2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)
April 4-7, 2018, Washington, D.C., USA

MICROSCOPY CELL SEGMENTATION VIA ADVERSARIAL NEURAL NETWORKS

Assaf Arbelle and Tammy Riklin Raviv†*

The Zlotowski Center for Neuroscience, Ben-Gurion University of the Negev, Israel

ABSTRACT

We present a novel method for cell segmentation in microscopy images which is inspired by the Generative Adversarial Neural Network (GAN) approach. Our framework is built on a pair of two competitive artificial neural networks, with a unique architecture, termed Rib Cage, which are trained simultaneously and together define a min-max game resulting in an accurate segmentation of a given image. Our approach has two main strengths, similar to the GAN, the method does not require a formulation of a loss function for the optimization process. This allows training on a limited amount of annotated data in a weakly supervised manner. Promising segmentation results on real fluorescent microscopy data are presented. The code is freely available at: https://github.com/arbellea/DeepCellSeg.git

1. INTRODUCTION

Live cell microscopy imaging is a key component in the biological research process. However, without the proper analysis tools, the raw images are a diamond in the rough. One must obtain the segmentation of the raw images defining the individual cells prior to calculation of the cells' properties. Manual segmentation is infeasible due to the large quantity of images and cells per image.

Automatic segmentation tools are available and roughly split into two groups, supervised and unsupervised methods. The methods vary and include: automatic gray level thresholding [1], the watershed algorithm [2] and Active Contours [3, 4]. Another approach is to support the segmentation algorithm with temporal information from tracking algorithms as was proposed by [5, 6]. All these methods assume some structure in the data that may not fit every case.

Supervised methods, on the other hand, do not assume any structure rather aim to learn it from the data. Classic machine learning methods generally require two independent steps, feature extraction and classification. In most cases the feature extraction is based either on prior knowledge of the image properties such as in [7] or general image properties such as smoothing filters, edge filters, etc. A widely used toolbox which takes a pixel classification approach is Ilastik [8], using a random forest classifier trained on predefined features extracted from a user's scribbles on the image.

Recent developments in the computer vision community have shown the strength Convolutional Neural Networks (CNNs) which surpass state of the art methods in object classification [9], semantic segmentation [10] and many other tasks. Recent attempts at cell segmentation using CNNs include [11, 12]. The common ground of all CNN methods is the need for an extensive training set alongside a predefined loss function such as the cross-entropy (CE).

In this work we present a novel approach for microscopy cell segmentation inspired by the GAN [13] and extension thereof [14, 15, 16, 17]. The GAN framework is based on two networks, a generator and a discriminator, trained simultaneously, with opposing objectives. This allows the discriminator to act as an abstract loss function in contrast to the common CE and L_1 losses. We propose a pair of adversarial networks, an **estimator** and a discriminator for the task of microscopy cell segmentation. Unlike the original GAN [13], we do not generate images from random noise vectors, rather estimate the underlaying variables of an image. The estimator learns to output some segmentation of the image while the discriminator learns to distinguish between expert manual segmentations and estimated segmentations given the associated image. The discriminator is trained to *minimize* a classification loss on two classes, manual and estimated, i.e. minimizing the similarity between the two. The estimator, on the other hand, is trained to *maximize* the discriminator's loss and effectively, maximize the similarity. In [18], semantic segmentation of natural images are generated for a set of predefined class. However, the main difference lays in our need to separate instances of a single class (cells) and not to separate different classes. The method also differs in choice of discriminator architecture and training method.

Our contribution is three-fold. We expand the concept of the GAN for the task of cell segmentation and in that reduce the dependency on a selection of loss function. We propose a novel architecture for the discriminator, referred to as the "Rib Cage" architecture (See section 2.4.2), which is adapted to the problem. The "Rib Cage" architecture includes several cross connections between the image and the segmentation,

*This study was partially supported by the The Negev scholarship at Ben-Gurion University); The Kreitman School of Advanced Graduate Studies)

†Israel Science Foundation (1638/16); The Israel Defense Forces (IDF) Medical Corps and Directorate of Defense Research & Development, Israeli Ministry of Defense (IMOD DDR&D); The Israeli ministry of science, thechnology and space (63551)

allowing the network to model complicated correlation between the two. Furthermore we show that accurate segmentations can be achieved with a low number of training examples therefore dramatically reducing the manual workload.

