Learning Cross-Representation Affinity Consistency for Sparsely Supervised Biomedical Instance Segmentation Supplementary Material

Xiaoyu Liu[1,*] Wei Huang[1] Zhiwei Xiong[1,2,†]
Shenglong Zhou[1] Yueyi Zhang[1,2] Xuejin Chen[1,2] Zheng-Jun Zha[1] Feng Wu[1,2]
[1]University of Science and Technology of China
[2]Institute of Artificial Intelligence, Hefei Comprehensive National Science Center

1. Training Details of Confidence Pixel Selection Network (CPSN)

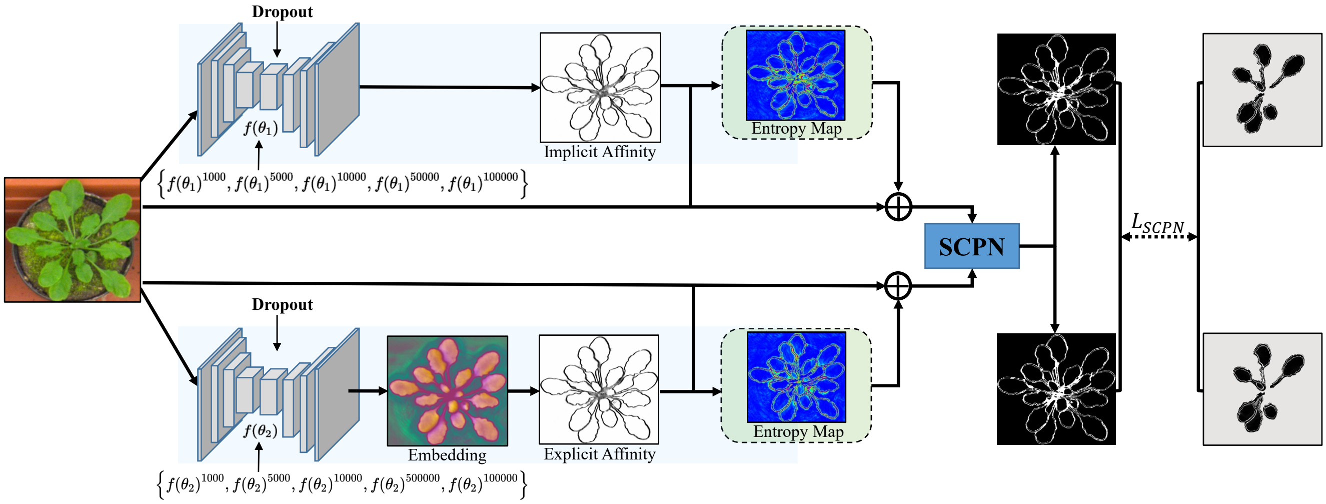
Dropout
$f(\theta_1)$
$\left\{f(\theta_1)^{1000}, f(\theta_1)^{5000}, f(\theta_1)^{10000}, f(\theta_1)^{50000}, f(\theta_1)^{100000}\right\}$
Implicit Affinity
Entropy Map
SCPN
L_{SCPN}
Dropout
$f(\theta_2)$
$\left\{f(\theta_2)^{1000}, f(\theta_2)^{5000}, f(\theta_2)^{10000}, f(\theta_2)^{500000}, f(\theta_2)^{100000}\right\}$
Embedding
Explicit Affinity
Entropy Map

Figure 1. Diagram of the training process of the CPSN.

The training process of the CPSN is shown in Fig.1. We sample five segmentation models trained for different training iterations of the network $f(\theta_1)$ and $f(\theta_2)$ respectively and add a certain rate of dropout to these models. Considering that the two networks $f(\theta_1)$ and $f(\theta_2)$ are trained for 200K iterations and that the prediction errors mainly appear in the first half of the whole training stage, we empirically sample models at these training iterations to ensure the diversity of simulated segmentation errors, which are described by these two model sets $F(\theta_1)$ and $F(\theta_2)$ as follows:

$$\begin{aligned} F(\theta_1) &= \left\{f(\theta_1)^{1000}, f(\theta_1)^{5000}, f(\theta_1)^{10000}, f(\theta_1)^{50000}, f(\theta_1)^{100000}\right\} \\ F(\theta_2) &= \left\{f(\theta_2)^{1000}, f(\theta_2)^{5000}, f(\theta_2)^{10000}, f(\theta_2)^{50000}, f(\theta_2)^{100000}\right\} \end{aligned} \tag{1}$$

where the superscript of each model represents the number of training iterations it was trained.

