Scellseg: a style-aware cell instance segmentation tool with pre-training and contrastive fine-tuning

Dejin Xun[1], Deheng Chen[2], Yitian Zhou[3], Volker M. Lauschke[3,4,5], Rui Wang[2*], Yi Wang[1,6,7*]

1. Pharmaceutical Informatics Institute, College of Pharmaceutical Sciences, Zhejiang University, Hangzhou, Zhejiang 310058, China
2. State Key Lab of CAD&CG, Zhejiang University, Hangzhou, Zhejiang 310058, China
3. Department of Physiology and Pharmacology, Karolinska Institutet, Stockholm, Sweden
4. Dr Margarete Fischer-Bosch Institute of Clinical Pharmacology, Stuttgart, Germany
5. University of Tuebingen, Tuebingen, Germany
6. Innovation Institute for Artificial Intelligence in Medicine of Zhejiang University, Hangzhou, Zhejiang 310018, China
7. State Key Laboratory of Component-based Chinese Medicine, Tianjin University of Traditional Chinese Medicine, Tianjin 300193, China

Corresponding Authors:
Prof. Yi Wang, zjuwangyi@zju.edu.cn
Prof. Rui Wang, ruiwang@zju.edu.cn

Abstract

Deep learning-based cell segmentation is increasingly utilized in cell biology and molecular pathology, due to massive accumulation of diverse large-scale datasets and excellent performance in cell representation. However, the development of specialized algorithms has long been hampered by a paucity of annotated training data, whereas the performance of generalist algorithm was limited without experiment-specific calibration. Here, we present a deep learning-based tool called Scellseg consisted of novel pre-trained network architecture and contrastive fine-tuning strategy. In comparison to four commonly used algorithms, Scellseg outperformed in average precision on three diverse datasets with no need for dataset-specific configuration. Interestingly, we found that eight images are sufficient for model tuning to achieve satisfied performance based on a shot data scale experiment. We also developed a graphical user interface integrated with functions of annotation, fine-tuning and inference, that allows biologists to easily specialize their own segmentation model and analyze data at the single-cell level.

Introduction

Image-based single cell profiling is widely used in biological, pharmaceutical and medical applications, including in quantitative cytometry (1), spatial transcriptomics (2), high-content drug screening (3) and cancer metastasis analysis (4). However, due to a lack of robust and facile single-cell analysis algorithms, average profiling has remained the most commonly used method which may cause loss of information and mislead interpretation of feature associations (5). In recent years, deep learning has revolutionized the field of computer vision (6) and catalyzed the advancement of single cell segmentation methods.

Differences across cell types, microscopy instruments, treatment methods, imaging modalities, and staining protocols can generate cell images with considerable diversity. As a consequence, cell segmentation algorithms were mostly developed for specific datasets (7-9) and these methods performed poorly when applied to other styles of cell images. To overcome this limitation, generalist algorithms have been developed. In 2018, a data science bowl challenge tried to segment nuclei from a large number of different styles of microscopy images using 841 diverse images containing 37,333 nuclei (10). Inspired by this competition, Stringer el al. annotated 608 images containing more than 70,000 segmented objects and developed a generalist algorithm named Cellpose, with an excellent performance in segmenting cell bodies from many image styles (11). However, while these generalist algorithms outperformed previous methodologies, their applicability remains limited and they cannot segment well in all kinds of images styles. Therefore, adaptation of segmentation models from the source domain to the target domain remains an important question.

Fine-tuning of pre-trained models has been successfully used in computer vision (12-15) and natural language processing (16-17) due to its lower input requirements and more rapid convergence to a better performance. For cell instance segmentation, there have been several seminal attempts, such as transferring a model pre-trained on in vitro data to in situ tissue images (18), or fine-tuning a pre-trained nuclear segmentation model to satisfy different needs from laboratories, facilities or institutions (19). However, these studies did not keep up with the development of modern cell segmentation algorithms, which have better instance representation and trained by more generalized datasets. Besides, there were huge demands in segmenting other cell-like instances, such as C. elegans (20), but specialized evaluation datasets for these experiments had not been developed. Hence, the development of a high-performance universal computational pipeline based on the fine-tuning of pre-trained models remains a challenging but important objective in automated image analysis.

In this work, we established a fine-tuning pipeline for cell segmentation algorithms and present a style-aware cell segmentation architecture named Scellseg based on attention mechanisms and hierarchical information for improved extraction and utilization of style features. We furthermore incorporate a contrastive learning strategy to leverage information from unlabelled and pre-trained data. To evaluate the generalizability of the pipeline, we benchmarked our model on three fundamentally different styles of data from C. elegans, label-free phase-contrast cell images, and subcellular organelles. Furthermore,

we provide estimates about the extent of data required to fine-tune a satisfying model and demonstrate how instance representation and pre-trained datasets can influence model transferability. To facilitate uptake of this pipeline, we developed a graphical user interface (GUI) can conduct annotation, fine-tuning and inference, thus making the model accessible for a wide range of users without coding experience. The model can be found at https://github.com/cellimnet/scellseg-publish.

Keywords

cell instance segmentation, domain adaptation, fine-tuning, contrastive learning, style transfer

Results

Design of Scellseg with pre-trained architecture and contrastive fine-tuning strategy

We first established a pre-trained and fine-tuning pipeline for the cell segmentation model. For initial training we utilized a large hand-labelled dataset from Cellpose (11). Then, the model can be used for inference on new experiments and, if segmentation performance was not satisfactory, such as fail of detecting instances or boundary of segmentation instance cannot meet the needs of downstream analysis, several images from the data pool can be annotated and fine-tuning of the model will be performed using the new labelled data (shot data) (Fig. 1A). This workflow generated a style-aware structure to better extract and comprehend style-related information and developed a new fine-tuning strategy based on contrastive learning (21-25) to better make use of diverse data features, including the unlabelled data (query data) and pre-trained data. The resulting model, which we named Scellseg, contains two branches, a mask branch to compute the segmentation map of input and a contrast branch to explore the information between three types of data (Fig. 1B). The mask branch is utilized during pre-training, fine-tuning and inference, whereas the contrast branch is only applied during fine-tuning.

The mask branch utilized the Cellpose algorithm because of its superior performance in cell segmentation. Cellpose is a member of the U-Net (26) family of algorithms that consist of a downsampling pass that extracts features from input data, an upsampling pass that organizes different features to fit for the final task, and a concatenation operation that relays the information extracted from downsampling process to the upsampling pass. Here, we re-divided this structure, the last Conv Unit was split from upsampling pass and named as Tasker, and the left was named as Extractor, containing the downsampling, upsampling and concatenation parts. Extractor can be used as a general feature extractor and Tasker can be adapted for different computer vision tasks, such as segmentation and simultaneous class prediction (27). To

improve model sensitivity for different styles, we followed the operation in Attention U-Net (28) and added attention gates (AGs) when passing the features extracted from downsampling to the upsampling pass. These AGs give the feature map weights to highlight salient features useful for a specific task and suppress feature activation in irrelevant regions. We used dense units from Hover-Net (27) to consider the information from early layers, aiming to delineate accurate object boundaries. To consider different-level style information, we also fed corresponding hierarchical style embeddings into these dense units (Fig. 2).

Conventional fine-tuning strategies only use labelled data and, thus, the large extent of unlabelled data, and data used to pre-train the model has not been exploited well. To augment data utilization, we developed a new fine-tuning method to make better use of all these three resources based on contrastive learning (CFT). During fine-tuning, the mask branch is used to compute the segmentation map of shot data and then a segmentation loss function (see Methods section for details) is used for computing the loss between the output and ground truth to optimize the model. The contrast branch is used to compute the respective style embeddings of shot, query and contrast data (subset of the pre-trained data) and then a contrast loss function was designed to minimize the difference between embeddings of shot and query data from the same experiment while maximizing the difference between embeddings of shot and contrast data (Fig. 1B). The loss function was defined as:

$$L_{contrast} = \frac{MSE(shot,query)}{MSE(shot,contrast)+10^{-5}} \quad (1)$$

where MSE represents mean square error and was used to compute the difference between embeddings, and 10-5 was added to prevent divisor from equaling to zero. This contrast loss was added into the segmentation loss function, so the final loss function was defined as:

$$L_{total} = L_{segmentation} + L_{contrast} \times Sigmoid(\alpha) \quad (2)$$

Where α is a scalar to control the weight of contrast loss, also learnt during the fine-tuning process. A sigmoid function was used to assure that the coefficient of contrast loss changed smoothly between zero and one. Then the total loss optimizes the model via backpropagation.

Transferability of Scellseg with contrastive fine-tuning strategy on three evaluation datasets

Three datasets were used to evaluate the transferability of models in this work, named BBBC010_elegans (20), LIVECell_bv2(29) and mito (created and manually labelled by ourselves), representing three levels of cell-like images from model organism, cells to subcellular structures (Fig. 3A). Combined, the datasets contained 230 images and 91,024 segmentation objects. We visualized the distribution of areas and numbers for cells per image (Fig. S1). The average areas for three datasets are about 1000, 150 and 100,000, and numbers of instance in each image ranged from 2 to 2,815. We used t-distributed stochastic neighbor embedding (30) (t-SNE) to visualize the style embeddings (see

definition in ref. 11) of these evaluation datasets together with pre-trained datasets and noted that the style of data in each evaluation dataset was the major cluster determinant (Fig. 3B).

To compare the influence of different instance representation, we benchmarked Scellseg against four other models, U-Net2 (26), U-Net3 (31), HoVer (27) and Cellpose (11). These four models were set with identical network structure and pre-trained with the same datasets and training strategies. We used the training data of each dataset to fine-tune the model at ten different random states, most models achieved great improvements after fine-tuning, for the BBBC010_elegans dataset, all models yielded at least 35% higher average precision, and for U-Net3 model, fine-tuning even yielded an increase of 62.1%. The different models differed drastically in performance, and representation of Cellpose (used in Cellpose and Scellseg model) outperformed other methods. Scellseg with contrastive fine-tuning achieved the best performance on all three evaluation datasets, especially on the BBBC010_elegans dataset. At the universally used intersection over union (IoU) threshold of 0.5, our method and Cellpose all achieved high average precision for BBBC010_elegans (0.882 for Scellseg; 0.868 for Cellpose), LIVECell_bv2 (0.783 for Scellseg; 0.784 for Cellpose) and mito (0.927 for Scellseg; 0.922 for Cellpose). In contrast, Scellseg performed considerably better at higher thresholds, such as 0.75 on all three data sets ([0.670, 0.493, 0.634] for Scellseg compared to [0.587, 0.475, 0.571] for Cellpose, respectively; Fig. 3C-F).

