Cell Segmentation Proposal Network For Microscopy Image Analysis

Saad Ullah Akram[1,2], Juho Kannala[3], Lauri Eklund[2,4], and Janne Heikkilä[1]

[1]Center for Machine Vision and Signal Analysis, [2]Biocenter Oulu,
[4]Faculty of Biochemistry and Molecular Medicine,
and [4]Oulu Center for Cell-Matrix Research, University of Oulu, Finland
[3] Department of Computer Science, Aalto University, Finland

Abstract. Accurate cell segmentation is vital for the development of reliable microscopy image analysis methods. It is a very challenging problem due to low contrast, weak boundaries, and conjoined and overlapping cells; producing many ambiguous regions, which lower the performance of automated segmentation methods. Cell proposals provide an efficient way of exploiting both spatial and temporal context, which can be very helpful in many of these ambiguous regions. However, most proposal based microscopy image analysis methods rely on fairly simple proposal generation stage, limiting their performance. In this paper, we propose a convolutional neural network based method which provides cell segmentation proposals, which can be used for cell detection, segmentation and tracking. We evaluate our method on datasets from histology, fluorescence and phase contrast microscopy and show that it outperforms state of the art cell detection and segmentation methods.

Keywords: cell proposals, cell segmentation, cell detection, convolutional neural network, deep learning

1 Introduction

In the last few decades advances in automation and optics of microscopes have led to rapid growth in the number and resolution of images being captured, with single experiments in developmental biology producing tera-bytes of data. Often the processes being investigated are subtle and to obtain biologically meaningful quantification, it is necessary to analyze large number of cells in multiple samples. Doing these analyses manually is a very inefficient and tedious use of a biologists time and is dependent on the skill level of biologists leading to very subjective and often non-reproducible results. These factors have increased the importance of automated and semi-automated analysis methods.

Recently, convolutional neural networks (CNNs) have outperformed the state of the art methods in multiple biomedical instance level segmentation challenges [13,5]. These methods either prioritize boundary pixels by increasing their weights [13] or detect them explicitly in addition to binary segmentation [5]. These networks have large receptive fields allowing them to utilize large spatial

context and predict very accurate segmentation masks at instance level. However, since these methods provide only one set of segmentations they can still fail in some ambiguous regions. When analyzing microscopy sequences, temporal information can resolve many of these ambiguities. Cell proposals provide a computationally efficient way of exploiting both temporal and spatial context by reducing the number of alternative hypothesis for a region. So far, cell proposals have been used for cell detection [3,4] and tracking [14,2]. These methods rely on MSER[3], superpixels[14] and blob detection[2] for proposal generation and use hand crafted features to score them. Deep learning has also been applied recently for proposing cell candidate bounding boxes [1], however they use thresholding to obtain segmentation masks, which are not very accurate and it is not trivial to obtain segmentation masks using their approach for images from other microscopy modalities.

In this paper, we propose a CNN based method which first proposes cell bounding boxes using a fully convolutional neural network (FCN) [1]. It then uses a second CNN to predict segmentation masks for each proposed bounding box. Recently, [6] have shown the effectiveness of a similar idea for general object segmentation. Our novel contributions are: (1) a new network for cell segmentation proposal generation and (2) a single network model which can segment cells from multiple microscopy modalities. We compare our method's proposals with proposals from two state of the art cell detection methods and our cell detections with three state of the art cell segmentation/detection methods and show that our method outperforms them on three different datasets which represent cells with varying appearance and imaging conditions.

2 Method

Our method has two stages, in the first stage (Sec. 2.1) a convolutional neural network (CNN) proposes cell bounding boxes along with their scores, i.e. probability of them being a cell. In the second stage (Sec. 2.2), a second CNN utilizes the proposed bounding boxes to predict segmentation masks for cells.

