ISOO$_{DL}$: INSTANCE SEGMENTATION OF OVERLAPPING BIOLOGICAL OBJECTS USING DEEP LEARNING

Anton Böhm[1] *Annekathrin Ücker*[2,4] *Tim Jäger*[2,4,5] *Olaf Ronneberger*[1,6] *Thorsten Falk*[1,3]

[1] Department of Computer Science, [2] Center for Complementary Medicine, [3] BIOSS Centre for Biological Signalling Studies, University of Freiburg, Germany

[4] Institute of Integrative Medicine, University of Witten/Herdecke, [5] Institute of Complementary Medicine, University of Bern, [6] DeepMind, London, UK

ABSTRACT

Image segmentation is an important first step for the quantitative analysis of biomedical images. We present a method to simultaneously segment and classify translucent overlapping objects in 2D images. For this we propose an approach using a fully-convolutional neural network simultaneously solving two tasks: object detection and instance segmentation. Object detection predicts reference points, object class labels and sizes. To solve the problem of multiple labels per location, we lift our label-space from 2D to 3D, resulting in a non-overlapping representation of the instance masks. To our knowledge it is the first method that handles overlapping biological objects using deep learning making it easily applicable to a large variety of challenging datasets.

Index Terms— instance segmentation, object detection, overlapping objects, deep learning

1. INTRODUCTION

Detection, classification and segmentation are problems that frequently occur in biomedical image analysis. One example is the quantification of shape variations of different types of cells which requires proper cell instance segmentation and classification. In 2D projections, as e.g. in brightfield microscopy, clustering cells appear to overlap which makes conventional image segmentation impossible. In this work we provide a general approach that can handle these problems. It is based on Convolutional Neural Networks (CNN) that have demonstrated drastic performance improvements on pixelwise semantic segmentation [1, 2] in the last years. However, semantic segmentation does not distinguish object instances and only few of the existing approaches provide workarounds for instance aware semantic segmentation. An even smaller subset of them can be generalized to handle overlapping objects. We solve all mentioned problems with a single Fully Convolutional Network.

We thank the German Federal Ministry for Economic Affairs and Energy (FKz. ZF4184101CR5) and the DFG (EXC 294) for funding our research.

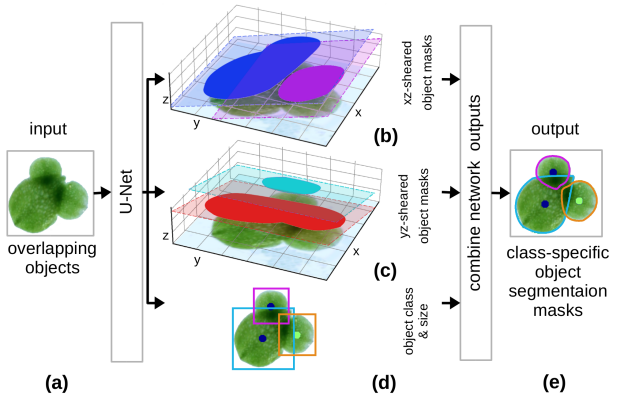

Fig. 1. Proposed segmentation pipeline. **(a)** Raw input image with overlapping translucent objects. **(b-c)** Predicted segmentations. The overlapping objects are encoded as sheared 2D masks in a 3D volumetric image. Shearing in xz- and in yz-direction is required to resolve ambiguities. **(d)** Predicted bounding boxes and object centers. **(e)** Resulting overlapping class-specific segmentation masks, displayed as outlines.

Our method is able to separate overlapping translucent objects providing complete segmentation masks for each instance. For this, we lift our label-space from 2D to 3D and perform the object segmentation in 3D. To obtain pixel-accurate class-specific object masks in 2D, we combine the 3D representation with object reference points and bounding boxes which are predicted by the network as well. The overview of our method is shown in Fig. 1.

Our main contributions are: 1. We present a method that produces complete class-labeled instance masks of overlapping translucent objects. 2. We introduce an object detection framework that does not generate proposals and refines them, but predicts reference points containing information about the location, size and class of an object.

