Deep Networks with Shape Priors for Nucleus Detection

ICIP 2018 - Athens, Greece

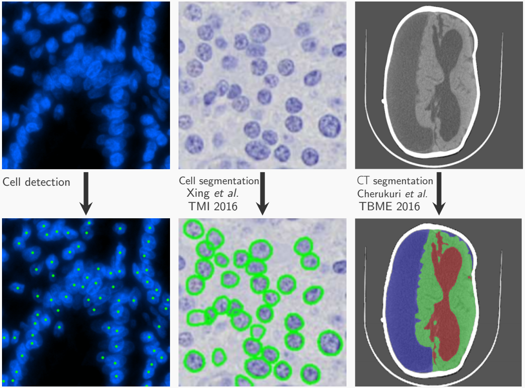
Mohammad Tofighi, Tiantong Guo, Jairam K.P. Vanamala, Vishal Monga

Department of Electrical Engineering
Pennsylvania State University

Wednesday 10th October, 2018

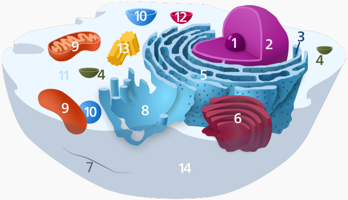
Table of contents

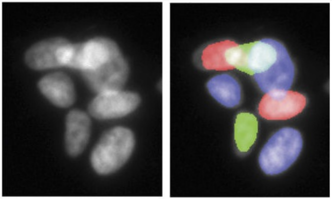
Medical Image Processing Applications



Cell Nucleus Detection and Challenges

- Morphological methods [1]
- Challenges:
  - Overlapping cells,
  - Different nucleus shapes
- Deep learning based methods are proposed [2]
- Pros: Learned features can boost the performance
- Cons: Fail in challenging cases; naive learning of features
- Solution: Learn better! Guided by the expert domain knowledge



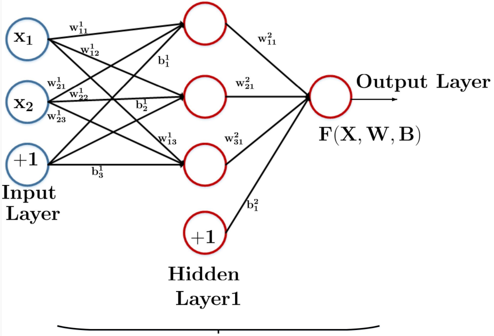


---

[1] Y. Al-Kofahi *et al.*, IEEE TBME 2010
[2] A. Cruz *et al.*, MICCAI 2013

Introduction to Neural Networks

- Deep learning models inspired by the biological neural networks.
- They have been used for several applications: [3] [4]
  - Classification: image segmentation, object detection, speech recognition, ...
  - Regression: Image super-resolution, denoising, ...

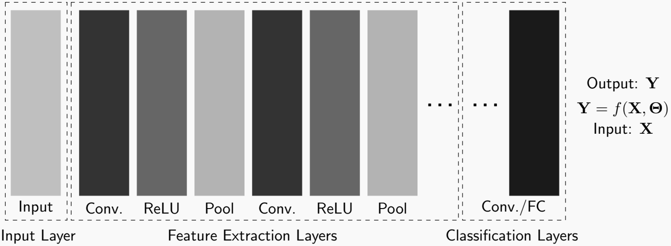


Simple Neural Network

[3] J. Long *et al.*, CVPR 2015
[4] Y. LeCun *et al.*, Nature 2015

Introduction to Neural Networks

- One mostly used NN: Convolutional Neural Network (CNN)
- A mapping $\mathbf{Y} = f(\mathbf{X}, \boldsymbol{\Theta})$ is learned by minimizing the cost function $E(f(\mathbf{X}, \boldsymbol{\Theta}), \mathbf{G})$ between the output $\mathbf{Y}$ and the ground truth $\mathbf{G}$
- Using a stochastic gradient descent method and an error back-propagation algorithm [5] [6]

