The Journal of Systems and Software 211 (2024) 111986

Contents lists available at ScienceDirect

The Journal of Systems & Software

journal homepage: www.elsevier.com/locate/jss

A data-driven active learning approach to reusing ML solutions in scientific applications☆

Hamideh Hajiabadi [a,*], Christopher Gerking [c], Lennart Hilbert [b], Anne Koziolek [c]

[a] Helmholtz Information and Data Science School for Health (HIDSS4Health) Karlsruhe Institute of Technology, Karlsruhe, Germany
[b] Zoological Institute and Institute of Biological and Chemical Systems (IBCS), Karlsruhe Institute of Technology, Karlsruhe, Germany
[c] Institute of Information Security and Dependability (KASTEL), Karlsruhe Institute of Technology, Karlsruhe, Germany

ARTICLE INFO

Keywords:
Reuse of machine learning solutions
Data-driven approaches
Interactive microscopy image segmentation

ABSTRACT

Artificial intelligence can revolutionize scientific projects, but scientists face challenges in reusing, integrating, and deploying cost-effective and high-quality machine learning solutions. Determining suitable algorithms and parameters is difficult, especially for non-programmer scientists. Some algorithms, like deep learning-based methods, offer flexibility but require extensive training on annotated data. This poses a hurdle in labor-intensive tasks like biological image segmentation that relies on expert annotations.

In this paper, we present a data-driven framework designed to assist scientists in selecting, reusing, and training machine learning solutions for microscopy image segmentation. The framework is based on establishing a mapping between object morphology features and the optimal segmentation algorithms and settings for individual objects. This mapping is iteratively refined through a combination of unsupervised learning and active learning iterations. To expedite convergence, objects are initially clustered based on their morphology. In each active learning iteration, the most informative and uncertain samples are selected and queried within a specific cluster. Through a biological case study, we demonstrate that our method enables the selection and training of segmentation algorithms specific to object types. Additionally, the selective requests for user input significantly reduce the number of user interactions required for this task.

1. Introduction

For scientists seeking to integrate machine learning (ML) into their research, a key challenge lies in identifying the most suitable ML algorithm and associated (hyper-)parameters for their specific research problem. It is crucial to recognize that there is often no universally superior ML algorithm or set of algorithm settings. The optimal solution typically hinges on an empirical examination of the data. Moreover, many scientists face limitations in ML and programming expertise, compounding the difficulty of this task. A diverse array of ML solutions is available, with some, like deep learning (DL), excelling at fully supervised learning from raw data, handling diverse data variations through learning from large annotated datasets (Alemi Koohbanani et al., 2020). However, the costly process of data annotation, reliant on domain expertise, hinders widespread DL adoption. In contrast, traditional ML solutions, relying on feature engineering without annotated data, work with smaller datasets. Yet, configuring parameters for one data variation may compromise performance for others (Jakhar and Kaur, 2020).

One domain of science, in which exploration of datasets by use of ML can aid expert scientists, is object analysis in biological imaging data. Here, object segmentation from images is an important data analysis step. The segmentation process involves separating the image into identifiable regions or objects of interest. The literature already offers various approaches, including traditional ML or DL-based methods. In many cases of bioimaging data, annotations are costly or even unattainable, so the application of traditional ML algorithms is more suitable for projects especially when feature engineering does not require complex procedures or extensive domain knowledge. However, a one-size-fits-all setting may not be applicable for all object types, making effective approaches for parameter adjustment a valuable objective.

One approach to train a DL method on a smaller set of annotations is called "weakly supervised" training. This approach enables expert scientists to interactively provide annotations for a subset of the dataset, which are then used for model training. Subsequently, the trained model can be applied to the remaining data. For instance,

☆ Editor: Prof. Raffaela Mirandola.

* Corresponding author.
E-mail addresses: hamideh.hajiabadi@kit.edu (H. Hajiabadi), christopher.gerking@kit.edu (C. Gerking), lennart.hilbert@kit.edu (L. Hilbert), koziolek@kit.edu (A. Koziolek).

https://doi.org/10.1016/j.jss.2024.111986
Received 3 March 2023; Received in revised form 8 January 2024; Accepted 26 January 2024
Available online 30 January 2024

WEKA (Arganda-Carreras et al., 2017), Ilastik (Berg et al., 2019), and nuClick (Alemi Koohbanani et al., 2020) are interactive, weakly supervised ML tools used for image pixel classification. They involve annotation of sample objects via a graphical user interface to facilitate training. Users of these tools can iteratively provide feedback, correcting or adding labels until the model demonstrates satisfactory performance for application to the entire dataset. Compared to fully supervised DL segmentation tools, this category of tools reduces the required training data size. However, these weakly supervised tools still require expert scientists to give annotations based on visual impressions and intuition to identify informative objects for training.