2. RELATED WORK

In the last years many approaches appeared that deal with instance segmentation. Some of them inspired our work: The

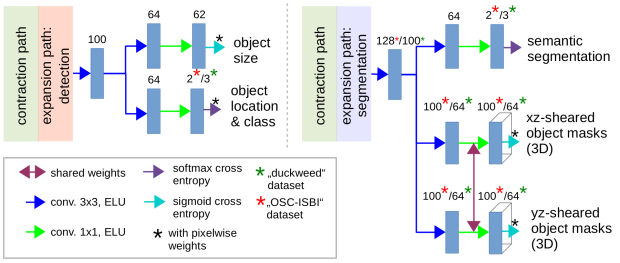

Fig. 2. Two-headed U-Net [2] architecture: Two expansion paths share feature maps (FM) from one contraction path. The channel number is shown on top of each FM. Parameters without dataset indication are used for both datasets. The spatial resolution of all FMs after the last 3×3 convolution is 452×452, only valid parts of convolutions are propagated.

authors of [2] separate touching objects by lines of one pixel width which are assigned to the background class label. The network is forced to learn this local "misclassification" via high pixelwise loss weights. With decreasing object distance the weights between two objects smoothly increase. The strategy leads to separated object instances but can not deal with overlapping objects. We use the same weighting scheme to add gaps between touching reference points which we present to the network as disks with fixed diameter of 9 pixels.

In [3] and [4] a grid with distinct class labels per grid cell is put on each object. The labels are ordered so that two touching objects always have different labels on touching borders. Similar to our work the authors transform the object masks to a higher dimension making them separable. Our method can be seen as a generalization of the mentioned methods, making it possible to also distinguish overlapping objects and therefore providing a more flexible approach.

In this work we show the suitability of our method for translucent objects, but, theoretically, it can be applied to images containing occluded non-transparent objects as well. For this, we suggest to generate the training data synthetically, following the strategies described in [5, 6].

In [7] the authors generate object bounding box proposals in the first stage which are then classified and regressed in a second stage. [8] extends the method, additionally predicting the binary segmentation mask per proposal.

3. METHOD

In contrast to [7, 8], we do not generate proposals and successively refine them. Instead we directly let a neural network predict class-labeled object reference points and corresponding segmentation masks in a higher dimensional space to disambiguate image regions occupied by multiple objects.

Our network architecture is based on the "U-Net" [2] with four resolution levels. The contraction path of the U-Net alternates between convolutions and pooling steps which reduce the spatial dimensions of the input image from 636×636 to 32×32 pixels at the lowest resolution and encode the relevant image information into 1024-dimensional feature vectors. We extend the U-Net to simultaneously solve the detection and segmentation tasks using two independent expansion paths (see Fig. 2). Additionally, we replace ReLUs by ELUs [9] and substitute the two last convolution layers of the original U-Net by new modules, to produce the proposed detections as 2D heat maps, and segmentation masks as 3D sigmoidal activations. In section 3.3 we show, how to combine all network outputs to get the final class-labeled object segmentation masks.

We represent object $O_k := (m_k, \mathbf{c}_k, \mathbf{b}_k, \mathbf{p}_k, y_k)$ as tuple, where $m_k : \Omega \to \{0, 1\}$ is its binary mask denoted as binary function of the image domain $\Omega \subset \mathbb{N}^2$, $\mathbf{c}_k \in \mathbb{R}^2$ is its bounding box center, $\mathbf{b}_k = (w_k, h_k)^\top \in \mathbb{R}^2$ are bounding box width and height, $\mathbf{p}_k \in \mathbb{R}^2$ is a user-defined object reference point in close proximity to $\mathbf{c}_k$ and $y_k \in \{1, \dots, C\}$ is its class label. We denote the set of all objects as $\mathcal{O} = \{O_1, \dots, O_K\}$. A hat above a letter indicates predicted variables, primed variables indicate 3D representations. $\mathbf{x} = (x, y)^\top \in \Omega$ are 2D image coordinates and $\mathbf{x}' = (\mathbf{x}^\top, z)^\top \in \Omega \times \mathbb{Z}$ are corresponding coordinates in 3D space.