We displayed the segmentation results of Scellseg before and after using the contrastive fine-tuning strategy and found that the fine-tuned model had exhibited considerably better instance detection (Fig. 4). Fine-tuning also resulted in the ability of distinguishing between adjacent cells and allowed segmentation of all mitochondria scattered around the nuclei in mito dataset. Furthermore, our contrastive fine-tuning strategy outperformed the classic method on the Cellpose test set after fine-tuning on the three evaluation datasets. However, as expected, all re-trained models suffered a sharp decline compared with the initial generalization ability (Fig. S2).

Pre-trained dataset scale experiments

To explore how the pre-trained dataset can influence model transferability, we used different subsets of the Cellpose training set. The initial subset (Sneuro) only contains one style of images from the Cell Image Library (32), then additional styles of images were sequentially added, such as fluorescent cells (Sfluor), non-fluorescent and membrane-labelled cells (Scell), other microscopy data (Smicro), as well as non-microscopy images ("Sgeneral", corresponding to the full Cellpose train set; Fig. 5A).

We pre-trained Scellseg with Sneuro, Sfluor, Scell, Smicro and firstly tested the generalization ability of each model by applying it directly without any adaptation on three evaluation datasets. For BBBC010_elegans, the model trained with Sneuro, Sfluor, and Scell does not result in successful structure recognition until the pre-trained dataset contains microscopy instances with structures beyond cells. For LIVECell_bv2, the generalization ability also increased with the richness of the dataset from Sneuro to Smicro, and model trained on Smicro even outperformed the model trained on Sgeneral. For the

mito dataset, surprisingly the model trained Sfluor and Smicro outperformed all others (Fig. 5B).

Next, we tested the transferability of each model (Fig. 5C). As expected, the transferability of Scellseg increased with the richness of the dataset from Sneuro to Smicro, and Scellseg pre-trained with Smicro achieved similar transfer performance on three evaluation datasets compared to Scellseg pre-trained with Sgeneral ([0.555, 0.451, 0.564] and [0.554, 0.454, 0.557], mean average precision [mAP] of Smicro and Sgeneral respectively). On the BBBC010_elegans dataset, a model pre-trained on Sneuro achieved only very poor transfer performance (0.013 mAP) and performance increased substantially only after the addition of different styles of fluorescent images (0.436 mAP). As more cell-like images added, performance increased further to 0.554 with the full pre-trained dataset.

Shot data scale experiments and ablation experiments

To explore the extent of annotated data required for fine-tuning, we made a shot data scale experiment on these evaluation datasets. We set 10 scale levels, and for each shot number, we randomly sampled 10 times from the training pool to fine-tune the model, followed by testing of transferability. For this evaluation, we focused on Scellseg with CFT and Cellpose with classical fine-tuning because these models clearly outperformed the other three algorithms. For all datasets, we observed that initial performance is relatively low with large variance (Fig. 6A). As the shot number increases, the performance improves drastically and variance becomes smaller and Scellseg significantly outperformed Cellpose. For BBBC010_elegans, LIVECell_bv2 and mito dataset, Scellseg with CFT get [2.0%, 4.8%, 18.5%] final improvement respectively compared to [4%, 6%, 12.2%] for Cellpose with classical fine-tuning. We conducted curve fitting using Hyperbola function for each method per dataset for further inspection of the transferability across different shot numbers. The results show that, when increasing shot number, different methods converged to different values and the mAP converged differently across datasets. For mito dataset, mAP persistently increased while for BBBC010_elegans, the rate of its convergence is relatively fast, and whatever the dataset is, performance plateaued at about eight shots. Similar results were obtained using the mean Aggregated Jaccard Index (33) as a means to evaluate transfer performance (Fig. S3).

To verify the function of our contrastive fine-tuning strategy, we conducted ablation experiments. Importantly, our contrastive fine-tuning strategy outperformed Scellseg using the classic fine-tuning method at different shot-number experiments on all three evaluation datasets (Fig. 6B). Due to the similar performance of the model when “only” trained on Smicro, we conducted the same shot data scale experiments and ablation experiments as Scellseg pre-trained on the Sgeneral dataset and, excitingly, our style-aware pipeline worked and again outperformed Cellpose with classic fine-tuning strategy (Fig. S4).

Graphical user interface

To also make the design of cell segmentation models accessible for scientists without coding experience, we designed a GUI (Fig. 7). Firstly, we re-organized the GUI of Cellpose

and divided the functionality into three parts, i) view and draw, ii) fine-tuning and iii) inference.

For basic annotations, users can modify the mask of an instance directly at single-pixel resolution without deleting the whole mask. We also developed a cell list management system, users can control to delete or edit the corresponding mask and provide annotations, thereby allowing to provide ground truth for segmentation and cell class prediction simultaneously. Furthermore, users can easily save or load cell lists.

In the second functionality, users can fine-tune the pre-trained model with their own manually labelled data. The system allows users to choose a pre-trained model from Scellseg, Cellpose and HoVer. Furthermore, each model can be combined with either the contrastive or classic fine-tuning strategy, presented above. We hope this will not only give biologists and pathologists more flexibility and versatility for their image analysis tasks, but also help algorithm engineers to easily conduct experiments to study such pre-trained and fine-tuning pipelines.

Finally, users can use the fine-tuned model to conduct inference either for one image or use batch inference. After annotation or inference, users can also download images of each single instance for further analysis.

Discussion

Accurate cell instance segmentation is still a challenging task for many laboratories. Although generalization models have been developed, these typically require large annotated datasets, collection of which is time consuming, particularly when a large number of styles of segmented objects are supposed to be covered. To augment data utilization, we firstly established a pipeline for the fine-tuning of pre-trained cell segmentation algorithms. On this basis, we proposed a style-aware pipeline, yielding the best transferability on three different benchmarking datasets. Specifically, the work achieved three main innovations: Firstly, we refined the architecture of Cellpose through introducing attention mechanisms and hierarchical information, making the model more sensitive to different styles. Secondly, we implemented a contrastive fine-tuning strategy to leverage the information from both unlabelled and pre-trained data. Finally, we organized a benchmarking dataset containing three levels of cell images for further use in segmentation algorithm development.

Importantly, after fine-tuning, AP for BBBC010_elegans and mito can reach up to about 0.9 at a threshold of 0.5, more than 37% and 36% improvements respectively, resulting in model performance that is generally acceptable for researchers to conduct reliable downstream analysis. Our benchmarking showed that among four different instance representations, topological maps generated by the Cellpose model constituted the best way to introduce rich instance information. Results could be further improved by our style-aware pipeline, which exhibited the best transferability on all three evaluation datasets, indicating that introducing such style relevant information can benefit the fine-tuning process. Notably, while generalization ability overall declined after fine-tuning to a specific task, our contrastive fine-tuning considerably improved generalizability.

Furthermore, emerging methods like continual learning (34) are also worth to investigate in this context.

In the pre-trained dataset scale experiments, we observed that transferability of Scellseg-CFT increased with the richness of the pre-trained dataset, suggesting that our Scellseg-CFT pipeline can also benefit from large-scale and high-diversity datasets. Notably, the generalization ability of Scellseg pre-trained with Sfluor containing different styles of fluorescent images, outperformed all other models on the mito evaluation dataset, indicating that model pre-training on diverse but specialized data may yield greater performance than both low-diversity specialized dataset (such as Sneuro) or high-diversity generalized datasets (such as Scell that also contains nonfluorescent images). However, we did not observe similar phenomena on transferability. We also noted that the generalization ability increased with the addition of more different microscopy instances beyond cells to other non-cell instances like C. elegans, again demonstrating the success of our style-aware pipeline.

For the shot data scale experiment, it is not surprising that performance increases with shot number. However, what excited us is that we observed a large payoff when increasing shot number from 1 to 3, whereas performance plateaued after approximately eight shots. These results are of high practical relevance as they indicate that the annotation of only about eight images is sufficient to yield a sufficiently fine-tuned model. Few-shot (35), one-shot (36) and zero-shot (37) learning strategies can be studied to further reduce the number of annotated images needed. Notably, at small shot numbers, different shot data can have very large impacts on the fine-tuning process, whereas we observed that as the shot number increases variance becomes substantially smaller. In the future, active learning (38) on cell instance segmentation promises to refine shot data selection for fine-tuning.

In this work, we did not research the influence of basic model backbone and all models were based on convolutional neural networks (CNN). In recent years, self-attention architectures (such as Transformer (39)) have shown great success and there studies have attempted to apply it to computer vision (40). Such transformer architectures have better expressive ability but require more data for accurate training. Nevertheless, we believe that such approaches will eventually provide an important improvement in computer vision compared to CNN.

By integrating attention mechanisms and hierarchical information for style-aware segmentation with a contrastive fine-tuning strategy, Scellseg features the highest transferability when benchmarked on three diverse imaging datasets against currently used segmentation methods. Scellseg optimizes cell and object recognition in diverse microscopy data and, combined with an easy-to-use GUI, can make advanced parallelized segmentation accessible also to researchers and histologists without coding experience.

Methods

The code was written in Python programming language v.3.7.4. All experiments were conducted on NVIDIA GeForce RTX 2080Ti. The deep learning framework used Pytorch

(41) v.1.7.1.

Datasets

Pre-trained dataset. We used dataset published by Stringer et al in Cellpose (11). It contains totally 608 images and over 70000 segmented instances. Same as the Cellpose, 540 images were used as train set (the last of every 8 images was chosen as valid set) and 68 images was used as test set. Here, the train set was used to pre-train the models and the test set was used to test the generalization ability in Fig. S2. Besides, a subset of the train set was used as the contrast data, it contains totally seven styles of images and each style has five images (Fig. S5).

In pre-trained dataset scale experiments, we established four other subsets from the full train set. The first level only contains 89 images from Cell Image Library, then, 199 “Cells, fluorescent” images were added in to form Sfluor, 93 “Cells, nonfluorescent” and “Cells, membranes” images were added into Sfluor to form Scell, 74 other microscopy images were added into Scell to form Smicro.