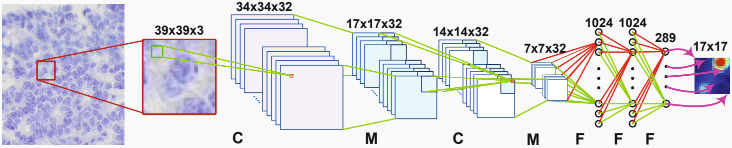


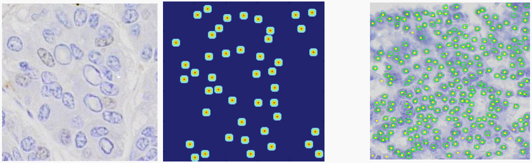
[5] D.E. Rumelhart *at al.*, Nature, 1986

[6] Y. Lacun *at al.*, Neural Computation, 1989

Review: CNN/Deep Learning for Cell Nuclei Detection

- Recent works on cell detection based on CNN/Deep Learning technique: SC-CNN [7], SR-CNN [8] SSAE [9], LIPSyM [10]





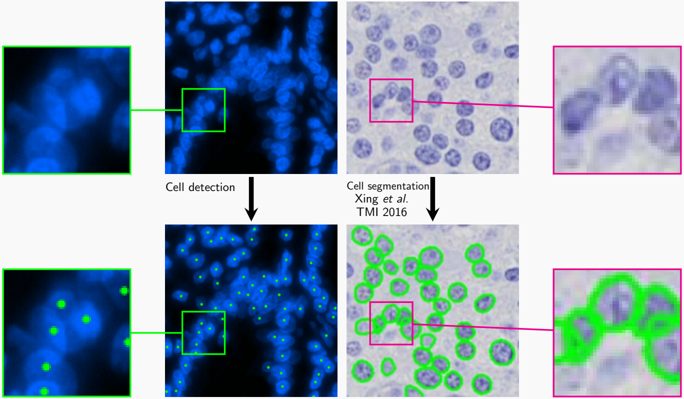
Original image **Proximity mask** **Detection**

---

[7] Sirinukunwattana *et al.*, TMI 2016
[8] Xie *et al.*, MICCAI 2015
[9] Xu *et al.*, TMI 2016
[10] Kuse *et al.*, JPI 2011

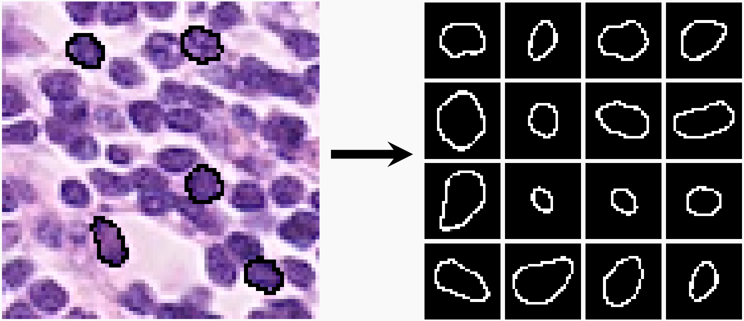
Challenges in Cell Nuclei Detection

- Overlapping cell: false positive and false negative detections
- Varying shapes of the nuclei: decrease detection and segmentation accuracy

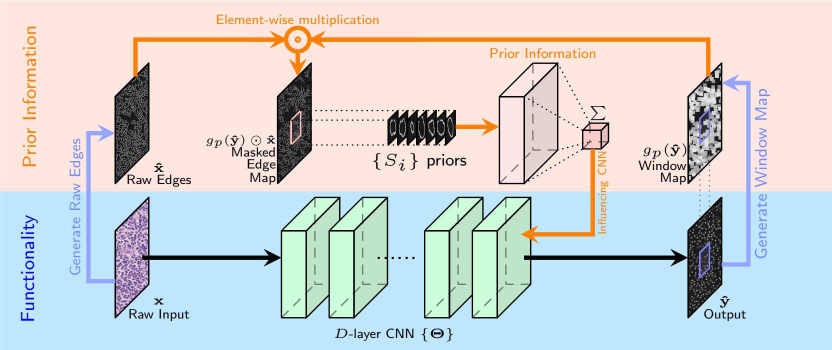


Building Informative Priors

- Our Solution: Shape Prior Guided CNN



Proposed Method: SP-CNN Structure



Cost Function

- Suppose the shape set as $\mathbf{S} = \{\mathcal{S}_i | i = 1, 2, \ldots, n\}$
- CNN cost function