2.1 Proposal Bounding Boxes

Our first network, shown in the top half of Fig. 1, is modified from the network in [1]. Briefly, it predicts k bounding boxes and their scores at each pixel in the last feature map, removes duplicate proposals using non-maxima suppression and returns the remaining N bounding boxes as the cell proposals. Details of how this network proposes bounding boxes [1,12] and how it is trained are available in [1].

2.2 Proposal Segmentation

Network Structure: Our second network, which is used for predicting segmentation masks is shown in the bottom half of Fig. 1. This network takes the image

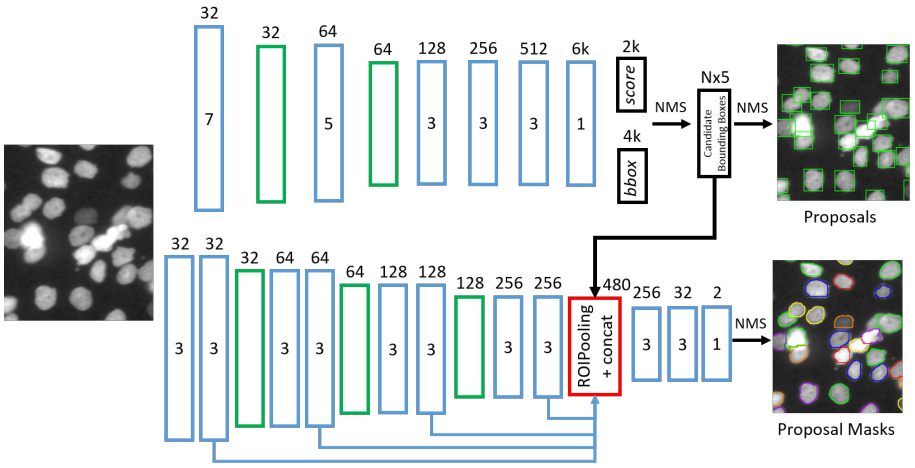


Fig. 1: Cell Segmentation Proposal Network: Top half shows the first network, which proposes N bounding boxes and their scores. Bottom half shows the second network which generate segmentation masks for the N proposals. Convolution (filter size is shown in the box), max-pooling, and ROI-Pooling + concatenation layers, with the number of feature maps on top of each layer, are shown. Proposed bounding boxes and segmentation masks after non-maxima suppression (NMS) are shown for a selected area from *Fluo-N2DL-HeLa* dataset.

and N proposed bounding boxes from first network as its inputs and predicts N segmentation masks of size 25x25 each. First part of this network contains eight convolution layers which are applied to the whole image. Second part uses region of interest (ROI) pooling layers [7] to extract fixed size (25x25) features maps from four sets of feature maps as shown in Fig. 1. ROI-Pooling layer uses adaptive max pooling of the region inside the bounding box in a given feature map to extract a fixed sized feature map. The ROI-pooled feature maps are concatenated to obtain a feature map of size 25x25x480 for each proposed cell bounding box. It is important to select features from layers at different depth so that the network can use both coarse high level information to predict which regions belong to the cell being segmented and fine low level information to predict accurate localization of cell boundaries. The fixed size feature map extracted for each proposed bounding box is used by a small sub-network, consisting of three convolution layers, which picks the appropriate combination of features from different depths so that it can better leverage both fine and coarse information. For each proposed bounding box, this network outputs a 25x25 probability map, which is resized back to the original bounding box size using bicubic interpolation, thresholded and largest connected component is used as the segmentation mask.

All convolutional layers use a stride of 1 pixel and padding to preserve the feature map size. ReLu non-linearities are used after all convolutional layers

except the last one. Local contrast normalization layers with same normalization parameters as ZF model [15] are used after first eight convolutional layers. All max-pooling layers use a filter of size 3x3, padding of 1 and stride of 2 to reduce feature map size and increase the receptive field.