[*] This work was done during Xiaoyu Liu's internship at Institute of Artificial Intelligence, Hefei Comprehensive National Science Center.
[†] Corresponding author: zwxiong@ustc.edu.cn.

We then use predictions from these models to simulate various segmentation errors. Using this simulation data, we train a robust CPSN to locate segmentation errors and generate highly confident pixels. To further improve the CPSN robustness, we add dropout with a certain rate to the last two layers of these models so that their predictions contain more noise.

Given the input image I, affinity map A and affinity entropy map H, the predicted binary error map B with the same size as A, which is denoted by $B = \text{CPSN}\,(I \oplus A \oplus H)$, where $\oplus$ is the concatenation operation. We then adopt the binary cross-entropy loss for SCPN, L_{SCPN} is described as

$$\mathcal{L}_{\text{CPSN}} = \frac{1}{|R_a^L|} \sum_{i \in R_a^L} \sum_{n=1}^{N} L_{ce}\left(b_{n,i}, m_{n,i}\right) \tag{2}$$

where R_a^L is the annotated affinity region, $m_{n,i}$ is 1 if the prediction of pixel affinity $a_{n,i}$ is not confident and 0 otherwise.

The SCPN network has the same network as the segmentation networks, *i.e.*, 3D U-Net [5] and 2D Residual U-Net [1] for the 3D volumetric datasets (AC3, CREMI-C) and 2D datasets (CVPPP, BBBC039V1), respectively. We train the SCPN networks using Adam optimizer [4] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$, a learning rate of $1e-4$, and a batch size of 2 on one NVIDIA TitanXP GPU for 200K iterations. In each iteration, we randomly sample a model from $F(\theta_1)$ and $F(\theta_2)$ to predict affinity map with errors.

2. Size Distribution of Annotated Instances

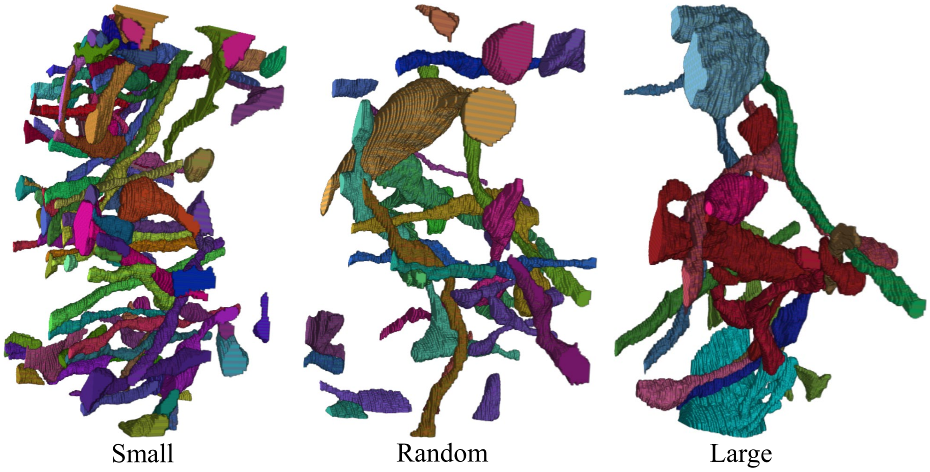

Figure 2. 3D annotated masks with different size distributions on the AC3 dataset. The instances on the left and right have smaller and larger sizes than the median size of all instances on the dataset. The instances in the middle are randomly annotated for experiments in the main text.

