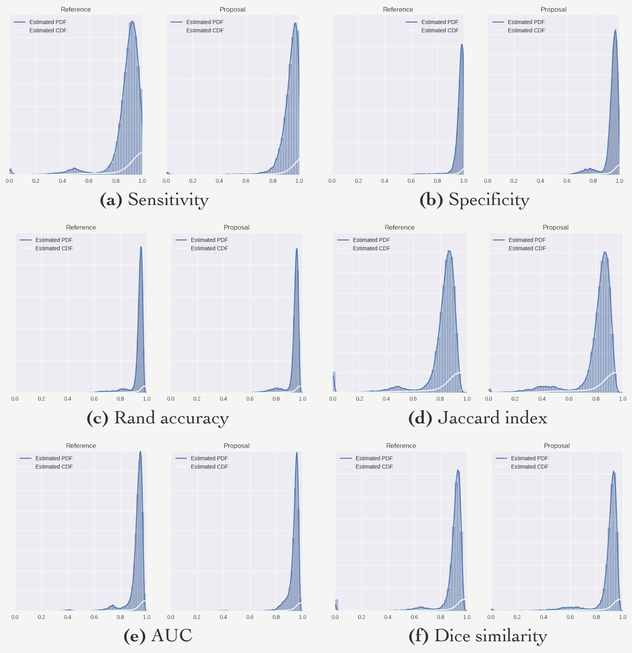
Results

The methods are tested on a dataset consisting of 11814 manually segmented images from 162 different cases. The dataset is partitioned into 2/3 for training, 1/6 for validation, and 1/6 for testing.

The method is tested on a dataset with manually segmented images. We train the network on 8473 images, and evaluate the method on 1661 images (from different patients than in the training set).

The results are obtained using a *NVIDIA Titan X (Pascal)* GPU, and we are able to process about 10 frames per second during training, and about 3 frames per second during evaluation. In training time, we use cropped sections of 321×321 pixels, and the evaluation is carried out on the full frame (1040×1388 pixels).

We evaluate our methods on the test set, and compute several evaluation measures from three major categories: object detection, object-level segmentation agreement, and object-level shape similarity. The difference being that for object-level measures, we compare single nucleis: the reference with the proposal with largest overlap, and the proposal against the reference with the largest overlap. For object detection, we say that objects are true positive, false positive or false negative based on overlap fraction between reference and proposed segmented nuclei, and count these numbers for all the frames.



(a) Sensitivity **(b)** Specificity

(c) Rand accuracy **(d)** Jaccard index

(e) AUC **(f)** Dice similarity

Figure 2: Distributions of various segmentation evaluation measures.

Object detection	**Value**	
Sensitivity	0.65	
F_1-score	0.76	
Object-level segmentation	**Mean**	**Median**
Sensitivity	0.90	0.93
Specificity	0.95	0.97
Rand accuracy	0.93	0.95
Jaccard index	0.80	0.85
AUC	0.93	0.94
Dice similarity	0.88	0.91
Object-level shape similarity	**Mean**	**Median**
Hausdorff distance	3.56	3.16

Table 1: Segmentation evaluation. Object-level segmentation is computed from averages from the distributions shown in Figure 2. Corresponding distributions are used to compute the object detection, and object-level shape similarity measures.