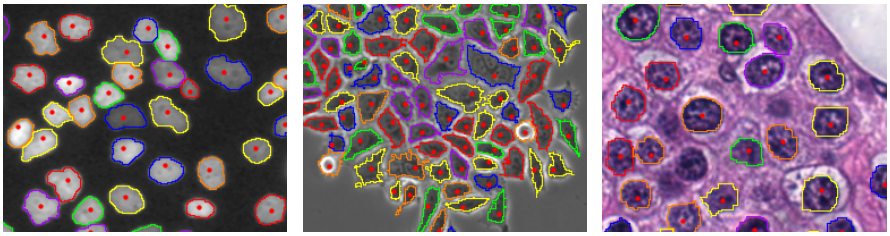
Training: The bounding boxes proposed by the first network are used to train the segmentation network. The overlap of these bounding boxes with the ground truth bounding boxes is computed and if the intersection over union (IoU) overlap is greater than 0.5, then these boxes are used for training; otherwise, they are ignored. For each bounding box, a 25x25 binary segmentation mask is used as the target output during training. This mask is obtained by cropping the region inside the predicted bounding box, resizing this region to 25x25 using nearest-neighbor interpolation and labeling all pixels except those of the largest cell within that box as background. The loss function used for training is a pixel-wise softmax log loss.

Implementation Details: To use the exact same network for all datasets, we resize the images in each dataset so that the mean cell bounding box is $\sim$25x25 pixels. Images in some datasets are quite large so we split these images in equal sized smaller images so that no image dimension is larger than 500 pixels to reduce GPU memory requirement. Since there are very few training images, we use horizontal and vertical flips, and 90 degree rotations to augment training data.

We initialize our segmentation network by picking weights randomly from a Gaussian distribution with zero-mean and 0.01 standard deviation. We use learning rate of 0.0001 for first 40k iterations then it is reduced to 0.00001 for next 10k iterations.

2.3 Cell Detection and Segmentation

There are not many widely used cell proposal generation method which makes it difficult to compare our performance. However, there are few popular cell segmentation and detection methods available publicly. In order to compare our method against these methods, we use stronger non-maxima suppression to remove most duplicate proposals and use the selected proposals (***Ours-Greedy***) as cell detections and their masks as cell segmentations of our method. The IoU and score threshold values which maximize average precision and f-score on the training data are used for each model. Since we use IoU > 0, this can result in some pixels having multiple labels, we assign these pixels the label of the cell (proposal) with the highest score. We would like to point out that these detections can be considered as a weak baseline when using our proposals for cell detection or segmentation. Better performance can be obtained by using dynamic programming [3] or integer linear programming [4], which can select the optimal set of proposals.



(a) Fluo-N2DL-HeLa (b) PhC-HeLa (c) Hist-BM

Fig. 2: Datasets: Ground truth cell markers (•) and boundaries are marked.

3 Experiments

3.1 Data-sets

We evaluate our method on three datasets *Fluo-N2DL-HeLa* [10], *PhC-HeLa* [3] and *Hist-BM* [8]. Figure 2 shows one sample region from each dataset along with the Ground Truth (GT) segmentation masks and cell markers.

Fluo-N2DL-HeLa data-set is from ISBI cell tracking challenge [10] and it contains 2 time-lapse sequences (92 frames each) of fluorescent HeLa cells cultured and imaged on two dimensional surface. The GT for this data-set contains markers for all 34,060 cells in all frames and segmentation masks for all 874 cells in 4 frames. It also includes segmentation masks for few cells in other frames but since those frames are not exhaustively segmented, we do not use them. Some of the challenges with this data-set are: many cell clusters, frequent cell divisions, low contrast, variation in cell sizes and intensities.

PhC-HeLa data-set [3] consists of 22 phase contrast images of cervical cancer colonies of HeLa cells, split in 2 sets (training and test). The GT for this dataset consists of cell markers for all 2,228 cells. Challenges with this dataset include high variation in cell shapes and sizes, missing cell boundaries, and high cell density.