In the sparse instance-level supervision setting of the main text, we randomly annotate the complete instances without considering the size of each instance. In this section, we validate the impact of the size distribution of randomly annotated instances on performance. We statistic the median size of all instances on the AC3 dataset, and make two sets of annotated instances with a comparable number of annotated voxels: one set where the size of each instance is smaller than the median size, while the other set is the opposite.

We visualize the 3D annotated masks with different size distributions in Fig. 2. The corresponding quantitative segmentation results are shown in Table 1. The results based on larger and smaller annotated instances perform worse than those based on randomly annotated instances. The network can learn more diverse instance morphological structure information from randomly annotated instances and have better generalization on the testset.

size	small	random	large
VOI↓ / ARAND ↓	1.038 / 0.085	0.903 / 0.065	0.958 / 0.078

Table 1. Comparison of different segmentation results trained by instance annotation with different size distributions.

3. Visualization of Affinity Maps and Corresponding Entropy Maps

In this section, we provide more visualization of affinity maps and corresponding entropy maps, where low-entropy and high-entropy pixels indicate reliable and unreliable affinity predictions, respectively.

The final instance segmentation results are generated from the affinity maps by different post-processing algorithms. The 3D results are generated by the Waterz [2] post-processing, which integrates a watershed algorithm to generate over-segmentation and a percentile-based agglomeration algorithm to merge over-segmentation. This post-processing takes into account the computational efficiency and performance of the electronic microscopy volumetric images, whose input is the nearest neighbor affinity maps along three directions *i.e.*, the lateral directions (denoted by x-axis and y-axis) and the axial direction (denoted by z-axis).

We give the affinity maps and corresponding entropy maps of the AC3 dataset in Fig. 3 and the Cremi-C dataset in Fig. 4. Following [5], the resolution of the input patch to the network is set as $18 \times 160 \times 160$, which is obtained by random cropping from the provided volume. Thus, the adjacent output patches of the entropy map are not continuous, which has no effect on the performance since we perform the affinity cross supervision for each patch.

The 2D results are generated by the Mutex [6] post-processing, which optimizes a long-range affinity graph to cluster pixels into instances. Following [6, 3], the input of the post-processing includes multiple short-range and long-range affinity maps along two directions (denoted by x-axis and y-axis), and the graph edges span 1,3,5,9,27 pixels. We provide two examples of affinity graph representation for the BBBC039v1 dataset in Fig. 5 and the CVPPP dataset in Fig. 6, respectively.

Method	#Params (M)	FLOPs (GMAC)
$f(\theta_1)$	1.48	87.29
$f(\theta_2)$	1.48	87.47

Table 2. Computational Complexity on two beanches.

4. Computational Complexity Analysis

The parameters and FLOPs for two branches in our method with an input size of $18 \times 160 \times 160$ are shown in 2. $f(\theta_2)$ has a slightly higher complexity with similarity learning. Note that, however, we only use one branch $f(\theta_2)$ during inference.

References

[1] Emran Mohammad Abu Anas, Saman Nouranian, S Sara Mahdavi, Ingrid Spadinger, William J Morris, Septimu E Salcudean, Parvin Mousavi, and Purang Abolmaesumi. Clinical target-volume delineation in prostate brachytherapy using residual neural networks. In *MICCAI*. Springer, 2017. 2

[2] Jan Funke, Fabian Tschopp, William Grisaitis, Arlo Sheridan, Chandan Singh, Stephan Saalfeld, and Srinivas C Turaga. Large scale image segmentation with structured loss based deep learning for connectome reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(7):1669–1680, 2019. 3

[3] Wei Huang, Shiyu Deng, Chang Chen, Xueyang Fu, and Zhiwei Xiong. Learning to model pixel-embedded affinity for homogeneous instance segmentation. In *AAAI*, 2022. 3

[4] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 2

[5] Kisuk Lee, Jonathan Zung, Peter Li, Viren Jain, and H Sebastian Seung. Superhuman accuracy on the snemi3d connectomics challenge. *arXiv preprint arXiv:1706.00120*, 2017. 2, 3