Evaluation datasets. Three datasets were used to evaluate the transferability of different models, here we called BBBC010_elegans, LIVECell_BV2 and mito. BBBBC010_elegans was downloaded from the Broad Bioimage Benchmark Collection (42), containing 100 images of roundworm C. elegans in a screen to find novel anti-infectives. There are two phenotypes in this dataset, for worms treated with ampicillin, they appear curved in shape and smooth in texture, while untreated worms appear rod-like in shape and slightly uneven in texture. The dataset provides two channels (brightfield and a GFP channel) but, in this work, we only used the brightfield channel. We discarded the images having heavily crossed instances because it is not the focus of our work, the problem may be solved by some special postprocessing algorithm (20) or introducing the z-axis information when designing the ground truth. Finally, 49 images were reserved, 10 were used as the train set and 39 were used for testing.

LIVECell_bv2 dataset was just released recently (29), consisting of 536 phase-contrast images and over 330,000 segmented instances. These images were achieved using label-free phase-contrast imaging and cells in this dataset have small spherical morphology and homogeneous population. Consistent with usage in that research, 386 were used as the train set and 152 were used for testing.

We also generated a novel dataset called Mito dataset, which consisted of 49 fluorescent images of mitochondria in high content screening study. The images were acquired by ImageXpress Micro Confocal (Molecular Devices). Each image contains two distinct channels, i.e. nuclear channel stained with Hoechst (Sigma) and mitochondria channel stained with Tetramethylrhodamine Methyl Ester (TMRM, Sigma). All these images were fully manually segmented by a single human operator (D.J.X.), like the BBBBC010_elegans dataset, 10 were used as train set and 39 were reserved for testing. Because there are no clear boundaries of individual cell, Mito dataset was enrolled to compare the performance of different segment algorithms on mitochondrial segmentation at the single-cell-level.

All these three datasets were organized in Cellpose format. The summary information can be seen in Table S1.

Models

When training a cell instance segmentation model, we usually provide raw images and corresponding masks which label the individual instance with different positive integers per image. Although different values can represent different instances in these masks, it is impractical to directly predict such masks because the max value in each mask represents the number of instances, which is different across images, model has to pre-set a very large shape of last conv unit (for example, (200, 2, 224, 224) in Pytorch (41) tensor shape format) to cover all instances, for one thing, the high or low value of each instance is unmeaningful, it will cause a serious memory waste and model cannot be employed in images with an amount of instances (for example, the max number of instances in LIVECell_bv2 dataset is 2815), for another, model may not learn well in such a high dimensionality. It is challenging to find an excellent representation of instances and till now, there has been four main methods: U-Net2 (26), U-Net3 (31), HoVer (27) and Cellpose (11).

For the classic U-Net model (usually called as U-Net2), we directly map the annotated masks to 2-classes, zero represents background and one represents instance. This method usually performs poorly on touching cells because instance information was completely discarded. In 2018, Fidel A. et al (31) introduced cell borders as the third class to make the network notice the original gap between cells (usually called as U-Net3), they yield a significant improvement compared with U-Net2. In 2019, Simon et al (27) demonstrated a state-of-art model on multiple independent multi-tissue histology image datasets, for each cell per image, they generated horizontal and vertical distance maps to bring in rich instance information, when inference, marker-controlled watershed (43) was used as the postprocessing to create the final masks. In 2020, Stringer et al (11) generated topological maps through a process of simulated diffusion from masks, and when on a test image, they used gradient tracking (44) to recover individual cells.

Here, we wanted to compare how expressive power different methods can provide, so we used the same architecture as Cellpose, only changed the final shape of convolutional layer, loss function and postprocessing to adapt to each method. Scellseg model adopted the representation of Cellpose because the best performance in the experiments.

Expect the different parts of Scellseg, all other architecture sets were same as Cellpose too.

Two-channel 224×224 images were set as input for all 5 models in this work. The primary channel contains instances you want to segment and the second optional channel can provide extra information such as nuclei channel to help the model learn better. The hierarchical level of Conv Units was set as [32, 64, 128, 256]. We computed the style embeddings through applying the average pooling on feature map of last Conv Unit and the dimensionality of each level style embeddings after being concatenated in upsampling pass is [256, 384, 448, 480].

Pre-train segmentation models

Pre-train different models with General Cellpose data. We trained five models (U-Net2, U-Net3, HoVer, Cellpose, Scellseg) with full Cellpose train set mentioned above, which contains totally 540 images, 64 images were reserved for validating.

For U-Net2 and U-Net3, a learning rate of 0.2 was too high to make models converge well, so we changed it to 0.002. For HoVer and Scellseg, the loss function is same as Cellpose, which was defined as:

$$L_{segentation} = BCE(y_{b,2}, lbl_{b,0}) + 0.5 \times MSE(y_{b,0:2}, 5 \times lbl_{b,1:3}) \quad (3)$$

Where BCE represents the binary cross-entropy loss and MSE represents the mean square error loss, y donates the ultimate output of model, lbl represents ground truth, and subscripts corresponded to respective dimensions in y or lbl, b donates the batch size, here we set to 8.

All models were trained for 500 epochs with stochastic gradient descent, the mean diameter was set to 30, all other training hyper-parameters were same as Cellpose.

Pre-train Scellseg for pre-trained dataset scale experiments. We trained four other models across different subsets of Cellpose mentioned above: Sneuro, Sfluor, Scell, Smicro. For each subset, the last of every 8 images was reserved for validating (11, 36, 47, 56, respectively). All other training hyper-parameters were same as Scellseg pre-trained with full train set.

Train logs of all models can be seen in Fig. S6.

Fine-tune segmentation models

Classic fine-tuning strategy (FT). This strategy was used for U-Net2, U-Net3, HoVer and Cellpose, when fine-tuning, batch size was set to 8, epoch was set to 100, we use Adam optimizer, the initial learning rate was set to 0.001 and every quarter of epochs it would be reduced by half. Before fed to the network, the image-mask pairs were resized, random rotated and reshaped, the ultimate shape of input is (8, 2, 224, 224).

Contrastive fine-tuning strategy (CFT). The initial α in contrast loss function was set to 0.2, initial learning rate of α was set to 0.1, every quarter of epochs it would be reduced by half too. For query and contrast data, they were resized, random cropped, random rotated and reshaped before fed to the network, the ultimate shape of input is (8, 2, 224, 224) too. Other parameters were the same as classic fine-tuning strategy.

Whatever classic or contrastive fine-tuning strategy, we fine-tuned all layers because this method won the best performance compared with fixing downsampling part or the whole extractor (Fig. S7). For each dataset, we computed the instance diameter using shot data without using the automated method provided by Cellpose, which was used in resizing current mean diameter of instances to mean diameter which model used when pre-training.

Benchmarking

Metrics. We used average precision (AP) and Aggregated Jaccard Index (AJI), described in Cellpose to evaluate the segmentation performance, notably, the definition of this AP is different from AP used in LIVECell paper (so it cannot be compared with each

other). Expect the transferability of Scellseg with contrastive fine-tuning strategy on three evaluation datasets compared with other four models experiment, we averaged the AP or AJI over IoU from 0.5 to 0.95 with a step size of 0.05 for convenient comparison and reserving the overall performance information simultaneously. In mathematical notation:

$$\mathrm{mAP} = (AP_{0.50} + AP_{0.55} + \cdots + AP_{0.9} + AP_{0.95}) \,/\, 10 \quad (4)$$

$$\mathrm{mAJI} = (AJI_{0.50} + AJI_{0.55} + \cdots + AJI_{0.9} + AJI_{0.95}) \,/\, 10 \quad (5)$$

See the respective papers for the detail definitions.

Shot data scale experiments. We totally set 10 scale levels, for BBBC010_elegans and mito is [1, 2, 3, 4, 5, 6, 7, 8, 9, 10] and for LIVECell_bv2 is [1, 2, 4, 8, 16, 32, 64, 128, 256, 386]. For each shot number experiment, we ransom sampled 10 times from the train set to fine-tune the pre-trained model. To eliminate the error of different train data, the random state was entirely same across models. For example, for mito dataset, there are totally 10 images in the train set, in a 3-shot experiment, we sampled the 9th, 5th, and 2nd images on one occasion, it will be used as the train data for all models.

Statistical Analysis

All figures were made using GraphPad PRISM 8.0 software (GraphPad Software, Inc., CA, USA). All graphs display mean values, and the error bars donate the standard deviation (SD). Statistical analysis was conducted with a two-way repeated measures analysis of variance (ANOVA, matching: stacked) followed by Sidak's multiple comparisons test in Fig. 3C-E and two-way ANOVA in Fig. 6A. A nonlinear regression curve fit was performed using Hyperbola function in Fig. 6A, the mathematical notation of Hyperbola function is defined as:

$$\mathrm{Y} = \frac{B_{max} * X}{K_d + X} \quad (6)$$

The detail definition of this function can be seen in GraphPad.

Data and code availability

Three evaluation datasets used in this work will be made available at https://scellseg-data.s3.cn-northwest-1.amazonaws.com.cn/evaluation_datasets.zip and the code will be made available at https://github.com/cellimnet/scellseg-publish upon publication.

Acknowledgements

The authors are grateful for the support from ZJU PII-Molecular Devices Joint Laboratory and support from "Medicine + X" interdisciplinary center of Zhejiang University. Y.W. is supported by National Natural Science Foundation of China (No. 81822047, No. 82173941), the Innovation Team and Talents Cultivation Program of National Administration of Traditional Chinese Medicine (No. ZYYCXTD-D-202002). R.W. is supported by National Natural Science Foundation of China (No. 61872319) and Natural

Science Foundation of Zhejiang Provincial (No. LR18F020002).

Author Contributions

Y.W. and R.W. proposed the concept and supervised the overall project. D.J.X. established the pipeline, organized datasets, written the code of pipeline, conducted experiments, performed data analysis, designed and finished the code writing of graphical user interface. D.H.C. participated in part of the code writing of graphical user interface. D.J.X., V.M.L., Y.W., R.W., Y.T.Z. participated in the preparation of the manuscript.