Ground truth segmentation masks for this dataset are obtained by greedily selecting the largest MSER region for each ground truth marker under the constraints that the selected MSER region contains only one cell marker, has little overlap with previously selected regions and markers which are inside smaller regions are processed first.

Hist-BM data-set [8] consists of 11 images stained with Hematoxylin and Eosin of human bone marrow from eight different patients. The ground truth for this dataset consists of markers for all 4,202 cell nuclei and ambiguous regions. We split this dataset in two sets, with first five images in set 1 and rest in set 2.

Ground truth segmentation masks for this dataset are generated using multi-label graph cuts. Terminal edge costs are set using cell and background Gaussian mixture models, learnt from pixels within radius of 6 from markers and pixels outside radius of 20 from all markers respectively. Cells are divided into 7 sets

so that cells adjacent to each other have a different label. Then pixels within radius of 6 from a marker are fixed to the terminal node representing that cell's label to separate cells in contact with each other.

3.2 Evaluation Criteria

Average Precision (AP): We use average precision (AP) - area under precision-recall curve - to evaluate segmentation proposals. Proposals are first sorted by their score, then a proposal is counted as true positive (TP) if it has intersection over union overlap (IoU) > 0.5 with any unmatched ground truth (GT) cell segmentation mask, otherwise it is counted as false positive (FP). GT cells which remain unmatched are counted as false negative (FN). We obtain a pair of recall ($R = \frac{TP}{TP+FN}$) and precision ($P = \frac{TP}{TP+FP}$) values after evaluating each proposal.

F-Score (F1): To evaluate detection performance we use same criteria as above to obtain recall (R) and precision (P) using all cell detections and compute F-Score ($F1 = \frac{2 \cdot P \cdot R}{P+R}$).

SEG: We evaluate accuracy of segmentation masks using the SEG measure, based on Jaccard similarity index, used in ISBI cell tracking challenge [10]. A detection (D) is matched with a GT cell (G) if and only if it contains more than half pixels of that GT cell, i.e. $|D \cap G| > 0.5 \cdot |G|$. For each GT cell and its matched detection, Jaccard similarity index is computed using $J(G, D) = \frac{|D \cap G|}{|D \cup G|}$. SEG is the mean of Jaccard similarity index of all GT cells and ranges between 0 to 1.

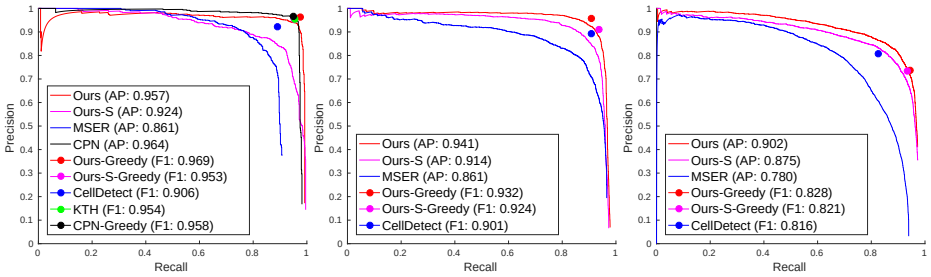
We use the SEG measure as defined above to evaluate proposal masks. When evaluating proposals, some pixels can be inside multiple proposals and as a result there might be multiple proposals which satisfy $|D \cap G| > 0.5 \cdot |G|$. We compute Jaccard similarity index for these proposals and match the GT cell with the best proposal, i.e. one having the highest Jaccard similarity index.

Implementation Details: All three datasets are split in two sets as described in Sec.3.1. One set is used for training the methods and the other for testing; this is repeated for both sets. Same non-maxima suppression settings (IoU=0.5) are used for all methods to remove duplicate proposals. The proposals from both sets are combined, sorted by their score and evaluated as either TP or FP. The detection results are combined similarly and SEG and F-Score computed for the whole dataset.