$$\boldsymbol{\Theta} = \arg\min_{\boldsymbol{\Theta}} \|f(\mathbf{x}; \boldsymbol{\Theta}) - \mathbf{y}\|_2^2 \tag{1}$$

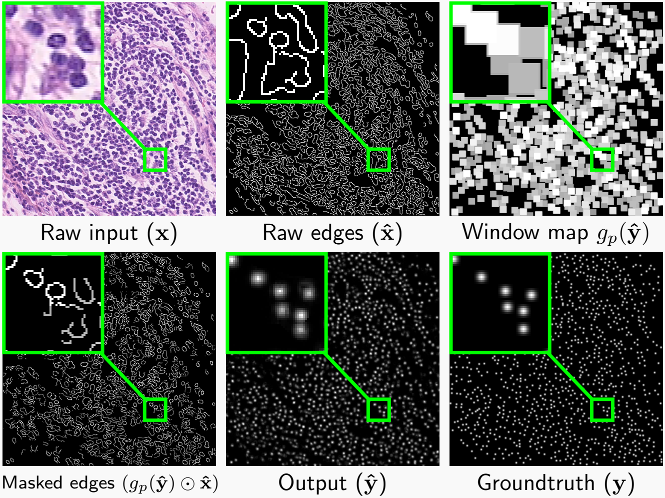
- Cost term of the shape priors

$$\sum_{i=1}^{n} \|(g_p(\hat{\mathbf{y}}) \odot \hat{\mathbf{x}}) * \mathcal{S}_i\|_2^2 \tag{2}$$

- Overall, the cost function of the SP-CNN is given as:

$$\boldsymbol{\Theta} = \arg\min_{\boldsymbol{\Theta}} \|f(\mathbf{x}; \boldsymbol{\Theta}) - \mathbf{y}\|_2^2 - \lambda \sum_{i=1}^{n} \|(g_p(\hat{\mathbf{y}}) \odot \hat{\mathbf{x}}) * \mathcal{S}_i\|_2^2 \tag{3}$$

SP-CNN Visual Illustrations

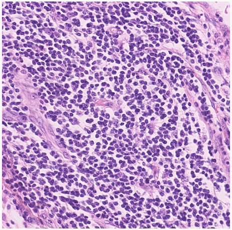
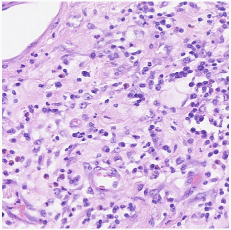
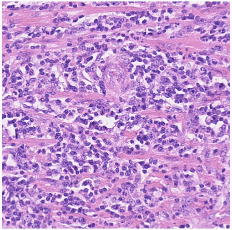


Shape Priors for Convolutional Neural Networks (SP-CNN)

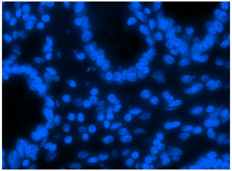
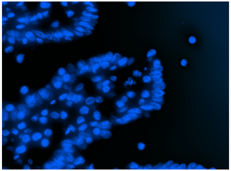
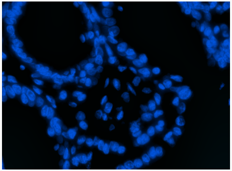
- Train CNN using the input image and the ground truth label
- Using the CNN output, put masks on each detected local maxima (done by maxpooling): $g_p(\hat{\mathbf{y}})$
- Extract raw edge image from the raw input image using simple Canny edge detection filter: $\hat{\mathbf{x}}$
- Element-wise multiplication: $(g_p(\hat{\mathbf{y}}) \odot \hat{\mathbf{x}}) \;\Rightarrow$ masked edge map
- Masks out the edges from $\hat{\mathbf{x}}$ that are surrounding the detected location in $\hat{\mathbf{y}}$: delete non-cell edges
- Convolve masked edge map with each of the shapes in set $\mathbf{S}$: shape prior information
- Add them up and feed it back to CNN