Reference

1. Y. Mun, Single-cell cytometry via multiplexed fluorescence prediction by label-free reflectance microscopy. Sci. Adv. 7, eabe0431 (2021).
2. V. Petukhov, R. J. Xu, R. A. Soldatov, P. Cadinu, K. Khodosevich, J. R. Moffitt, P. V. Kharchenko, Cell segmentation in imaging-based spatial transcriptomics. Nat. Biotechnol. (2021), doi:10.1038/s41587-021-01044-w.
3. E. Moen, D. Bannon, T. Kudo, W. Graf, M. Covert, D. Van Valen, Deep learning for cellular image analysis. Nat. Methods. **16**, 1233–1246 (2019).
4. C. Pan, O. Schoppe, A. Parra-Damas, R. Cai, M. I. Todorov, G. Gondi, B. von Neubeck, N. Böğürcü-Seidel, S. Seidel, K. Sleiman, C. Veltkamp, B. Förstera, H. Mai, Z. Rong, O. Trompak, A. Ghasemigharagoz, M. A. Reimer, A. M. Cuesta, J. Coronel, I. Jeremias, D. Saur, A. Acker-Palmer, T. Acker, B. K. Garvalov, B. Menze, R. Zeidler, A. Ertürk, Deep learning reveals cancer metastasis and therapeutic antibody targeting in the entire body. Cell. **179**, 1661-1676.e19 (2019).
5. M. H. Rohban, H. S. Abbasi, S. Singh, A. E. Carpenter, Capturing single-cell heterogeneity via data fusion improves image-based profiling. Nat. Commun. **10**, 2082 (2019).
6. J. Chai, H. Zeng, A. Li, E. W. T. Ngai, Deep learning in computer vision: A critical review of emerging techniques and application scenarios. Machine Learning with Applications. **6**, 100134 (2021).
7. A. Tareef, Y. Song, H. Huang, D. Feng, M. Chen, Y. Wang, W. Cai, Multi-pass fast watershed for accurate segmentation of overlapping cervical cells. IEEE Trans. Med. Imaging. **37**, 2044–2059 (2018).
8. N. E. Krasowski, T. Beier, G. W. Knott, U. Kothe, F. A. Hamprecht, A. Kreshuk, Neuron segmentation with high-level biological priors. IEEE Trans. Med. Imaging. **37**, 829–839 (2018).
9. N. O. Loewke, S. Pai, C. Cordeiro, D. Black, B. L. King, C. H. Contag, B. Chen, T. M. Baer, O. Solgaard, Automated cell segmentation for quantitative phase microscopy. IEEE Trans. Med. Imaging. **37**, 929–940 (2018).

10. J. C. Caicedo, A. Goodman, K. W. Karhohs, B. A. Cimini, J. Ackerman, M. Haghighi, C. Heng, T. Becker, M. Doan, C. McQuin, M. Rohban, S. Singh, A. E. Carpenter, Nucleus segmentation across imaging experiments: the 2018 Data Science Bowl. Nat. Methods. **16**, 1247–1253 (2019).

11. C. Stringer, T. Wang, M. Michaelos, M. Pachitariu, Cellpose: a generalist algorithm for cellular segmentation. Nat. Methods. 18, 100–106 (2021).

12. B. Chu, V. Madhavan, O. Beijbom, J. Hoffman, T. Darrell, Best practices for fine-tuning visual classifiers to new domains. Proc. Eur. Conf. Comput. Vis. 435–442 (2016).

13. Y. Guo, H. Shi, A. Kumar, K. Grauman, T. Rosing, R. Feris, SpotTune: transfer learning through adaptive fine-tuning. Proc. IEEE Conf. Comput. Vis. Pattern Recog. 4805-4814 (2019).

14. E. Cetinic, T. Lipic, S. Grgic, Fine-tuning convolutional neural networks for fine art classification. Expert Syst. Appl. **114**, 107–118 (2018).

15. K. You, Z. Kou, M. Long, J. Wang, Co-Tuning for transfer learning. Conference and Workshop on Neural Information Processing Systems. 17236–17246 (2020).

16. J. Howard, S. Ruder, Universal language model fine-tuning for text classification. Annual Meeting of the Association for Computational Linguistics. 328-339 (2018).

17. C. Lee, K. Cho, W. Kang, Mixout: effective regularization to finetune large-scale pretrained language models. International Conference on Learning Representations. (2020).

18. Y. Jin et al, http://biorxiv.org/lookup/doi/10.1101/2021.02.28.433289 (2021).

19. G. Zaki, P. R. Gudla, K. Lee, J. Kim, L. Ozbun, S. Shachar, M. Gadkari, J. Sun, I. D. C. Fraser, L. M. Franco, T. Misteli, G. Pegoraro, A Deep Learning Pipeline for Nucleus Segmentation. Cytom. Part A. **97**, 1248–1264 (2020).

20. C. Wählby, L. Kamentsky, Z. H. Liu, T. Riklin-Raviv, A. L. Conery, E. J. O'Rourke, K. L. Sokolnicki, O. Visvikis, V. Ljosa, J. E. Irazoqui, P. Golland, G. Ruvkun, F. M. Ausubel, A. E. Carpenter, An image analysis toolbox for high-throughput C. elegans assays. Nat. Methods. **9**, 714–716 (2012).

21. I. Cho, Y. Huo, S. Yoon, Weakly-supervised contrastive learning in path manifold for monte carlo image reconstruction. ACM Trans. Graph. **40**, 1-14 (2021).

22. B. Gunel, J. Du, A. Conneau, V. Stoyanov, Supervised contrastive learning for pre-trained language model fine-tuning. Proc. Int. Conf. Learn. Represent. (2021).

23. Y. Yu, S. Zuo, H. Jiang, W. Ren, T. Zhao, C. Zhang, Fine-tuning pre-trained language model with weak supervision: a contrastive-regularized self-training approach. Annual Meeting of the Association for Computational Linguistics. Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 1063–1077 (2021).

24. Y. Tian, C. Sun, B. Poole, D. Krishnan, C. Schmid, P. Isola, What Makes for Good Views for Contrastive Learning? Conference and Workshop on Neural Information Processing Systems. (2020).

25. T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A Simple Framework for Contrastive Learning of Visual Representations. Proceedings of Machine Learning Research. **119**, 1597-1607 (2020).

26. O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional Networks for Biomedical Image Segmentation. Medical Image Computing and Computer-Assisted Intervention. **9351**, 234-241 (2015).

27. S. Graham, Q. D. Vu, S. E. A. Raza, A. Azam, Y. W. Tsang, J. T. Kwak, N. Rajpoot, Hover-Net: Simultaneous segmentation and classification of nuclei in multi-tissue histology images. Med. Image Anal. **58**, 101563 (2019).

28. O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Y. Hammerla, B. Kainz, B. Glocker, D. Rueckert, Attention U-Net: Learning pytttWhere to Look for the Pancreas. The International Conference on Medical Imaging with Deep Learning. (2018).

29. C. Edlund, T. R. Jackson, N. Khalid, N. Bevan, T. Dale, A. Dengel, S. Ahmed, J. Trygg, R. Sjögren, LIVECell—A large-scale dataset for label-free live cell segmentation. Nat. Methods. **18**, 1038–1045 (2021).

30. Lvd. Maaten, G. Hinton, Visualizing data using t-SNE. J. Machine Learning Res. **9**, 2579–2605 (2008).

31. F. A. Guerrero-Pena, P. D. Marrero Fernandez, T. Ing Ren, M. Yui, E. Rothenberg, A. Cunha, Multiclass Weighted Loss for Instance Segmentation of Cluttered Cells. Proc. IEEE Int. Conf. Image Process. 2451–2455 (2018).

32. W. Yu, H. K. Lee, S. Hariharan, W. Y. Bu, S. Ahmed, https://doi.org/10.7295/W9CCDB6843.

33. N. Kumar, R. Verma, S. Sharma, S. Bhargava, A. Vahadane, A. Sethi, A Dataset and a Technique for Generalized Nuclear Segmentation for Computational Pathology. IEEE Trans. Med. Imaging. **36**, 1550–1560 (2017).

34. G. Zeng, Y. Chen, B. Cui, S. Yu, Continual learning of context-dependent processing in neural networks. Nat. Mach. Intell. **1**, 364–372 (2019).

35. D. Das, C. S. G. Lee, A two-stage approach to few-shot learning for image recognition. IEEE Trans. Image Process. **29**, 3336–3350 (2020).

36. C. Michaelis, I. Ustyuzhaninov, M. Bethge, A. S. Ecker, http://arxiv.org/abs/1811.11507 (2018).

37. X. Xu, I. W. Tsang, C. Liu, Complementary Attributes: A New Clue to Zero-Shot Learning. IEEE T. Cybern. **51**, 12 (2021).

38. Z. Zhou, J. Y. Shin, S. R. Gurudu, M. B. Gotway, J. Liang, Active, continual fine tuning of convolutional neural networks for reducing annotation efforts. Med. Image Anal. **71**, 101997 (2021).

39. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need. Conference and Workshop on Neural Information Processing Systems. 6000–6010 (2017).

40. Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, http://arxiv.org/abs/2103.14030 (2021).

41. A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: an imperative style, high-performance deep learning library. Conference and Workshop on Neural Information Processing Systems. (2019).

42. V. Ljosa, K. L. Sokolnicki, and A. E. Carpenter, Annotated high-throughput microscopy image sets for validation. Nat. Methods. **9**, 637–637 (2012).

43. Jierong Cheng, J. C. Rajapakse, Segmentation of clustered nuclei with shape markers and marking function. IEEE Trans. Biomed. Eng. **56**, 741–748 (2009).

44. G. Li, T. Liu, J. Nie, L. Guo, J. Chen, J. Zhu, W. Xia, A. Mara, S. Holley, S. T. C. Wong, Segmentation of touching cell nuclei using gradient flow tracking. J. Microsc. **231**, 47–58 (2008).