3.3 Baseline

We compare our method (*Ours*) with two cell proposal generation methods *MSER* [3,11] and *CPN* [1], and three cell detection and segmentation methods, *KTH* [9], *CellDetect*[1][3] and *CPN-Greedy* [1]. **CPN** uses a method similar to our first stage to propose cell candidates and **CPN-Greedy** uses stronger non-maxima suppression to obtain cell detections from CPN proposals. **CellDetect**

[1] `http://www.robots.ox.ac.uk/~vgg/software/cell_detection/`



(a) Fluo-N2DL-HeLa (b) PhC-HeLa (c) Hist-BM

Fig. 3: Precision vs Recall (at IoU = 0.5) for all three datasets. Average Precision (AP) and F-Score (F1) are shown in the legend.

uses MSER regions [11] as proposals, represents each proposal using hand crafted features, and uses structured SVM to score them. We use these scores to rank MSER regions during evaluation of **MSER** proposals. CellDetect then selects optimal set of MSER regions using dynamic programming and uses these selected regions as cell detections. **KTH** uses a band pass filter followed by thresholding to segment cells, watershed transform is then used to split cell clusters, and finally tracking is used to correct errors in segmentation. KTH software[2] does not provide access to segmentation results so we use their segmentation masks after the tracking stage during evaluation.

3.4 Results

Fig. 3 shows the precision-recall curves for the proposal generation methods and precision-recall values for cell detection methods. Precision-recall curve for our proposals not only remains significantly above the curve for MSER proposals, it is even slightly above the precision-recall values of CellDetect detections for all datasets. For Fluo-N2DL-HeLa dataset, CPN has better precision for all but very high recall values, however it has slightly lower recall than our method. Even though CPN uses simple thresholding, it is able to obtain good performance at IoU=0.5 as thresholding can provide a coarse mask. For higher IoU values, its performance degrades and gap in our method and CPN increases.

Table 1 compares the segmentation quality (SEG) of our method's proposals against other baselines and shows that our method's proposal masks are consistently better than MSER and CPN proposals for all three datasets. Table 1 also lists SEG and F-Score (detection performance) values for all cell detection methods. Our method has better detection and segmentation performance compared to other detection methods. The difference in both detection and segmentation performance of our method and CellDetect is quite large for all three datasets.

[2] `http://codesolorzano.com/celltrackingchallenge/Cell_Tracking_Challenge/KTH-SE.html`

Table 1: Cell segmentation (SEG) and detection (F-Score) results.

	Fluo-N2DL-HeLa		PhC-HeLa		Hist-BM	
	SEG	F-Score	SEG	F-Score	SEG	F-Score
Detections						
Ours-Greedy	**0.858**	**0.969**	0.761	**0.932**	**0.804**	**0.828**
Ours-S-Greedy	0.815	0.953	**0.769**	0.924	0.789	0.821
CellDetect [3]	0.734	0.906	0.717	0.901	0.682	0.816
KTH [9]	0.852	0.954	-	-	-	-
CPN-Greedy [1]	0.808	0.958	-	-	-	-
Proposals						
Ours	**0.874**	-	**0.818**	-	**0.823**	-
Ours-S	0.865	-	0.807	-	**0.823**	-
MSER [3]	0.757	-	0.779	-	0.768	-
CPN [1]	0.831	-	-	-	-	-

Our method also outperforms KTH method slightly, which has the best segmentation performance on Fluo-N2DL-HeLa dataset in ISBI cell tracking challenge [10] and uses tracking stage to correct errors in segmentation.

Cell boundaries produced by our method are quite accurate; most segmentation errors are due to (1) errors in localization of bounding boxes, which sometimes clips parts of cells and (2) the failure of segmentation stage to ignore parts of other cells in the proposed bounding box. We tried using a dilated proposal bounding box but it led to lower performance as even though it reduced clipping errors, the errors due to failure to ignore other cells increased. We also experimented with a fully connected layer at the end of the network, which was able to ignore other cells better but it produced coarse masks and did not improve performance.