3.1. Object Detection

An object reference point can be any point, approximately located in the middle of an object that uniquely describes it. We chose e.g. the position of the nucleus for the OSC-ISBI dataset and object bounding box centers for the duckweed dataset. We assign the object class label to the corresponding reference point.

At the spatial positions of the reference points we additionally predict the widths and heights of the object bounding boxes. The exact bounding box shape is not crucial for the approach, therefore, we roughly discretize the space of possible bounding box widths and heights according to

$$\tilde{w}_k = \left\lceil \frac{w_k}{D_{\max}} \cdot R \right\rceil \quad \text{and} \quad \tilde{h}_k = \left\lceil \frac{h_k}{D_{\max}} \cdot R \right\rceil,$$

where $D_{\max} = \max_j \sqrt{w_j^2 + h_j^2}$ is the size of the largest object in the training set and R is the number of discretization steps. We set $R = 31$ for all experiments. Discretization has the advantage that the network only performs classification tasks alleviating the problem of weighting regression losses to match classification losses.

The detection path produces two score maps: $\hat{s}_{\text{det}} : \Omega \times \{0, \dots, C\} \to [0, 1]$ estimating the probability for every pixel of belonging to class $c \in \{0, \dots C\}$ ($c = 0$: background, $c > 0$: object classes) and $\hat{s}_{\text{bbox}} : \Omega \times \{0, \dots, R\}^2 \to [0, 1]$ for discrete bounding box width and height (see Fig 2). During training we use weighted softmax cross entropy for the

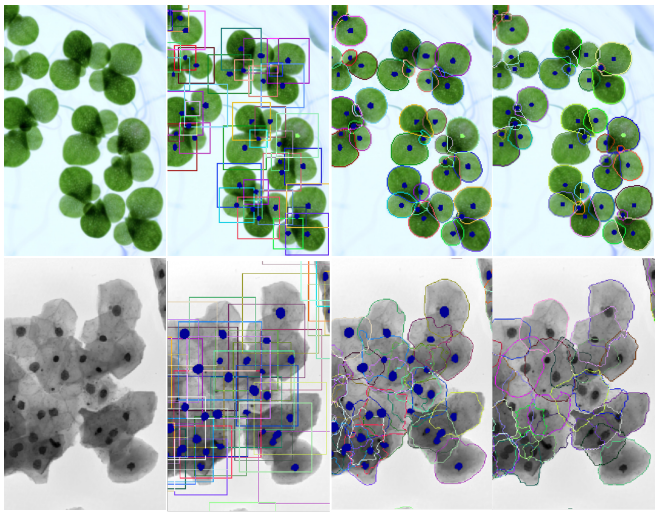

image detection segmentation ground truth

Fig. 3. Qualitative results on the "duckweed" (first row) and "OSC-ISBI" (second row) datasets.

object class and sigmoid cross entropy for the object bounding box sub-tasks. To make the detection more robust, we reduce the loss weight of pixels in the direct neighborhood of isolated object reference points on the loss. However, in areas of clustered reference points the influence is increased to properly separate them. For this, pixelwise loss weights are computed and applied as described in [2].

3.2. Object Segmentation

The object masks m_k can not be directly used for CNN training, because the 2D image does not provide any information about the depth of the objects in 3D space. However, a suitable transformation lifting the dimension of the label-space from 2D to 3D allows separation of the objects.