Dateset & Assessment Methods

- UW Dataset [11]: 100 H&E stained histology images of colorectal adenocarcinomas ($\sim$30k cells)

- PSU Dataset – EE & Department of Food Science: 120 Colonic Mucosa images ($\sim$26k cells)

- Test-Train split: UW $(50 - 50$, consistent with [11]), PSU $(20 - 80)$.
- For assessment Recall (R), Precision (P), and F1 Score are used:

$P = \frac{TP}{TP+FP}$, $R = \frac{TP}{TP+FN}$, and $F_1 = \frac{2PR}{P+R}$

[11] K. Sirinukunwattana *et al.* – TMI'16

Assessment Methods & Experimental Results

- All the results are obtained with same assessment procedure:

Table: Nucleus detection results for dataset of SC-CNN [12]

UW Dataset	Precision	Recall	F1 score
SP-CNN	**0.803**	**0.843**	**0.823**
SC-CNN [15]	0.781	0.823	0.802
CP-CNN [15]	0.697	0.687	0.692
SR-CNN[13]	0.783	0.804	0.793
SSAE[14]	0.617	0.644	0.630
LIPSyM[15]	0.725	0.517	0.604
CRImage[16]	0.657	0.461	0.542
PSU Dataset	**Precision**	**Recall**	**F1 score**
SP-CNN	**0.854**	**0.871**	**0.863**
SC-CNN [15]	0.821	0.830	0.825
SR-CNN [16]	0.797	0.805	0.801
SSAE [17]	0.665	0.634	0.649

[12] Sirinukunwattana *et al.*, TMI 2016
[13] Xie *et al.*, TMI 2016
[14] Xu *et al.*, TMI 2016
[15] Kuse *et al.*, JPI 2011
[16] Yuan *et al.*, Sci. Trans. Med. 2012

Precision-Recall Curve for Choosing the Optimal Threshold - UW Dataset

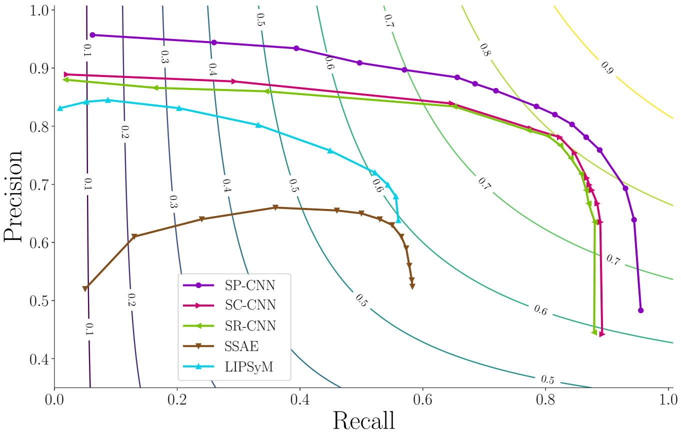


Figure: SC-CNN [17], SR-CNN [18] SSAE [19], LIPSyM [20]

[17] Sirinukunwattana *et al.*, TMI 2016
[18] Xie *et al.*, MICCAI 2015
[19] Xu *et al.*, TMI 2016
[20] Kuse *et al.*, JPI 2011