[6] Steffen Wolf, Alberto Bailoni, Constantin Pape, Nasim Rahaman, Anna Kreshuk, Ullrich Köthe, and Fred A Hamprecht. The mutex watershed and its objective: Efficient, parameter-free graph partitioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 3

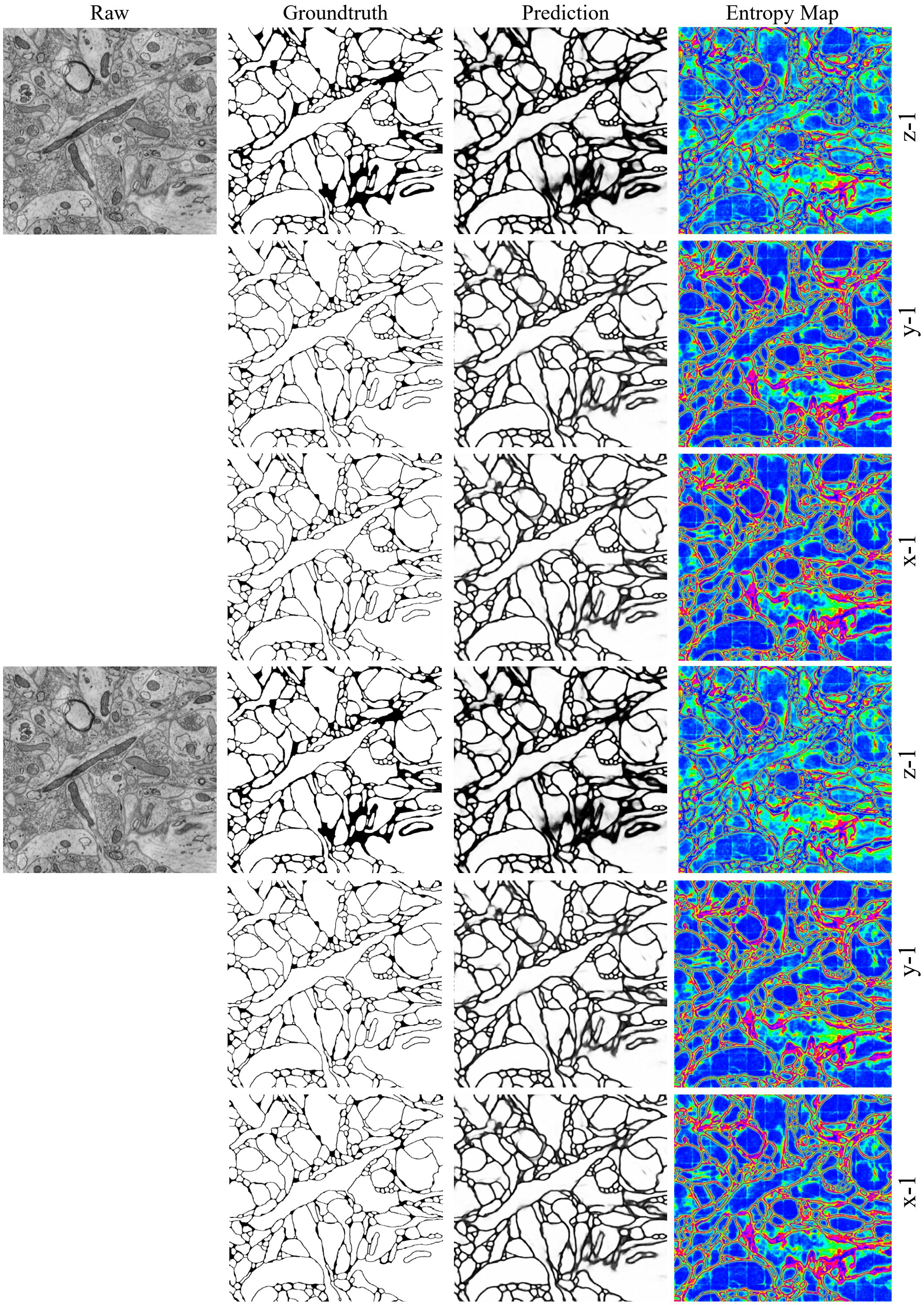

Figure 3. Two visualized examples of affinity maps in x, y, and z directions and corresponding entropy maps on the AC3 dataset.