Figures and Tables

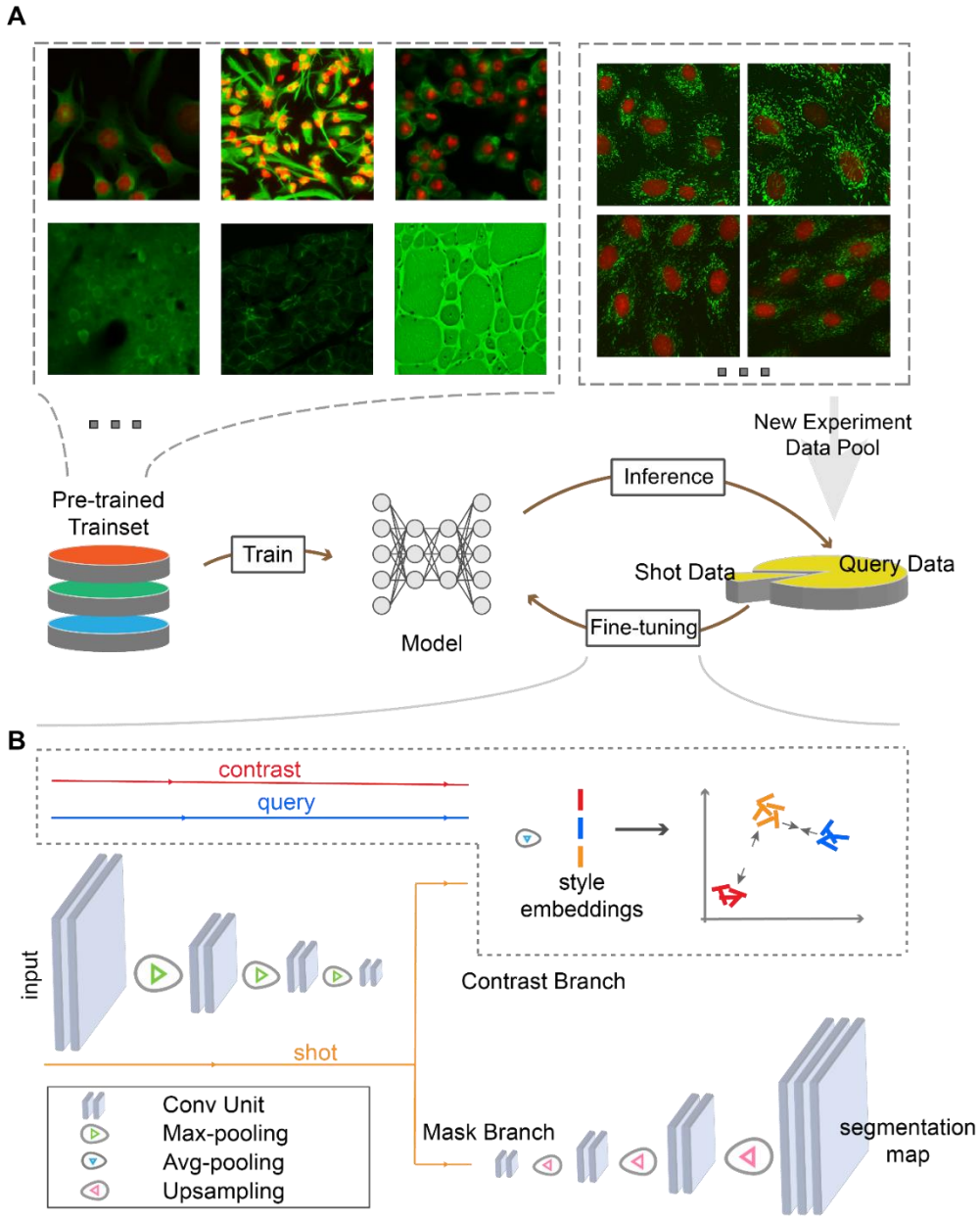

Fig. 1. Pipeline of pre-trained and fine-tuning strategy. (**A**), Overview of fine-tuning a pre-trained model for a new experiment. The shot data donate hand-labelled data while query data donate unlabelled data. (**B**), Diagram of the proposed contrastive fine-tuning strategy. The contrast data is a subset of pre-trained data. The network is a representation of U-Net family, the detailed architecture of our proposed model is shown in Fig. 2. Different colored lines and arrows mark the flow of data.

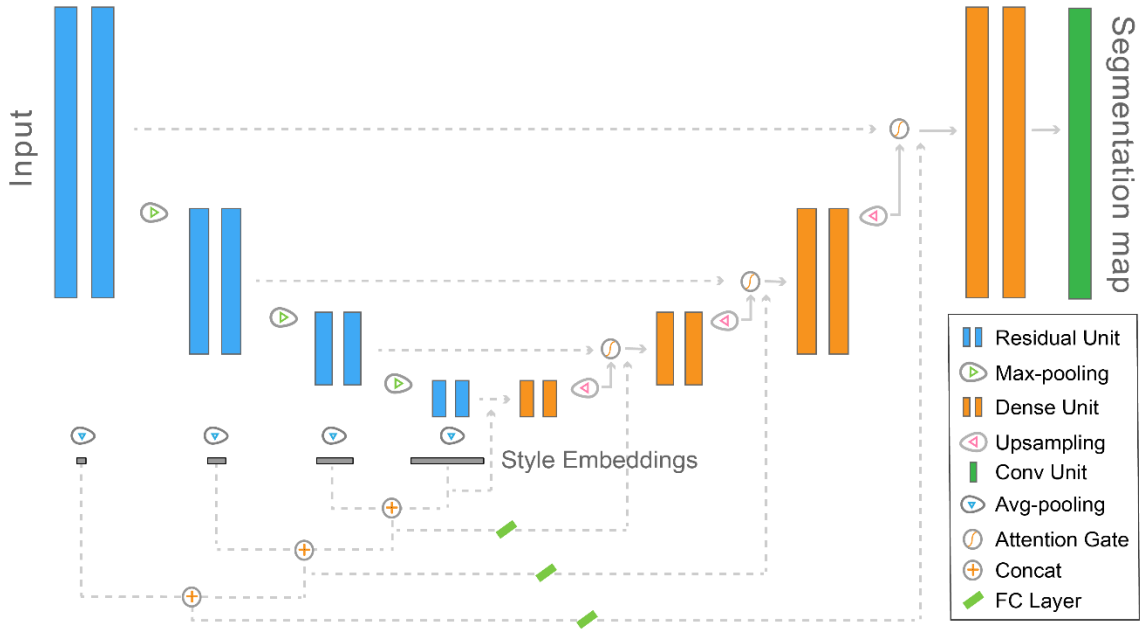

Fig. 2. Architecture of the proposed Scellseg. The Residual Unit refers to Cellpose, Dense Unit refers to Hover-Net, Attention Gate refers to Attention U-Net, Conv Unit represents a set of operations including BatchNorm2d, ReLU and Conv2d in Pytorch. (See the respective papers for the detail design). The blue parts (including Max-pooling operations) were called as downsampling pass, the dot lines (including operations marked on them) were called as concatenation part, specially, the orange parts (including Avg-pooling operations) were called as upsampling pass, and the last green Conv Unit was named as Tasker, downsampling, upsampling together with concatenation part were named as Extractor. Input images are progressively encoded and decoded to get the ultimate segmentation map. Style embeddings of each scale is obtained by using global average pooling on respective convolutional map.

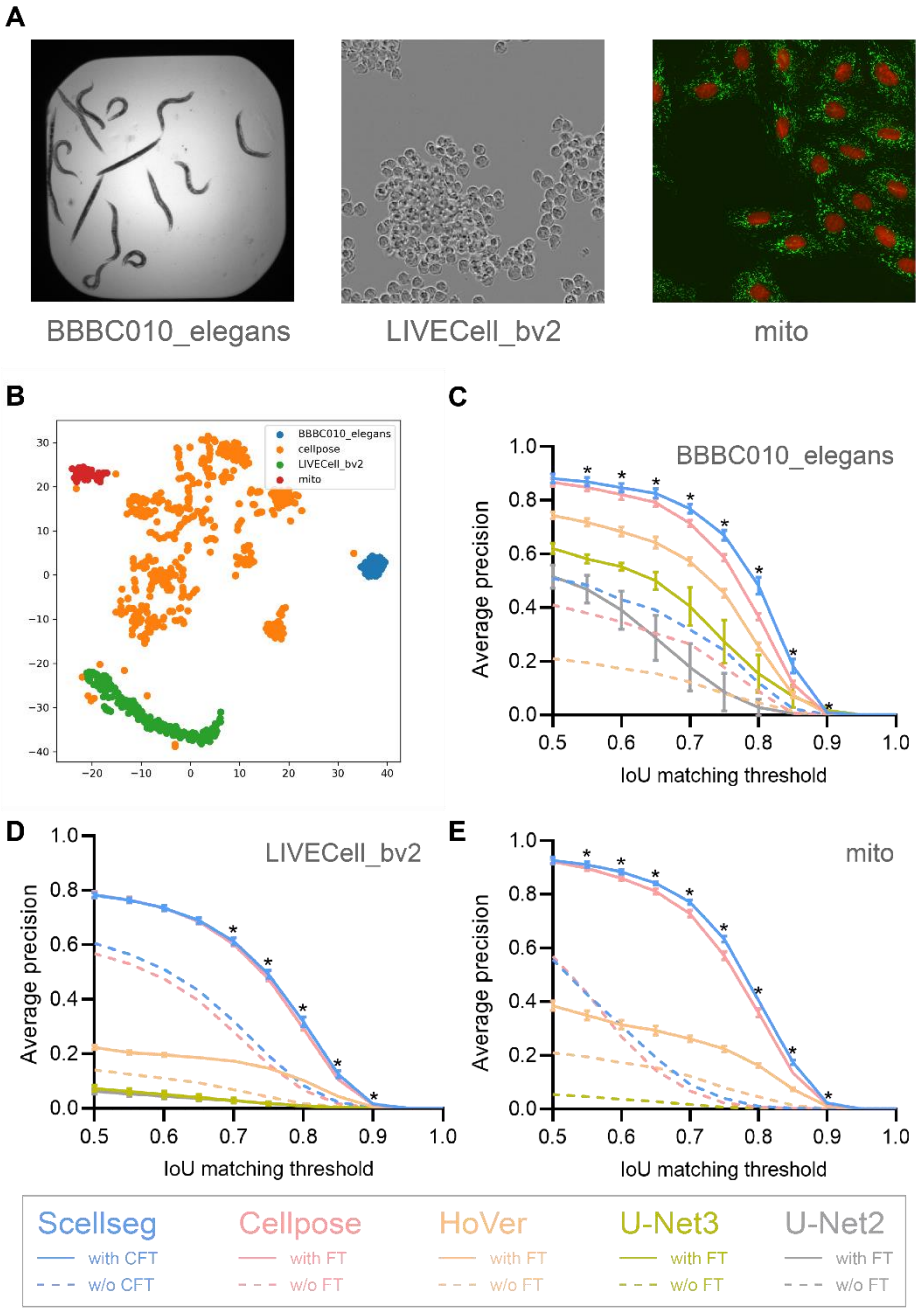

Fig. 3. Transferability of Scellseg with contrastive fine-tuning strategy on three evaluation datasets. (**A**), Example images of three datasets. (**B**), Visualization for style embeddings of three datasets and pre-trained dataset using t-SNE. (**C**-**E**), Performance of different models on BBBC010_elegans (**C**), LIVECell_bv2 (**D**) and mito (**E**) dataset. Different colors correspond to different models, the dotted lines denote the performance of applying models directly and the solid lines donate the performance after fine-tuning. For Scellseg, we use contrastive fine-tuning (CFT) strategy and for others is classic fine-tuning strategy (FT). We did not plot the line which corresponding performance is less than 0.01. Each pre-trained and fine-tuning pipeline was conducted 10 times at various

random states, error bars represent the mean ± SD. * donates P-value<0.05, determined by two-way ANOVA followed by Sidak's multiple comparisons test for Scellseg with CFT and Cellpose with FT.