Precision-Recall Curve for Choosing the Optimal Threshold - PSU Dataset

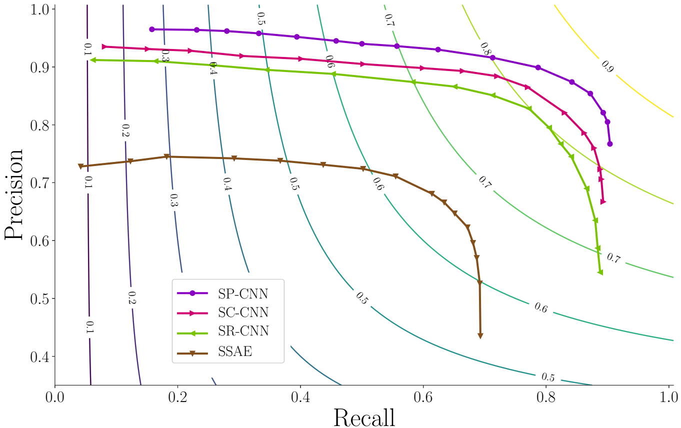


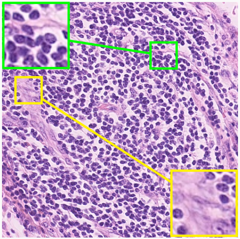
Figure: SC-CNN [21], SR-CNN [22] SSAE [23]

[21] Sirinukunwattana *et al.*, TMI 2016

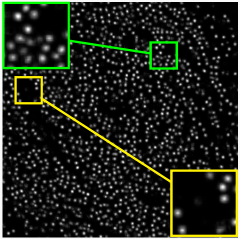
[22] Xie *et al.*, MICCAI 2015

[23] Xu *et al.*, TMI 2016

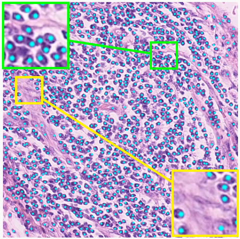
Example Results



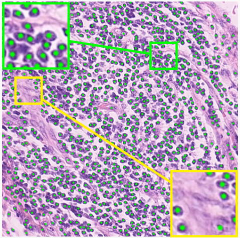
(a) Example image (partial image)



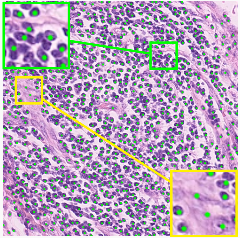
(b) Output of SP-CNN



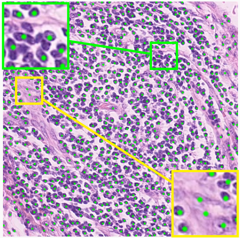
(c) Groundtruth



(d) Detection by SP-CNN; F1-score = 0.843

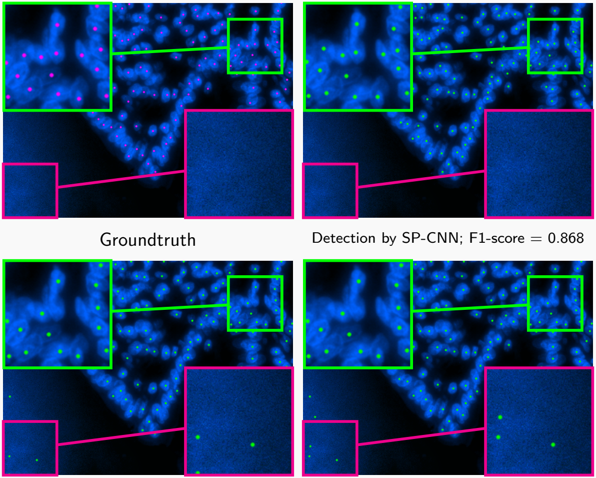


(e) Detection by SC-CNN; F1-score = 0.801



(f) Detection by SR-CNN; F1-score = 0.784

Example Results



Groundtruth

Detection by SP-CNN; F1-score = 0.868

Detection by SC-CNN; F1-score = 0.815

Detection by SR-CNN; F1-score = 0.809

Conclusion

- Shape prior guided convolutional neural networks help improve the performance of cell nuclei detection.
- Future research will be focused on designing data adaptive learning shapes.

Thanks For Your Attention!