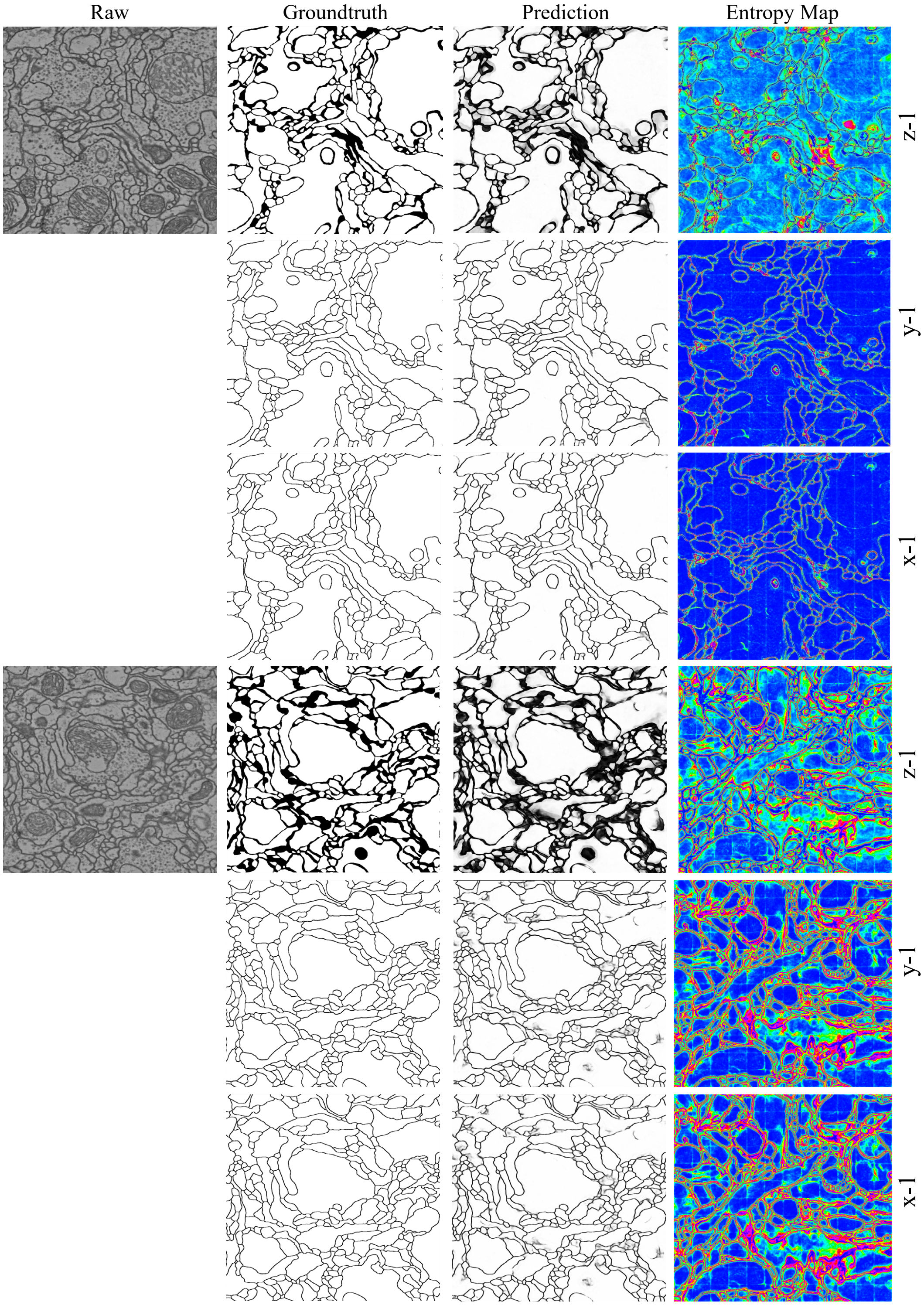

Figure 4. Two visualized examples of affinity maps in x, y, and z directions and corresponding entropy maps on the Cremi-C dataset.