The rest of the paper is organized as follows. Section 2 defines the problem and elaborates on the proposed solution. Section 3 presents the results for both a common adversarial and non-adversarial loss compared to the proposed method, showing promising initial results. Section 4 summarizes and concludes the work thus far.

2. METHODS

2.1. Problem Formulation

Let Ω define the image domain and let the image $I : \Omega \to \mathbb{R}^+$ be an example generated by the random variable $\mathcal{I}$. Our objective is to partition the image into individual cells, where the main difficulty is separating adjacent cells. Let the segmentation image $\Gamma : \Omega \to \{0, 1, 2\}$ be a partitioning of Ω to three disjoint sets, background, foreground (cell nuclei) and cell contour, also generated by some random variable $\mathcal{S}$. The two random variables are statistically dependent with some unknown joint probability $P_{\mathcal{I},\mathcal{S}}$. The problem we address can be formulated as the most likely partitioning $\hat{\Gamma}$ from the data I given only a small number, N, of example pairs $\{I_n, \Gamma_n\}_{n=1}^{N}$. Had $P_{\mathcal{S}|\mathcal{I}}(\Gamma|I)$ been known, the optimal estimator would be the Maximum Likelihood (ML) estimator:

$$\hat{\Gamma}_{opt} = \arg\max_{\Gamma} P_{\mathcal{S}|\mathcal{I}}(\Gamma|I) \tag{1}$$

However, since $P_{\mathcal{S}|\mathcal{I}}(\Gamma|I)$ is unknown and $\hat{\Gamma}_{opt}$ cannot be calculated, we learn the near-optimal estimator of Γ using the manual segmentation, Γ_M, as our target.

2.2. Estimation Network

We propose an estimator $\hat{\Gamma} = \mathcal{E}\left(I, \hat{\theta_{\mathcal{E}}}\right)$ in the form of a CNN with parameters $\theta_{\mathcal{E}}$. We wish to train the estimator $\mathcal{E}$ such that the estimated $\hat{\Gamma}$ will be as close as possible to the optimal ML estimation $\hat{\Gamma}_{opt}$. This is achieved by optimizing for some loss function $L_{\mathcal{E}}$ (defined in section 2.3):

$$\hat{\theta}_{\mathcal{E}} = \arg\min_{\theta_{\mathcal{E}}} L_{\mathcal{E}}\left(\mathcal{E}(I, \theta_{\mathcal{E}}), \hat{\Gamma}_{opt}\right) \tag{2}$$

2.3. Adversarial Networks

Unlike the GAN, aiming to *generate examples* from an unknown distribution, we aim to *estimate the variables* of an unknown conditional distribution $P_{\mathcal{S}|\mathcal{I}}(\Gamma|I)$. Defining the loss $L_{\mathcal{E}}$ either in a supervised pixel-based way, e.g. L_2 norm, or in an unsupervised global method, by a cost functional that constrains partition into homogenous regions while minimizing the length of their boundaries, is usually not well defined. We define the loss $L_{\mathcal{E}}$ by pairing our estimator with a discriminator. Let $\mathcal{E}_{\theta_{\mathcal{E}}}$ and $\mathcal{D}_{\theta_{\mathcal{D}}}$ denote the estimator and discriminator respectively, both implemented as CNN with parameters $\theta_{\mathcal{E}}$ and $\theta_{\mathcal{D}}$ respectively. The estimator aims to find the best estimation $\hat{\Gamma}$ of the partitioning Γ given the image I. The discriminator on the other hand tries to distinguish between Γ_M and $\hat{\Gamma}$ given pairs of either (I, Γ_M) or $\left(I, \hat{\Gamma}\right)$ and outputs the probability that the input is manual rather than estimated denoted as $D\left(I, \hat{\Gamma}\right)$. As is in the GAN case, the objectives of the estimator and the discriminator are exactly opposing and so are the losses for training $\mathcal{E}_{\theta_{\mathcal{E}}}$ and $\mathcal{D}_{\theta_{\mathcal{D}}}$. We train $\mathcal{D}_{\theta_{\mathcal{D}}}$ to *maximize* the probability of assigning the correct label to both manual examples and examples estimated by $\mathcal{E}_{\theta_{\mathcal{E}}}$. We simultaneously train $\mathcal{E}_{\theta_{\mathcal{E}}}$ to *minimize* the same probability, essentially trying to make $\hat{\Gamma}$ and Γ_M as similar as possible:

$$L_{\mathcal{D}} = \mathbb{E}\left[\log\left(D\left(I, \Gamma_M\right)\right) + \log\left(1 - D\left(I, \mathcal{E}_{\theta_{\mathcal{E}}}(I)\right)\right)\right] \tag{3}$$

$$L_{\mathcal{E}} = \mathbb{E}\left[\log\left(D\left(I, \mathcal{E}_{\theta_{\mathcal{E}}}(I)\right)\right)\right] \tag{4}$$

In other words, $\mathcal{E}_{\theta_{\mathcal{E}}}$ and $\mathcal{D}_{\theta_{\mathcal{D}}}$ are players in a min-max game with the value function:

$$\min_{\mathcal{E}_{\theta_{\mathcal{E}}}} \max_{\mathcal{D}_{\theta_{\mathcal{D}}}} \mathbb{E}\left[\log\left(D\left(I, \Gamma_M\right)\right) + \log\left(1 - D\left(I, \mathcal{E}_{\theta_{\mathcal{E}}}(I)\right)\right)\right] \tag{5}$$

The equilibrium is achieved when $\hat{\Gamma}$ and Γ_M are similar such that the discriminator can not distinguish between the pairs $\left(I, \hat{\Gamma}\right)$ and (I, Γ_M).

2.4. Implementation Details

2.4.1. Estimator Network Architecture

The estimator $\mathcal{E}_{\theta_{\mathcal{E}}}$ net is designed as a five layer fully CNN, each layer is constructed of a convolution followed by batch normalization and leaky-ReLU activation. The output of the estimator is an image with the same size as the input image with three channels corresponding to the probability that a pixel belongs to the background, foreground or cell contour.

2.4.2. Discriminator Network Architecture

The discriminator $\mathcal{D}_{\theta_{\mathcal{D}}}$ is designed with a more complex structure. The discriminators task is to distinguish manual and estimated segmentation images given a specific gray level (GL) image. The question arrises of how to design the discriminator architecture which can get both the GL and segmentation images as input. A basic design s that of a classification CNN where both images are concatenated in the channel axis, as done in [17]. However, we believe that this approach is not optimal for our needs since, for this task, the discriminator should be able match high level

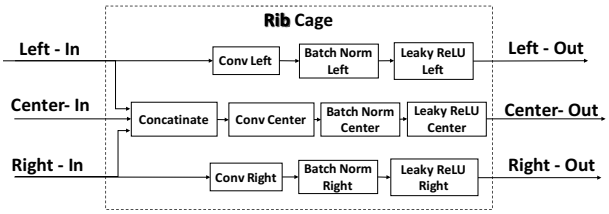

Fig. 1: The design of the basic building block for the discriminator. Each block has three inputs and three outputs.

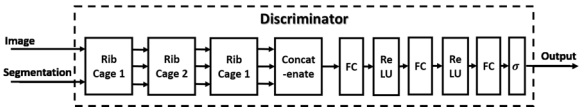

Fig. 2: The design of the Discriminator $\mathcal{D}_{\theta_\mathcal{D}}$. Three "Rib Cage" blocks (see Figure 1) are followed by two FC layers with ReLU activations and a last FC layer with a sigmoid activation, σ. The Center-In chanel of the first Rib Cage block is ommited.