First, we lift the domain of the mask functions m_k to 3D using

$$m'_k(\mathbf{x}') = \begin{cases} m_k(\mathbf{x}) & z = 0 \\ 0 & \text{otherwise} \end{cases}.$$

Then we shear the coordinates of object mask m'_k in xz-direction so that its transformed bounding box center $\mathbf{c}'_k = (\mathbf{c}_k^\top, 0)^\top$ stays at $z = 0$ (see Fig. 1) and describe all masks by a single function $m'^x : \Omega \times \mathbb{Z} \to \{0, 1\}$ where

$$m'^x(\mathbf{x}') := \max_{k \in \{1,\dots,K\}} \left(m'_k \circ (T_k^x)^{-1}\right)(\mathbf{x}').$$

$T_k^x : \mathbb{R}^3 \to \mathbb{R}^3$ is the object specific shear transformation. Superscripts $.^x$ and $.^y$ indicate shearing in xz- and yz-direction in the rest of the paper. As shown in Fig. 1 shearing in xz-direction resolves the problem of overlapping objects in y-direction. However, overlaps in x-direction stay unresolved. When also shearing in yz- direction and properly combining both representations we can separate all objects (see Sec. 3.3). Separation fails only in the pathological case of identical bounding box centers.

The shearing transformation in xz-direction is composed as follows $T_k^x = (T_{\mathbf{c}_k})^{-1} \circ T_s^x \circ T_{\mathbf{c}_k}$. The components of the transformation are: Translation $T_{\mathbf{c}_k}$ which shifts the object bounding box center $\mathbf{c}_k$ to the origin; shearing T_s^x with fixed angle γ which is chosen individually for every dataset depending on the maximal object size. The transformation T_k^y is constructed equivalently with yz-shearing T_s^y.

We train the network on m'^x and m'^y using sigmoid cross entropy as objective function. Between the sheared object masks we compute loss weights similar to [2], and balance the influence of foreground and background class by setting the weights higher for all positions $\{\mathbf{x}' \,|\, \exists k : \; m_k(\mathbf{x}) = 1\}$ that project on any object mask.

Additionally, we let the network perform a semantic segmentation. It puts the network focus on foreground regions which in our case has a stabilizing effect on the training process.

3.3. Post-processing: Combining the Network Outputs

A very important property of our lifting transformation is the existence of a corresponding back-transformation $\mathcal{T}^{-1} : (\hat{m}'^x, \hat{m}'^y, \hat{s}_{\text{det}}, \hat{s}_{\text{bbox}}) \mapsto \hat{\mathcal{O}}$.

We combine both representations described in Sec. 3.2 via the reference points. To retrieve the reference points, the output scores $\hat{s}_{\text{det}}$ of the detection path are first turned into a hard pixelwise classification $\hat{Y}$ using $\forall \mathbf{x} \in \Omega : \hat{Y}(\mathbf{x}) = \text{argmax}_{c \in \{0,\dots,C\}} \hat{s}_{\text{det}}(\mathbf{x}, c)$. The object locations $\hat{\mathcal{P}} := \{\hat{p}_1, \dots, \hat{p}_{\hat{K}}\}$ are the centers of mass of the connected components of $\hat{Y}$ for all foreground classes. As predicted bounding box extents $\hat{\mathbf{b}}_k$ and label $\hat{y}_k$ of object $\hat{O}_k$ we pick the class occurring most frequently in the area of the corresponding connected component. The number of detected objects $\hat{K}$ is the number of connected components for all classes.

The pseudo-probabilities $\hat{m}'^x, \hat{m}'^y : \Omega \times \mathbb{Z} \to [0, 1]$ output by the 3D network must be projected to 2D space to get the final segmentation. For each object $\hat{O}_k$ we first undo the shearing of $\hat{m}'^x$ and $\hat{m}'^y$ using the object-specific T_k^x and T_k^y, respectively, to map the probabilities for that object to the $z = 0$ plane using

$$p_k^x(\mathbf{x}') = (\hat{m}'^x \circ T_k^x)(\mathbf{x}') \quad \text{and} \quad p_k^y(\mathbf{x}') = (\hat{m}'^y \circ T_k^y)(\mathbf{x}').$$

To obtain 2D probability maps for each object, we first compute the joint probability $p_k^x(\mathbf{x}') \cdot p_k^y(\mathbf{x}')$ and then project along the z-axis weighted by a Gaussian $\mathcal{N}_{0,\sigma}$:

$$p_k(\mathbf{x}) = \sum_z p_k^x(\mathbf{x}') \cdot p_k^y(\mathbf{x}') \cdot \mathcal{N}_{0,\sigma}(z).$$

The Gaussian with standard deviation σ models localization inaccuracies. We only consider locations within the area

of the predicted bounding box $\hat{\mathbf{b}}_k$. Finally, we apply Otsu thresholding [10] per p_k to get the binary masks to which we assign the corresponding class labels yielding the final segmentation $\hat{m}_k$.