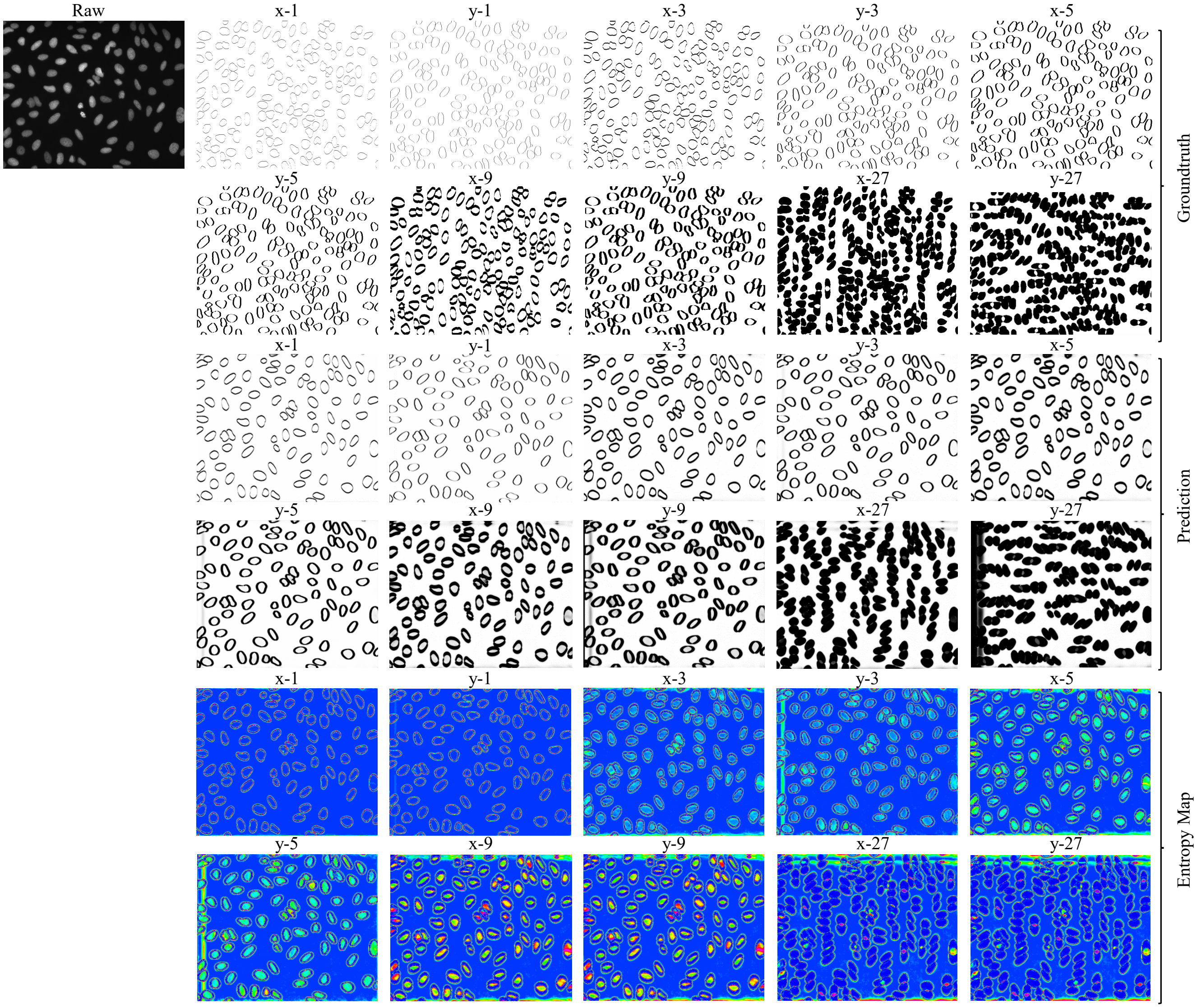

Figure 5. A visualized example of affinity maps in x and y directions and corresponding entropy maps on the BBBC039v1 dataset.