features from the GL and segmentation images. Yet these features may have very different appearances. For example, an edge of a cell in the GL image appear as a transition from white to black while the same edge in the segmentation image appears as a thin blue line. This difference requires the network to learn individual filters for each semantic region. Then, finding correlations between the two is a more feasible task. For these reasons we designed a specific architecture, referred to as a "Rib Cage" architecture, which has three channels. The first and second channels get inputs from the GL channel and segmentation channel respectively, each channel calculates feature maps using a convolutional layer, we refer to these channels as the "Ribs". The third channel, referred to as the "Spine", gets a concatenation of inputs from both the GL and segmentation channels and matches feature maps (i.e correlations). See Figure 1 for an illustration of the "Rib Cage" block. The discriminator is designed as three consecutive "Rib Cage" blocks followed by two fully-connected (FC) layers with leaky-ReLU activations and a final FC layer with one output and a sigmoid activation for classification. Figure 2 illustrates the discriminator design. The architecture parameters for the convolution layers are describes as $C\left(kernel\ size, \#\ filters\right)$ and FC layers as $F\left(\#\ filters\right)$. The parameters for the estimator: $C\left(9,16\right)$, $C\left(7,32\right)$, $C\left(5,64\right)$, $C\left(4,64\right)$, $C\left(1,3\right)$. The discriminator spine used half the number of filters as the ribs: $C\left(9,8\right)$, $C\left(5,32\right)$, $C\left(3,64\right)$, $C\left(4,64\right)$, $F\left(64\right)$, $F\left(64\right)$, $F\left(1\right)$.

2.4.3. Data

We trained the networks on the H1299 data set [19] consisting of 72 frames of size 512×640 pixels. Each frame captures approximately 50 cells. Manual annotation of 15 randomly

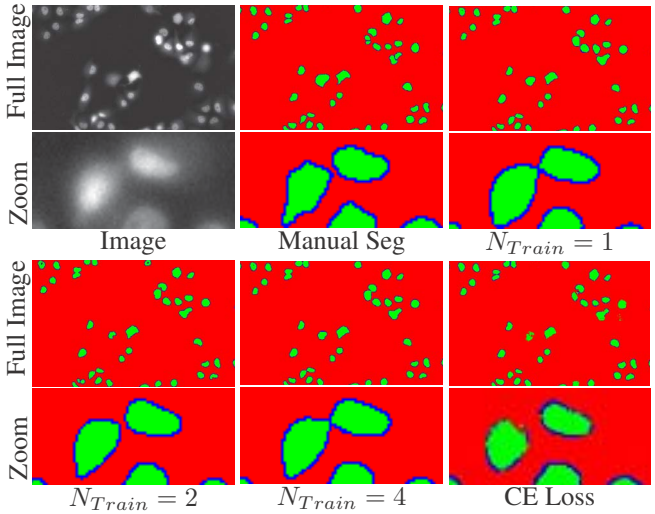

Fig. 3: Segmentation example of a validation image given a different number of training examples. The odd and even rows show the full image and a zoomed area respectively. Notice that in all cases the cells in the second row were correctly separated even though they are very close together. The bottom right shows the result when training with the CE loss.

selected frames was done by an expert. The annotated set was split into a training set and validation set. The training set was subsampled to $N_{Train} \in [1, 2, 4, 11]$ examples for training which were augmented using randomly cropped areas of size 64×64 pixels along with random flip and random rotation. The images were annotated using three labels for the background (red), cell nucleus (green) and nucleus contour (blue) encoded as RGB images.

3. EXPERIMENTS AND RESULTS

We conducted four experiments, training the networks with different values for N_{Train}. All other parameters were set identically. We evaluated the segmentation using the as described in the caption of Table 1. We compared the adversarial training regime to the common CE loss, training only the estimator. We furthermore evaluate our choice of RibCage discriminator versus a classification architecture (VGG16 [20]). We also compared our results to state of the art segmentation tool, Ilastik [8]. The manual annotation were done by an expert. The quantitative results of the individual cell segmentation are detailed in Table 1. Note that the amount of images in the training data had little effect on the results. Figure 3 shows an example of a segmented frame. It is clear that the networks learned a few distinct properties of the segmentation. First, each cell is encircled by a thin blue line. Second, the shape of the contour follows the true shape of the cell. Some drawbacks are still seen where two cells completely touch and the boundary is difficult to determine.