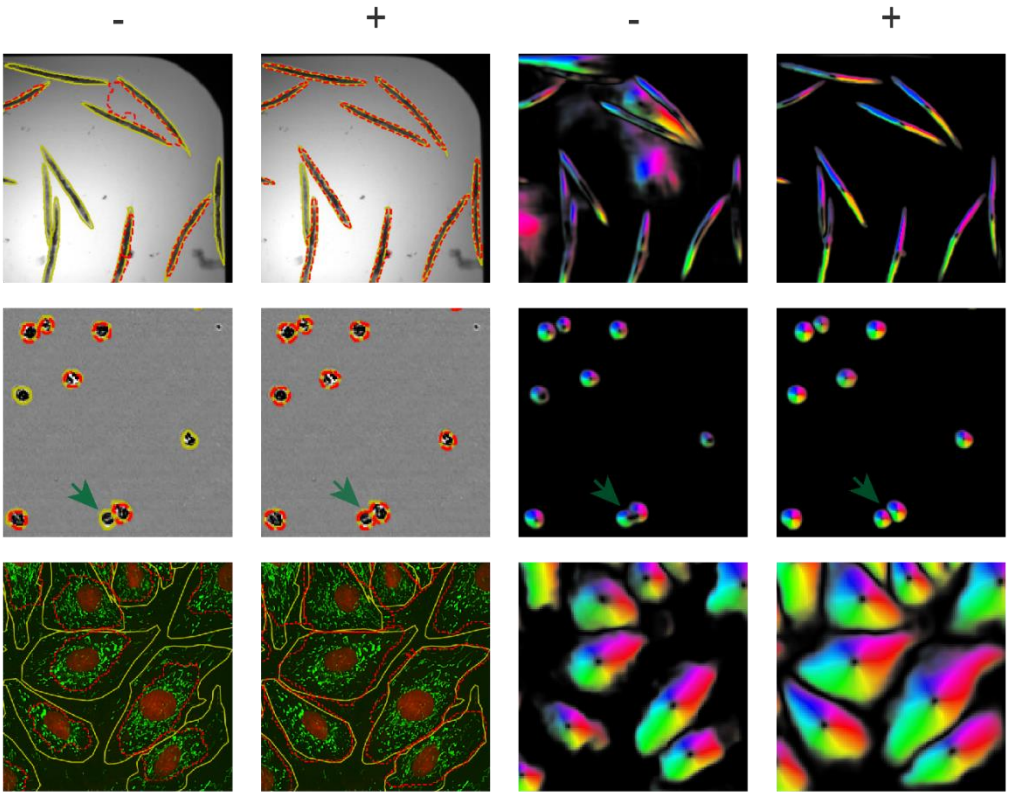

Fig. 4. Example Scellseg segmentation results before and after contrastive fine-tuning on three evaluation datasets. The right two columns show the direct topological maps outputted by Scellseg model. The left two columns show ultimate masks of different datasets, the ground truth masks are shown in yellow solid line, and the predicted masks are shown in dotted red line. Symbol “-” donate results of applying models directly and symbol “+” donate results after fine-tuning. Green arrow emphasizes the segmentation of adherent cells.

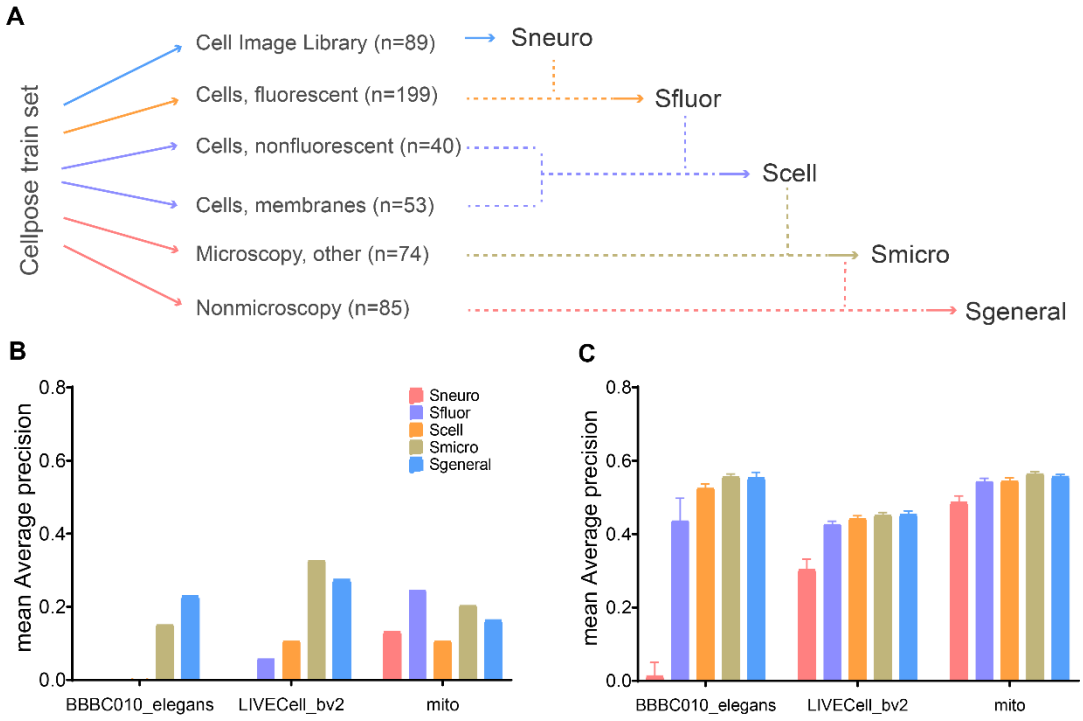

Fig. 5. Pre-trained dataset scale experiments. (**A**), Composition of different subsets from Cellpose train set. (**B**), Generalization ability of different pre-trained Scellseg models on three evaluation datasets. Generalization ability donates the performance of pre-trained model employed directly. (**C**), Transferability of different pre-trained Scellseg models on three evaluation datasets. Transferability donates the performance of pre-trained model after fine-tuning. Each pre-trained and fine-tuning pipeline was conducted 10 times at various random states, error bars represent the mean ± SD.

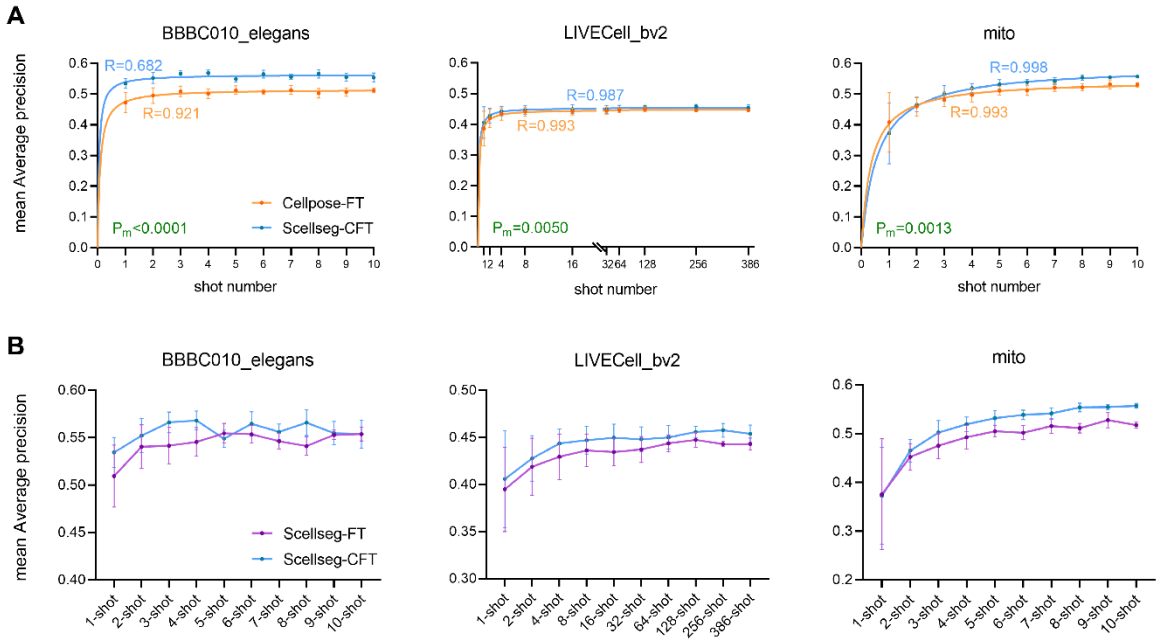

Fig. 6. Shot data scale experiments (A) and ablation experiments (B). Each pre-trained and fine-tuning pipeline was conducted 10 times at various random states, error bars represent the mean ± SD. (**A**), Performance of Cellpose-FT and Scellseg-CFT on three evaluation datasets at 10 shot data scales. Cellpose-FT donates Cellpose with classic fine-tuning strategy and Scellseg-CFT donates Scellseg with contrastive fine-tuning strategy. Another metric to evaluate the models is shown in Fig. S3. We performed nonlinear regression based on hyperbola function and corresponding R-value of fitted curve is plotted in the picture. A two-way ANOVA analysis was conducted for group comparison of Scellseg-CFT and Cellpose-FT per dataset and corresponding P_m-value is plotted in the picture. P_m-value<0.05 means the performance between Scellseg-CFT and Cellpose-FT is significant. (**B**), Ablation experiments for contrastive fine-tuning strategy. Scellseg-FT donates Scellseg with classic fine-tuning strategy, data of Scellseg-CFT is directly copied from (**A**).