Two of the challenges of biomedical image analysis are the large variation between sequences and lack of ground truth. Often a method is trained or designed for a particular set of sequences and works well for images captured in a narrow range of imaging conditions. Having a general method which can cope with a wider range of imaging settings is very important as it is not always feasible to design or tweak existing methods for the sequences being analyzed. As a small step towards achieving this goal, we train a single network model (Ours-S) using all three datasets. Equal number of training samples were used from each dataset. This model has slightly lower performance on two of the three datasets but on Fluo-N2DL-HeLa dataset its performance decreases considerably. Even with this lower performance it outperforms CellDetect on all three datasets.

4 Conclusions

In this paper, we have presented a deep learning based method for proposing cell segmentation candidates and demonstrated that it can produce excellent pro-

posals for three different microscopic modalities. We have compared our method against state of the art cell detection and segmentation methods and shown that our method outperform them on common evaluation metrics. We have also presented a single model trained on all three datasets and shown that its performance does not degrade significantly, which is promising and indicates that a single model for cell detection and segmentation can be trained without compromising too much on performance. Performance of such a model may even improve if it is trained on datasets which are imaged in somewhat similar imaging conditions; this is something we plan to investigate in future. We also plan to use our method's proposals for cell detection and tracking. Code is available at `https://github.com/SaadUllahAkram/CellProposalNetwork`.

References

1. Akram, S.U., Kannala, J., Eklund, L., Heikkilä, J.: Cell Proposal Network For Microscopy Image Analysis. In: ICIP (2016) 2, 6, 8
2. Akram, S.U., Kannala, J., Eklund, L., Heikkilä, J.: Joint Cell Segmentation And Tracking Using Cell Proposals. In: ISBI (2016) 2
3. Arteta, C., Lempitsky, V., Noble, J.A., Zisserman, A.: Learning to Detect Cells Using Non-overlapping Extremal Regions. In: MICCAI (2012) 2, 4, 5, 6, 8
4. Bise, R., Sato, Y.: Cell Detection From Redundant Candidate Regions Under Nonoverlapping Constraints. IEEE Transactions on Medical Imaging (2015) 2, 4
5. Chen, H., Qi, X., Yu, L., Heng, P.A.: DCAN: Deep Contour-Aware Networks for Accurate Gland Segmentation. In: CVPR (2016) 1
6. Dai, J., He, K., Sun, J.: Instance-aware Semantic Segmentation via Multi-task Network Cascades. In: CVPR (2016) 2
7. Girshick, R.: Fast R-CNN. In: ICCV (2015) 3
8. Kainz, P., Urschler, M., Schulter, S., Wohlhart, P., et al.: You Should Use Regression to Detect Cells. In: MICCAI (2015) 5
9. Magnusson, K.E.G., Jalden, J.: A Batch Algorithm using Iterative Application of the Viterbi Algorithm to Track Cells and Construct Cell Lineages. In: ISBI (2012) 6, 8
10. Maška, M., Ulman, V., Svoboda, D., Matula, P., et al.: A Benchmark for Comparison of Cell Tracking Algorithms. Bioinformatics (2014) 5, 6, 8
11. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust Wide-Baseline Stereo from Maximally Stable Extremal Regions. Image and Vision Computing (2004) 6, 7
12. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In: NIPS (2015) 2
13. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation. In: MICCAI (2015) 1
14. Schiegg, M., Hanslovsky, P., Haubold, C., Koethe, U., et al.: Graphical Model for Joint Segmentation and Tracking of Multiple Dividing Cells. Bioinformatics (2015) 2
15. Zeiler, M.D., Fergus, R.: Visualizing and Understanding Convolutional Networks. In: ECCV (2014) 4