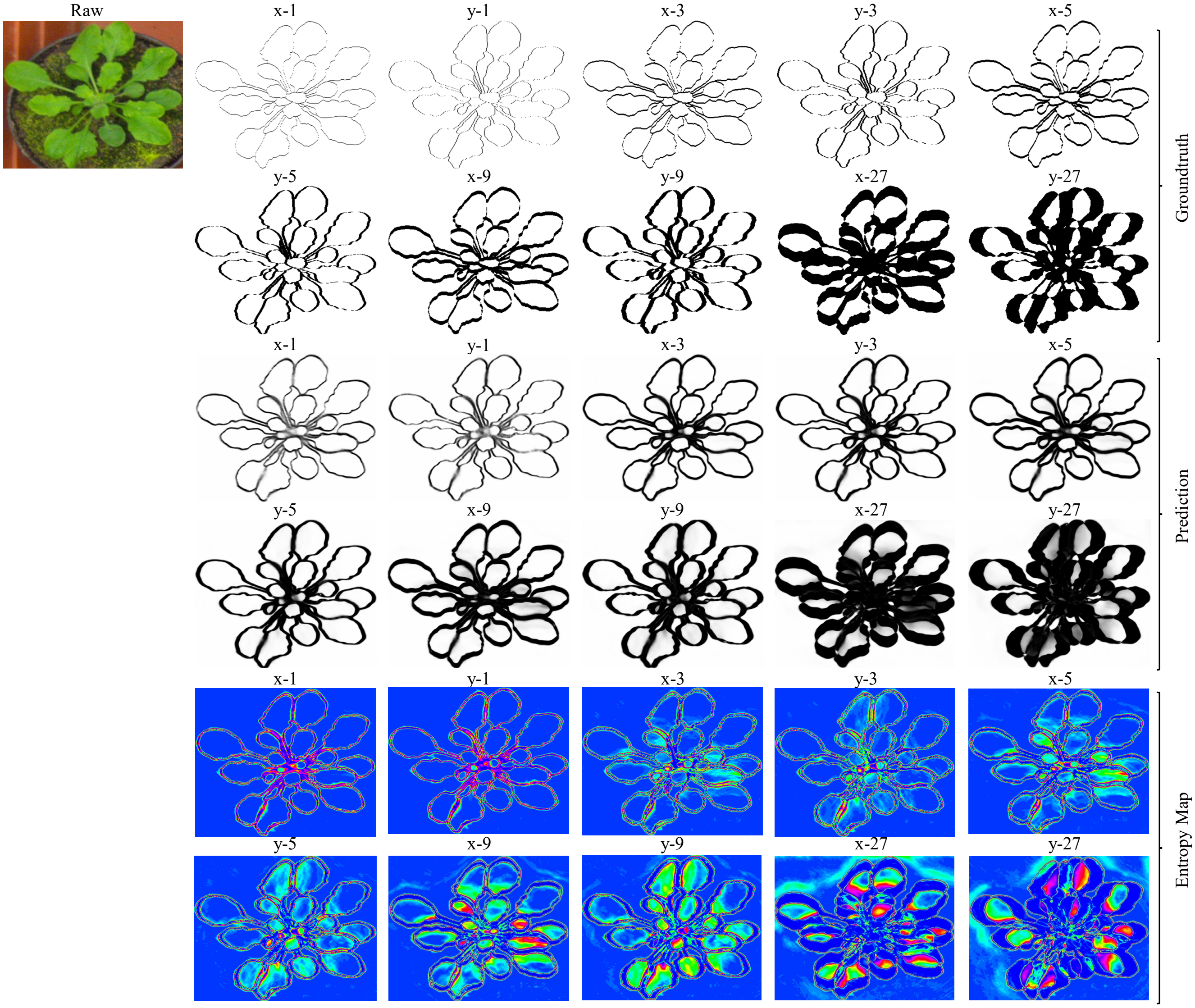

Figure 6. A visualized example of affinity maps in x and y directions and corresponding entropy maps on the CVPPP dataset.