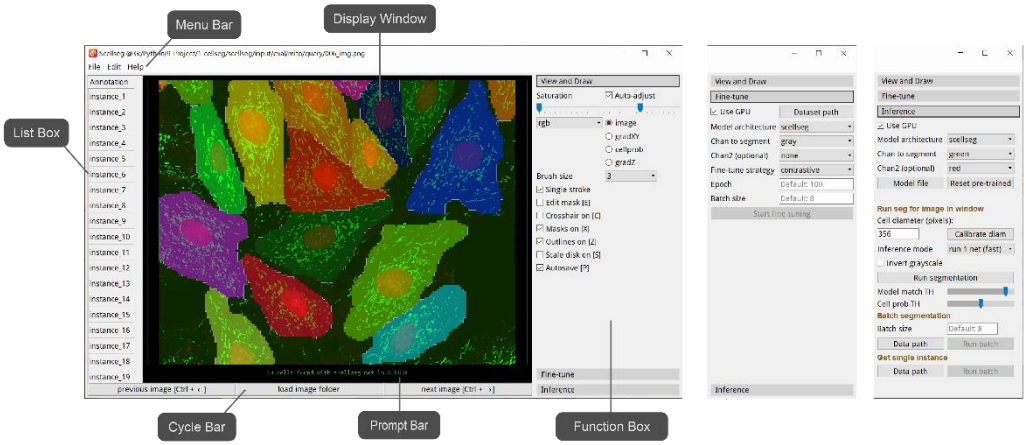

Fig. 7. Graphical user Interface (GUI). This GUI contains six modules: Menu Bar, List Box, Display Window, Prompt Bar, Cycle Bar and Function Box. There are three main functions: “View and draw”, “Fine-tuning” and “Inference”.

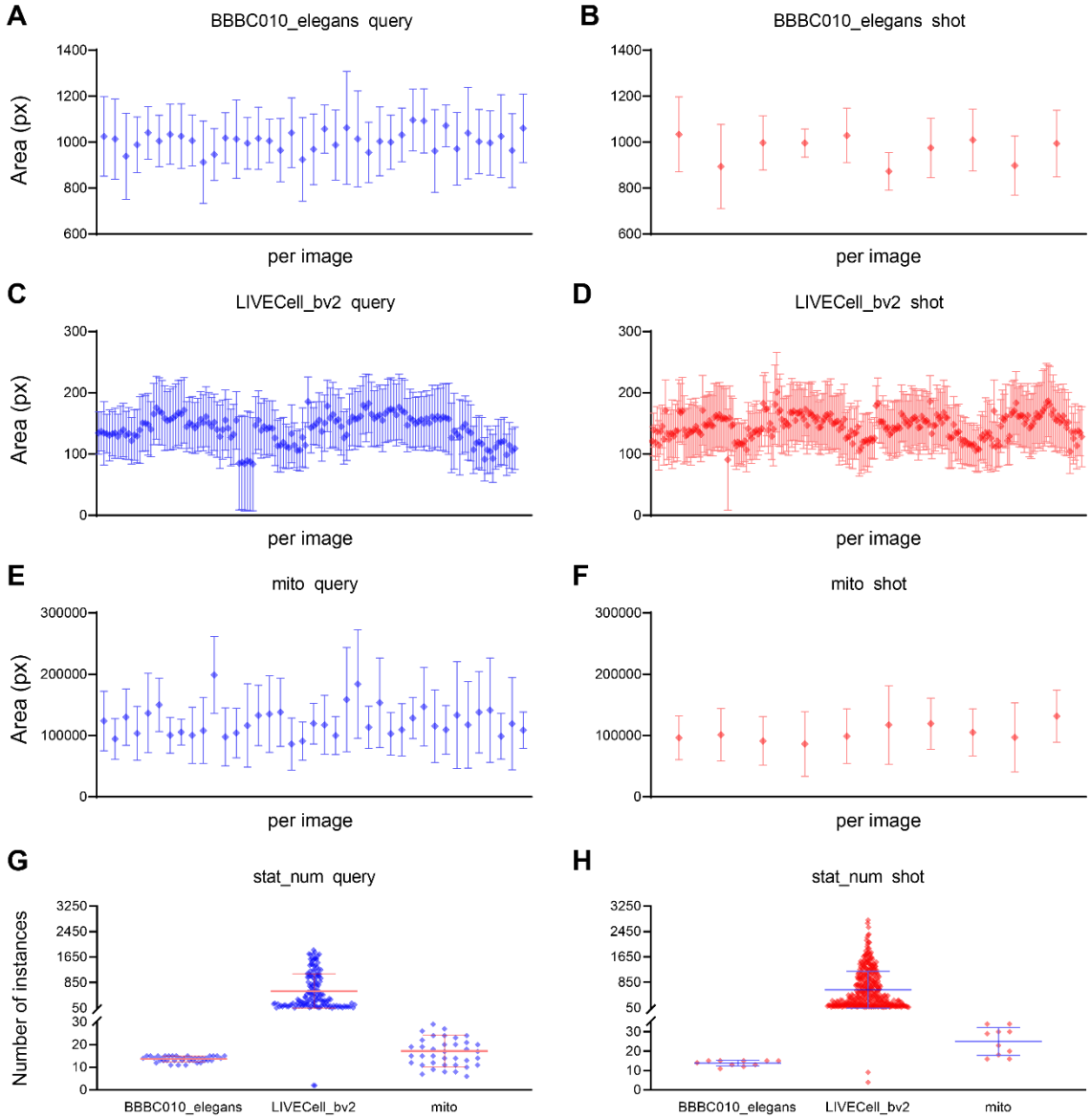

Fig. S1. Statistics of three evaluation datasets. (**A-F**), Distribution of cell areas in px for each image in query or shot data of BBBC010_elegans (**A, B**), LIVECell_bv2 (**C, D**), mito (**E, F**) dataset, error bars represent the mean ± SD. (**G-H**), Distribution of number of instances for each image in query data (**G**) or shot data (**H**) of three datasets, each dot represents one image in corresponding dataset, error bars represent the mean ± SD.

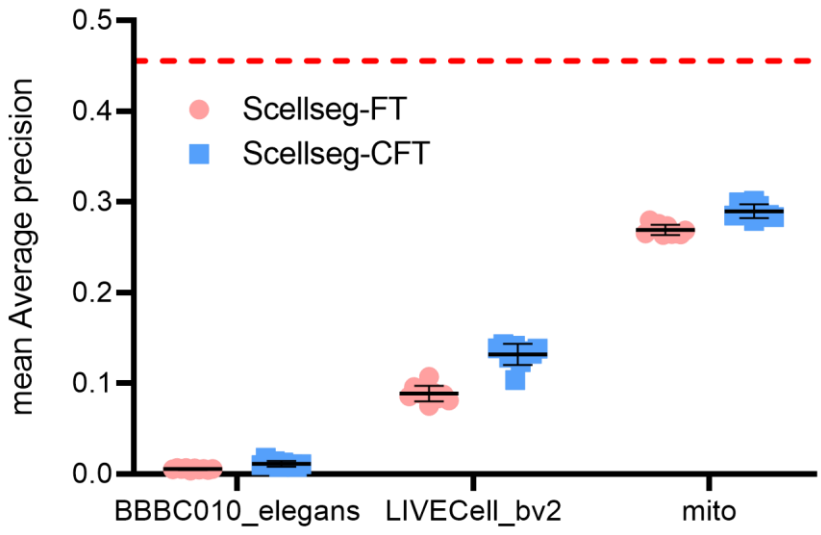

Fig. S2. Generalization ability on Cellpose test set of Scellseg models. Scellseg-FT donates Scellseg with classic fine-tuning strategy and Scellseg-CFT donates Scellseg with contrastive fine-tuning strategy. Red dot line donates employing Scellseg directly on test set. Each “pre-trained + finetuning” pipeline was conducted 10 times at various random states, error bars represent the mean ± SD.

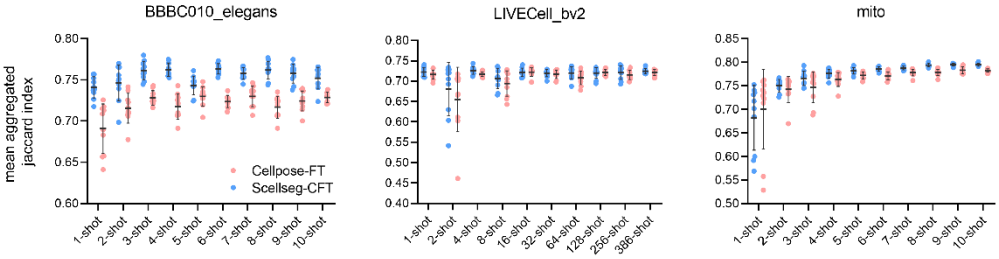

Fig. S3. Using mean Aggregated Jaccard Index metric to evaluate performance in shot data scale experiments. Error bars represent the mean ± SD.