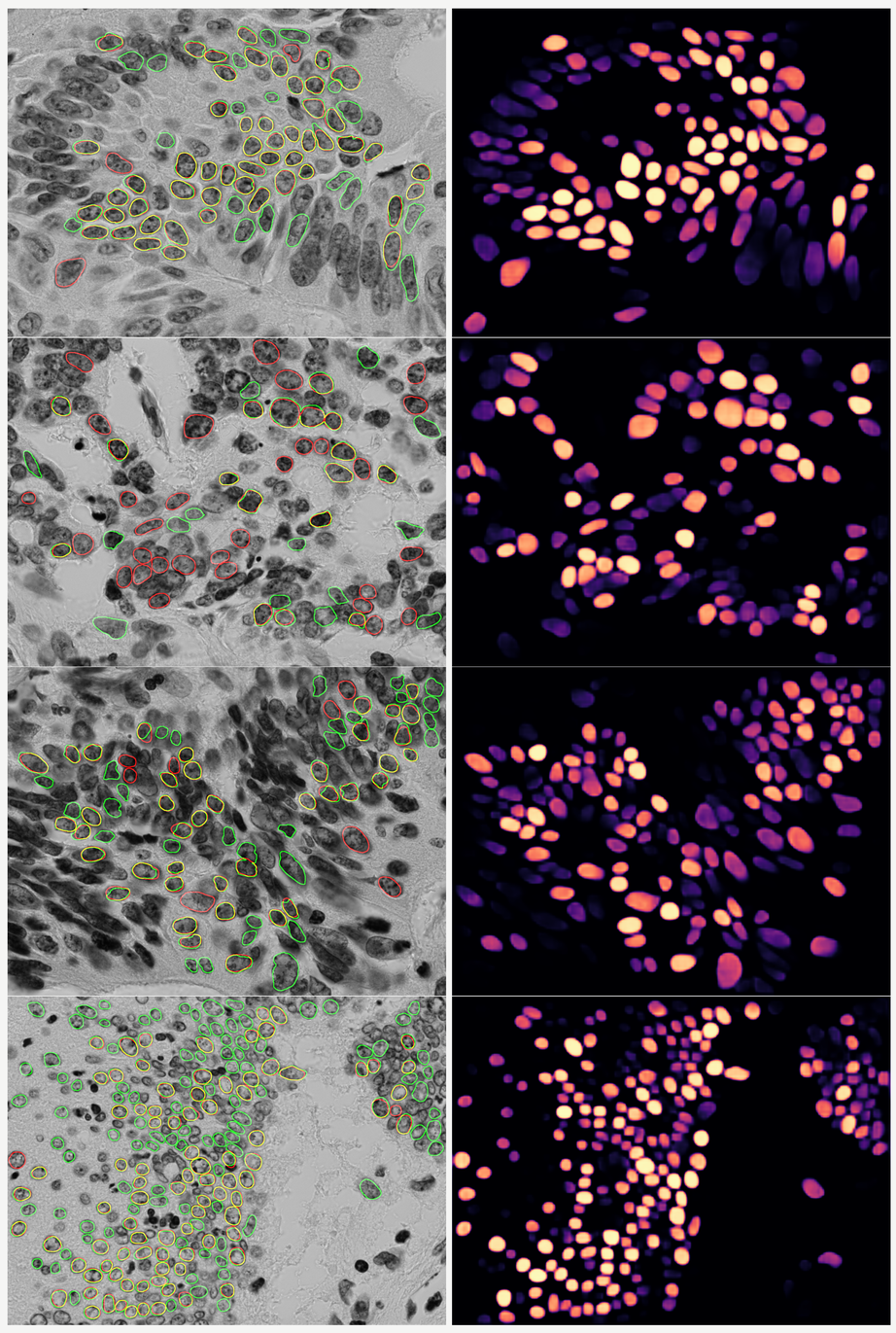


Figure 3: Left: Example input image with outlined annotations: green for reference and red for proposal segmentations, yellow for overlap. Right: Corresponding heatmap representing the probability of a pixel belonging to a nucleus. The proposed segmentations on the images to the left are the results of the binarization of the corresponding images to the right.

Discussion

As a proof of concept, the results are encouraging. The method shown here is able to differentiate between in focus and out-of-focus nuclei, assigning high probabilities to the former, and low to the latter. That being said, there is still room for improvements, both on the neural network, and in the probability binarization process.

References

Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *Iclr*, pages 1–14, 2016. URL `http://arxiv.org/abs/1412.7062`.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *Arxiv.Org*, 7(3):171–180, 2015. ISSN 1664-1078. doi: 10.3389/fpsyg.2013.00124. URL `http://arxiv.org/pdf/1512.03385v1.pdf`.

Yann Lecun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521:436–444, 2015. doi: 10.1038/nature14539.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR 2015*, pages 1–14, 2015. URL `http://arxiv.org/abs/1409.1556`.