3.4. Training

We train the network in the caffe framework [11] until convergence (250K iterations) using the ADAM solver [12] with the first and second momentum set to 0.9 and 0.999, respectively. We start with a base learning rate of 0.00001 which is reduced by a factor of 0.2 every 50K iterations. We regularize the training process with weight decay, setting the parameter to 0.00001. Since the test images are bigger than the network output, we process input images in tiles. The network is trained on augmented data. Data augmentation includes rotation, flipping, elastic deformation as described in [2] and random strictly increasing intensity transformation.

4. RESULTS

4.1. Evaluation scores

We quantitatively evaluate the quality of the predicted object masks as in [13], computing dice coefficient (DC), pixel-based true positive (TPp) and false positive (FPp) rates on the "good" segmentations with DC $> t$ and object-based false negative (FNo), measuring the rate of "bad" segmentations (DC $\leq t$) where $t = 0.7$ is the DC threshold.

4.2. Datasets

Duckweed Objects of interest are the individual leaves of the duckweed (*Lemna gibba* L.) plant rosettes. Every leaf can be diseased (showing chloroses) or healthy. Chloroses are areas with less chlorophyll in the leaves ranging in size from only few pixels to the whole leaf area. We predict these areas as second class in the semantic segmentation sub-task. Another challenge of the dataset is object size variation. As reference points we use the bounding box centers. We present these points to the network as one image containing disks with 9 pixel diameter around the bounding box centers. We use 64 discrete depth levels for m'^x and m'^y which we encode as 64 output channels, yielding shearing angle $\gamma = 29°$.

The dataset contains 33 images. We train the network on 22 images and test it on the remaining 11. Quantitative results are provided in Tab. 1. The class "diseased" contains many segments with areas of only few pixels. For these areas little spatial misalignment of a predicted segmentation mask causes a "mismatch", even though the detection is present. This leads to the observed high false negative rate for that class. We reduce the dice threshold to $t = 0.5$ to support this statement (see "diseased@0.5" in Tab. 1). The qualitative results are depicted in Fig. 3.

	DC	FNo	TPp	FPp
healthy	.930±0.055	.128±.068	.939±.066	.000±.000
diseased	.904±0.099	.500±.336	.934±.069	.000±.000
diseased@0.5	.816±0.152	.250±.194	.858±.137	.000±.000
all	.929±0.057	.129±.067	.939±.066	.000±.000

Table 1. Results on the "duckweed" dataset

	DC	FNo	TPp	FPp
Phoulady *et al.*[14]	.831±.079	.408±.163	.927±.098	.003±.002
Ramalho *et al.*[15]	.856±.078	.501±.180	.899±.113	.002±.001
Lee *et al.* [16]	.879±.087	.434±.168	.877±.123	.001±.001
ours	.863±.074	.370±.141	.895±.107	.001±.001
ours$_{\text{plus}}$	.855±.072	.334±.141	.912±.099	.002±.001