	ADV-1	ADV-2	ADV-4	ADV-11	CE - 11	Class Disc	Ilastik
Prec	**89.9%**	85.4%	86.8%	85.8%	83.6%	78.7%	81.2%
Rec	**82%**	**87.2%**	**86.8%**	**86.5%**	86.4%	81.14%	80.2%
F	**85.8%**	**86.3%**	**86.8%**	**86.1%**	84.8%	79.9%	80.7%
J	**80.6%**	**75.8%**	**77.4%**	**74.6%**	72.1%	60.2%	68.4%

Table 1: **Quantitative Results:** Each column represents an experiment with a different number of training examples, ADV-N_{Train}. CE Loss-11 and ClassDisc are experiments using the same estimator network trained with the pixel-based CE loss and a simple classification discriminator respectively. The last column is the comparison to the state of the art tool, Ilastik [8]. The rows are the results for individual cell segmentation. As is explained in [5] True positives (TP) are cells with Jaccard measure greater than 0.5. False positives (FP) are automatic segmentation not appearing in the manual segmentation and false negatives (FN) is the opposite. The measures are defines as $Prec = \frac{TP}{TP+FP}$, $Rec = \frac{TP}{TP+FN}$, $F - Measure = 2\frac{Prec*Rec}{Prec+Rec}$. J indicates the mean Jaccard measure for individual cells.

4. SUMMARY

In this work we propose a new concept for microscopy cell segmentation using CNN with adversarial loss. The contribution of such an approach is two-fold. First, the loss function is automatically defined as it is learned along side the estimator, making this a simple to use algorithm with no tuning necessary. Second, we show that this method is robust to low number of training examples surpassing.

The quantitative results, as well as the visual results, show clearly that both the estimator and our unique "Rib Cage" discriminator learn both global and local properties of the segmentation, i.e the shape of the cell and the contour surrounding the cell, and the fitting of segmentation edges to cell edges. These properties could not be learned using only a pixel-bases CE loss as is commonly done.

5. REFERENCES

[1] T. Kanade, et al., "Cell image analysis: Algorithms, system and applications," in *WACV*. IEEE, 2011, pp. 374–381.

[2] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," *IEEE PAMI*, vol. 13, no. 6, pp. 583–598, 1991.

[3] P. Bamford and B. Lovell, "Unsupervised cell nucleus segmentation with active contours," *Signal Processing*, vol. 71, no. 2, pp. 203–213, 1998.

[4] E. Meijering, et al., "Methods for cell and particle tracking," *Methods Enzymol*, vol. 504, no. 9, pp. 183–200, 2012.

[5] M. Schiegg, et al., "Graphical model for joint segmentation and tracking of multiple dividing cells," *Bioinformatics*, vol. 31, no. 6, pp. 948–956, 2014.

[6] A. Arbelle, et al., "Analysis of high throughput microscopy videos: Catching up with cell dynamics," in *MICCAI 2015*, pp. 218–225. Springer, 2015.

[7] H. Su, et al., "Cell segmentation in phase contrast microscopy images via semi-supervised classification over optics-related features," *MEDIA*, vol. 17, no. 7, pp. 746–765, 2013.

[8] C. Sommer, et al., "Ilastik: Interactive learning and segmentation toolkit," in *IEEE ISBI 2011*, March 2011, pp. 230–233.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012, pp. 1097–1105.

[10] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE CVPR*, 2015, pp. 3431–3440.

[11] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *arXiv preprint arXiv:1505.04597*, 2015.

[12] O. Z. Kraus, J. L. Ba, and B. J. Frey, "Classifying and segmenting microscopy images with deep multiple instance learning," *Bioinformatics*, vol. 32, no. 12, pp. i52–i59, 2016.

[13] I. Goodfellow, et al., "Generative adversarial nets," in *NIPS*, 2014, pp. 2672–2680.

[14] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[15] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[16] D. J. Im, et al., "Generating images with recurrent adversarial networks," *arXiv preprint arXiv:1602.05110*, 2016.

[17] P. Isola, et al., "Image-to-image translation with conditional adversarial networks," *arXiv preprint arXiv:1611.07004*, 2016.

[18] P. Luc, et al., "Semantic segmentation using adversarial networks," *arXiv preprint arXiv:1611.08408*, 2016.

[19] A. A. Cohen, et al., "Dynamic proteomics of individual cancer cells in response to a drug," *science*, vol. 322, no. 5907, pp. 1511–1516, 2008.

[20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.