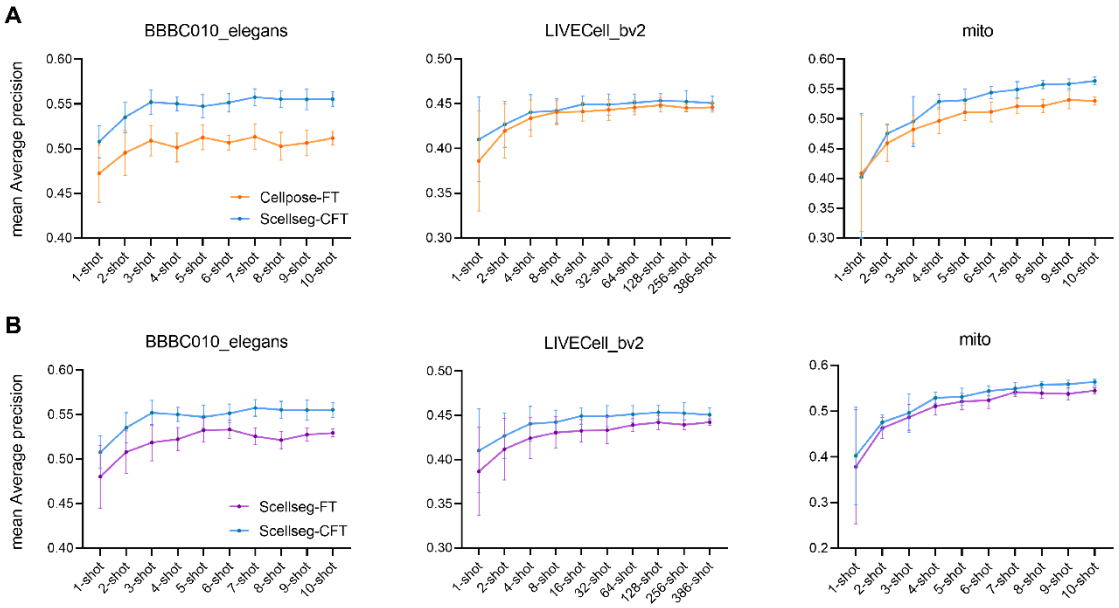

Fig. S4. Shot data scale experiments (A) and ablation experiments (B) on Scellseg pre-trained with Smicro. Each pre-trained and fine-tuning pipeline was conducted 10 times at various random states, error bars represent the mean ± SD.

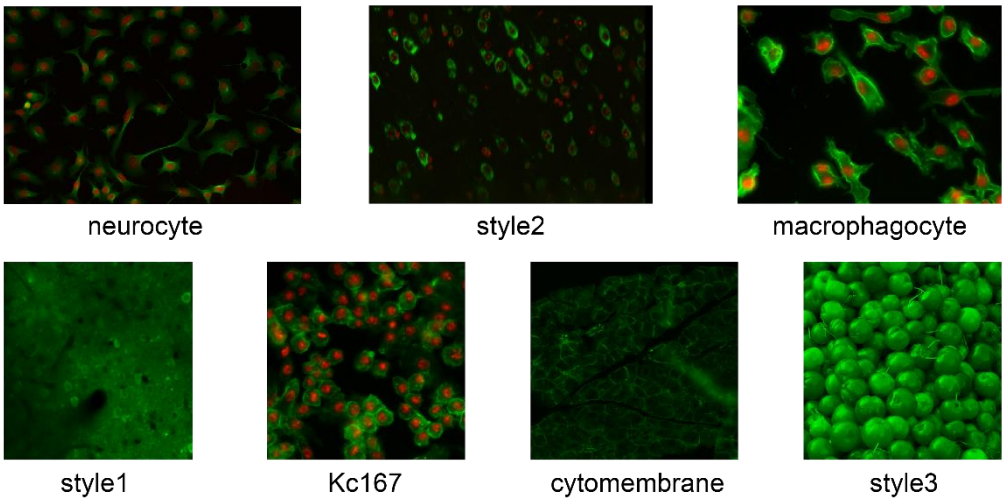

Fig. S5. Example image per style of Contrast data. There are totally 7 styles of images we used in our contrastive fine-tuning strategy, each style includes five images.

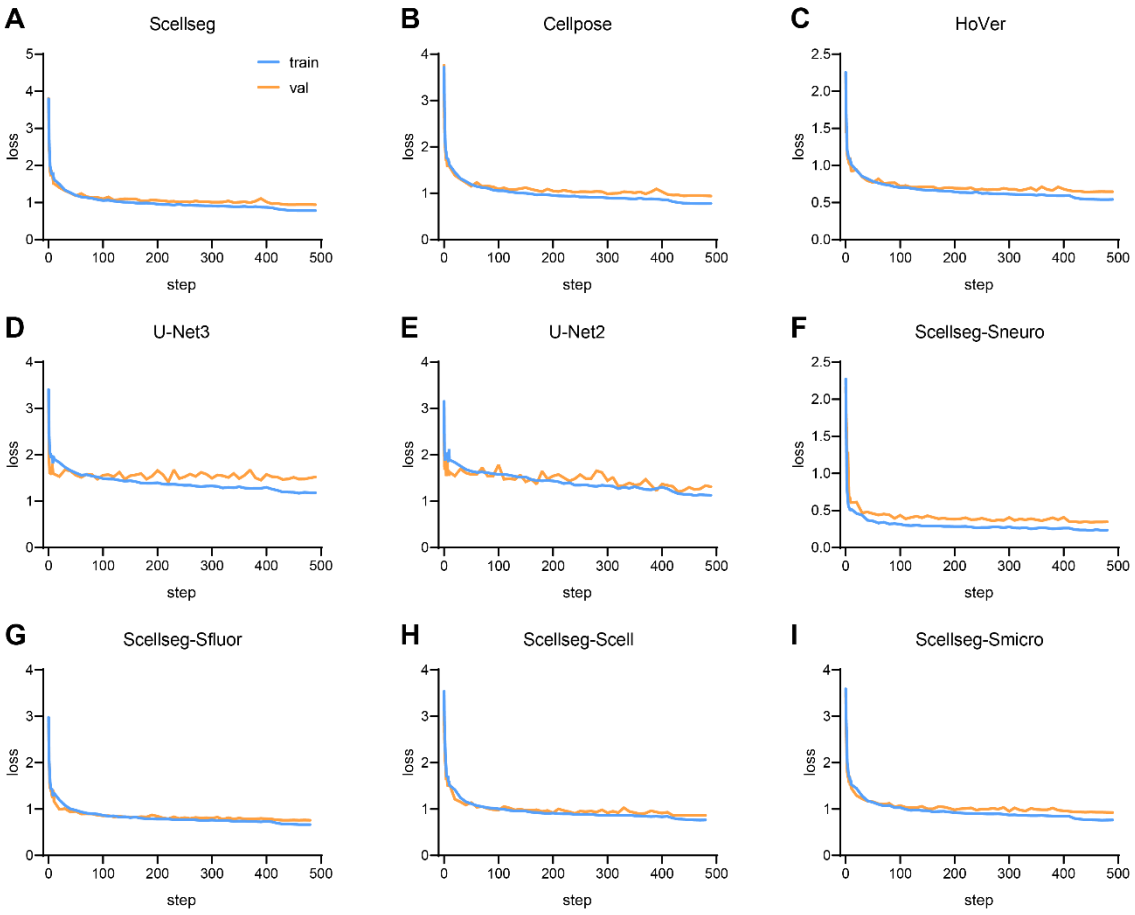

Fig. S6. Train logs of models in the paper.

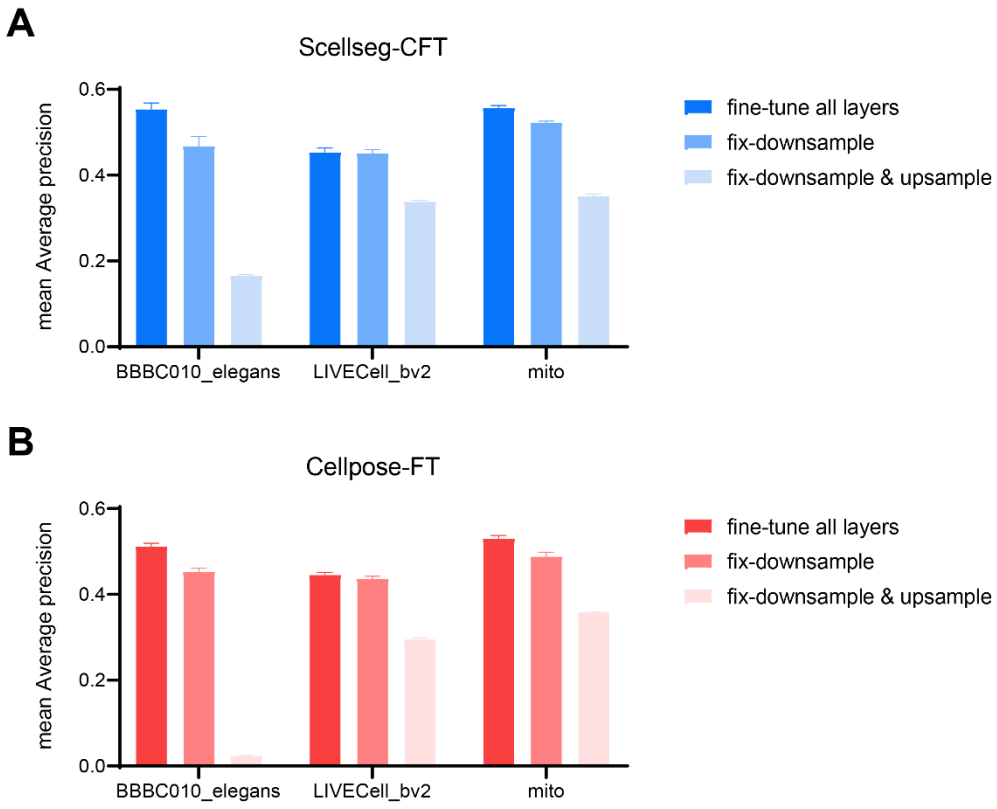

Fig. S7. Performance of different fine-tuning methods. We compared three kinds of fine-tuning methods for Scellseg-CFT (**A**) and Cellpose-FT (**B**) on three evaluation datasets, respectively are fine-tuning all layers of the model, fixing the downsampling layers and fixing downsampling-upsampling layers. Each fine-tuning method was conducted 10 times at various random states, error bars represent the mean ± SD.

Dataset	Number of images in train set	Number of images in test set	Number of instances in test set	Acquisition Modality	Shape of images	Characteristics
BBBC010_elegans	10	39	670	Bright Field	696*520*1	worms appear rod-like or curved in shape
LIVECell_bv2	386	152	533	Fluorescent, 10X	704*520*1	small spherical morphology, homogeneous population
mito	10	39	89821	Phase-contrast imaging, 60X	2048*2048*3	mitochondria scatter around the nuclei, no clear boundaries
Total	**406**	**230**	**91024**			

Table. S1. Summary statistics of three evaluation datasets.