Table 2. Results on the "OSC-ISBI" dataset

OSC-ISBI The dataset OSC-ISBI is part of "The Second Overlapping Cervical Cytology Image Segmentation Challenge - ISBI 2015" [17] and contains 8 images for training and 9 for testing. Although the dataset only contains objects of one class, their density is very high, making instance segmentation very difficult even for human experts. Therefore annotations with two confidence-levels are provided, high-confident for obvious and low-confident for ambiguous annotations. Using the bounding box centers as reference points results in a better segmentation, but, compared to other methods, worse detection accuracy. So we decide to use the cell nuclei as reference points. They are also provided by the challenge as binary segmentation masks and much easier to detect than the more abstract bounding box centers. Our predicted projection planes are therefore slightly offset from the ones defined on the bounding box centers. To make our approach more robust to this effect, we increase the standard deviation in the point-to-plane matching to $\sigma = 3$ (cf. $\sigma = 2$ for duckweed). We reduce the image resolution by a factor of two and increase the depth (=number of channels) of m'^x and m'^y to 100 yielding shearing angle $\gamma = 39°$. We could further improve the detection performance of our method, by adding all synthetically generated data from the first challenge (denoted in Tab. 2 as "**ours**$_{\text{plus}}$").

The network is pretrained on high-confident annotations only and finetuned on all annotations after 100K iterations. Qualitative results are depicted in Fig. 3.

5. CONCLUSION

We could show that our deep learning approach is able to segment biological images containing translucent overlapping object instances with high density. The regular shape and comparably high contrast of the duckweed dataset allows to successfully segment up to three overlapping leaves at the same location with very high precision. Segmentation of cells in the OSC-ISBI dataset is a challenge even for human experts. We found that the network benefits from including noisy low confidence annotations and on-par with the state-of-the-art in cervical cell segmentation.

6. REFERENCES

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

[2] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 234–241.

[3] Yi Li, Haozhi Qi, Jifeng Dai, Xiangyang Ji, and Yichen Wei, "Fully convolutional instance-aware semantic segmentation," *arXiv preprint arXiv:1611.07709*, 2016.

[4] Jonas Uhrig, Marius Cordts, Uwe Franke, and Thomas Brox, "Pixel-level encoding and depth layering for instance-level semantic labeling," in *German Conference on Pattern Recognition*. Springer, 2016, pp. 14–25.

[5] Ke Li and Jitendra Malik, "Amodal instance segmentation," in *European Conference on Computer Vision*. Springer, 2016, pp. 677–693.

[6] Victor Yurchenko and Victor S. Lempitsky, "Parsing images of overlapping organisms with deep singling-out networks," *CoRR*, vol. abs/1612.06017, 2016.

[7] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[8] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick, "Mask r-cnn," *arXiv preprint arXiv:1703.06870*, 2017.

[9] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *arXiv preprint arXiv:1511.07289*, 2015.

[10] Nobuyuki Otsu, "A Threshold Selection Method from Gray-level Histograms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[11] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 675–678.

[12] Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[13] Zhi Lu, Gustavo Carneiro, Andrew P Bradley, Daniela Ushizima, Masoud S Nosrati, Andrea GC Bianchi, Claudia M Carneiro, and Ghassan Hamarneh, "Evaluation of three algorithms for the segmentation of overlapping cervical cells," *IEEE journal of biomedical and health informatics*, vol. 21, no. 2, pp. 441–450, 2017.

[14] Hady Ahmady Phoulady, Dmitry B Goldgof, Lawrence O Hall, and Peter R Mouton, "An approach for overlapping cell segmentation in multi-layer cervical cell volumes," *The Second Overlapping Cervical Cytology Image Segmentation Challenge-IEEE ISBI*, 2015.

[15] Geraldo LB Ramalho, Daniel S Ferreira, Andrea GC Bianchi, Claudia M Carneiro, Fátima NS Medeiros, and Daniela M Ushizima, "Cell reconstruction under voronoi and enclosing ellipses from 3d microscopy," in *IEEE International Symposium on Biomedical Imaging (ISBI)*, 2015.

[16] Hansang Lee and Junmo Kim, "Segmentation of overlapping cervical cells in microscopic images with superpixel partitioning and cell-wise contour refinement," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 63–69.

[17] Zhi Lu, Gustavo Carneiro, and Andrew P Bradley, "An improved joint optimization of multiple level set functions for the segmentation of overlapping cervical cells," *IEEE Transactions on Image Processing*, vol. 24, no. 4, pp. 1261–1272, 2015.