Back-propagation Analysis of SP-CNN

- Training a neural network with gradient descent requires the calculation of the gradient of the cost function.
- The cost function of SP-CNN is as follows:

$$E(\mathbf{x};\boldsymbol{\Theta}) = \|f(\mathbf{x};\boldsymbol{\Theta}) - \mathbf{y}\|_2^2 - \lambda \sum_{i=1}^{n} \|(g_p(\hat{\mathbf{y}}) \odot \hat{\mathbf{x}}) * \mathcal{S}_i\|_2^2 \qquad (4)$$

- It has two terms: fidelity cost term and the cost term corresponding to the shape priors.
- Detection fidelity cost term is:

$$L = \|f(\mathbf{x};\boldsymbol{\Theta}) - \mathbf{y}\|_2^2, \qquad (5)$$

- The cost term for shape priors is:

$$P = -\lambda \sum_{i=1}^{n} \|(g_p(\hat{\mathbf{y}}) \odot \hat{\mathbf{x}}) * \mathcal{S}_i\|_2^2. \qquad (6)$$

Back-propagation for Fidelity Cost Term

For detection fidelity cost term the back-propagation is performed by:

- At iteration step t, weights are updated by:

$$\boldsymbol{\Theta}^{t+1} = \boldsymbol{\Theta}^{t} - \eta \frac{\partial L}{\partial \boldsymbol{\Theta}^{t}} \quad (7)$$

  where, η is learning rate for the stochastic gradient descent method and $\boldsymbol{\Theta}^{t}$ is the values of weights at previous iteration.
- $\boldsymbol{\Theta}$ consists of weights from D convolutional layers, following gradients are to be computed: $\frac{\partial L}{\partial W^d}$, where $d = 1, ..., D$.
- For simplicity, we focus on filters and assume that output image $\hat{\mathbf{y}}$ is of dimension $N \times N$.
- For computation of the gradients of the weights at last layer:

$$\frac{\partial L}{\partial W^d} = -(\mathbf{y} - \hat{\mathbf{y}}) \cdot \frac{\partial \hat{\mathbf{y}}}{\partial W^d} \quad (8)$$

- $\frac{\partial \hat{\mathbf{y}}}{\partial W^d}$ is obtained according to [24].

[24] Y. LeCun *et al.*, Proc. of the IEEE, 1998

Back-propagation for Shape Priors Cost Term

To carry the shape priors cost term into the $\boldsymbol{\Theta}$, we need to update Eq. (7) accordingly. Examining closely of the Eq. (6), we can re-write it as:

- Updated Eq. (7) will be:

$$\boldsymbol{\Theta}^{t+1} = \boldsymbol{\Theta}^{t} - \eta \frac{\partial L}{\partial \boldsymbol{\Theta}^{t}} - \eta \frac{\partial P}{\partial \boldsymbol{\Theta}^{t}}. \tag{9}$$

- Since, our network parameter $\boldsymbol{\Theta}$ consists of weights from D convolutional layers, following gradients are to be computed: $\frac{\partial P}{\partial \mathbf{W}^{l}_{m,n}}$, where $l = 1, ..., D$ and $\mathbf{W}$ is of dimension $k_1 \times k_2$ has m by n as the iterators.
- The equations for computing the gradients of weights at last layer are given by:

$$\frac{\partial P}{\partial \mathbf{W}^{l}_{m',n'}} = \sum_{i=0}^{N-k_1} \sum_{j=0}^{N-k_2} \frac{\partial P}{\partial x^{l}_{i,j}} \frac{\partial \mathbf{x}^{l}_{i,j}}{\partial \mathbf{W}^{l}_{m',n'}} = \sum_{i=0}^{N-k_1} \sum_{j=0}^{N-k_2} \delta^{l}_{i,j} \frac{\partial \mathbf{x}^{l}_{i,j}}{\partial \mathbf{W}^{l}_{m',n'}}, \tag{10}$$

Back-propagation for Shape Priors Cost Term

- where $x^l_{i,j}$ is the convolved input vector at layer l plus the bias represented:

$$\mathbf{x}^l_{i,j} = \sum_m \sum_n \mathbf{W}^l_{m,n} \mathbf{o}^{l-1}_{i+m,j+n} + \mathbf{b}^l, \tag{11}$$

  and the output vector at layer l given by $\mathbf{o}^l_{i,j} = max(\mathbf{x}^l_{i,j,}, 0)$.

- For $l = D$ and $\mathbf{x}^D = \hat{\mathbf{y}}$:

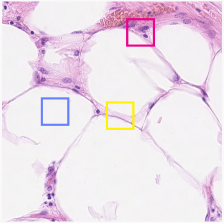
$$\delta^D_{i,j} = \frac{\partial P}{\partial \mathbf{x}^D_{i,j}} = -\sum_{i=1}^{n} g_p^{-1}(\mathbf{x}^D_{i,j} \odot \hat{\mathbf{x}}) * \text{rot}_{180^\circ} \left\{ S_{m,n} \right\}, \tag{12}$$

  where $g_p^{-1}(\cdot)$ is assign the weights to where it comes from - the “winning unit” because other units in the previous layer’s pooling blocks did not contribute to it hence all the other assigned values of zero. For the mathematical notations please refer to [25] and [26]..

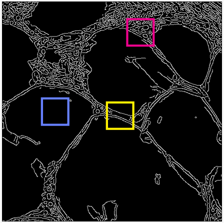
[25] V. Dumoulin *et al.*, arXiv 2016

[26] Y. LeCun *et al.*, Proc. of the IEEE, 1998

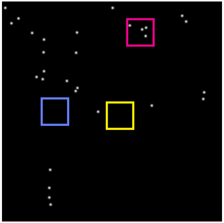
Preparation of Training Data



Raw input $(\mathbf{x})$



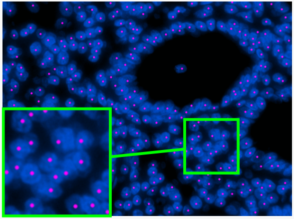
Raw edge $(\hat{\mathbf{x}})$



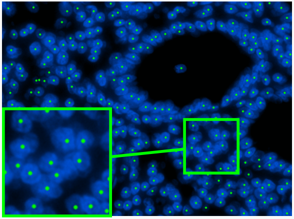
True labels $(\mathbf{y})$

✗ in void region	✗ no edges	✗ no center
✓ on tissue	✓ with edges	✗ no center
✓ on tissue	✓ with edges	✓ with centers

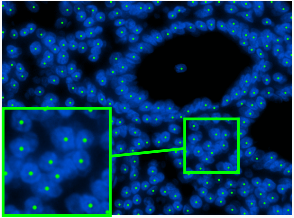
Example Results



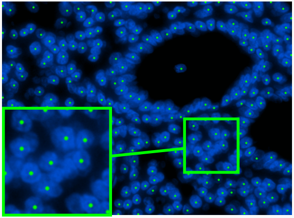
Groundtruth



Detection by SP-CNN; F1-score = 0.881

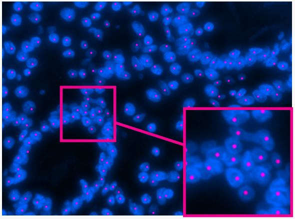


Detection by SC-CNN; F1-score = 0.838

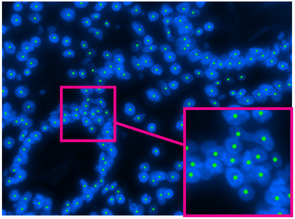


Detection by SR-CNN; F1-score = 0.827

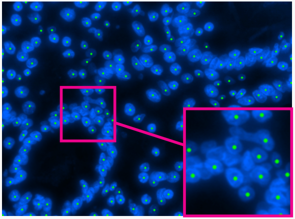
Example Results



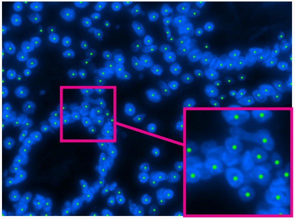
Groundtruth



SP-CNN



SC-CNN5



SR-CNN