Here, we introduce a novel, data-driven framework that reduces the need for user interaction even further by leveraging active learning. This framework not only optimizes the training processes but also assists in the selection of an appropriate segmentation algorithm. By reducing the effort and required domain expertise to customize segmentation algorithms to different datasets, our framework combines the ease of use provided by conventional methods with enhanced handling of objects with high variability. The framework comprises an unsupervised initial step to choose a segmentation algorithm and object-specific settings, followed by further parameter adjustment based on active learning. Our primary focus is on segmenting microscopy images, but our framework is adaptable and can, in principle, be applied also to other application cases where annotation by domain experts is difficult or costly. Moreover, the proposed feature engineering method does not necessitate complicated algorithms or extensive knowledge of the domain. The contributions of our paper are summarized as follows:

- We propose a framework for image segmentation tasks that enables biologists to use the full potential of existing segmentation algorithms. This framework can, in principle, be used in every scientific use case that shares similar characteristics.
- We propose the use of an unsupervised metric that assists in the initial choice of the segmentation algorithm as well as its initial settings.
- We assign every segmented object to its respective segmentation parameters, creating an object-to-parameter map that is updated throughout the active learning process.
- We conduct a biological case study, in which scientists using the framework obtain satisfying results with efficacy and efficiency in a few interactions.

Our ongoing work extends our prior investigation into weakly-supervised machine learning with interactive user input (Hajiabadi et al., 2022). A detailed list of improvements beyond our previous work can be found in the concluding part of Section 2.

The rest of the paper is structured as follows. The scientific use case is explained in detail in Section 2. A brief summary of the segmentation metrics is provided in Section 3, followed by a demonstration of our proposed framework in the next section (Section 4). We introduce research questions as well as the evaluation and experiment in Section 5. We then discuss our results in Section 5.1 and threats to validity in Section 6. Our conclusions are given in Section 8.

2. Scientific use case

The analysis of objects in biomedical image data usually involves quantifying multiple properties, for instance, object shape, object location, or the intensity of an experimentally applied label. In many cases, these measurements are based on the segmentation of objects, meaning the assignment of image pixels (or voxels) as “object” (1) or “not-object” (0) (Wu et al., 2000; Salem et al., 2016). One popular way to segment microscopy images is through conventional segmentation methods (Dougherty, 2018; Li et al., 2008) such as thresholding using algorithms to determine a threshold intensity based on the intensity distribution within the image. One other category of existing segmentation techniques for microscopy images, known as weakly supervised,

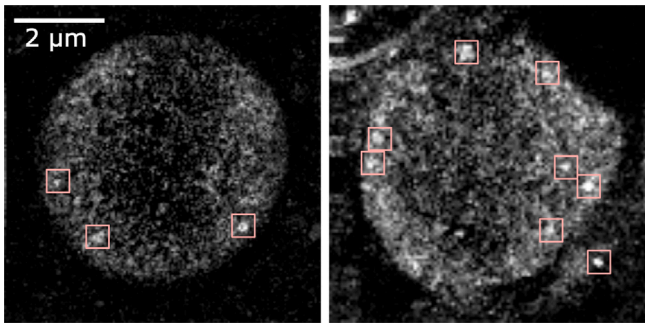

Fig. 1. Different object variations. Images are the nuclear mid-plane of a fixed zebrafish embryo at the sphere stage, which are recorded by STEDD super-resolution microscopy. Intensity distributions of RNA Polymerase II (Pol II) Serine 5 phosphorylation (Ser5P) were obtained by STEDD microscopy, while Pol II Ser2P intensity distributions are obtained by regular confocal microscopy from the same plane. Pol II Ser5P clusters are marked. Intensity scale from black to white adjusted to the 0.01th and the 99.99th percentile (Pancholi et al., 2021).

is based on learning from a partially-annotated dataset. The annotation is mostly provided by the biologists through user-friendly interactive pipelines (Sommer et al., 2011; Berg et al., 2019; McQuin et al., 2018; Carpenter et al., 2006). The most recent existing algorithms use DL to learn from raw data, but they require large annotated datasets (Apthorpe et al., 2016; Guerrero-Pena et al., 2018; Chen et al., 2020; Al-Kofahi et al., 2018; Yi et al., 2020).

For biologists, it can be a daunting task to select the appropriate segmentation method and tailor it to the specific data. Although conventional segmentation methods (which are mostly unsupervised) perform well in high-quality microscopy images of objects with easily distinguishable boundaries (Ivashkevich et al., 2011; Cantaloube et al., 2012), these methods often fall short when analyzing objects with ambiguous boundaries and varying attributes such as size, intensity, and morphology (Fig. 1) (Matula et al., 2010; Osterwald et al., 2015). In contrast, fully supervised DL-based segmentation algorithms are more flexible in handling data variations as they can adapt to the input and annotated output. However, creating annotated training datasets for these algorithms is a costly and time-consuming process that necessitates the knowledge of experts from the biological domain. There are also weakly supervised segmentation tools, trained on limited annotations, which enable biologists to provide interactive annotations for model training. Examples include WEKA (Arganda-Carreras et al., 2017) for image pixel classification, Ilastik (Berg et al., 2019) for cell segmentation, classification, tracking, and counting, as well as NuClick (Alemi Koohbanani et al., 2020) for microscopy image segmentation. While these tools reduce training data requirements, they still heavily depend on annotations provided by experts and may not accurately segment objects due to reliance on visual impressions and intuition.

This paper presents a pragmatic framework for optimizing the process of identifying and customizing ML segmentation solutions on a per-object basis. To this end, the framework initially explores a range of unsupervised conventional segmentation methods while accommodating object diversity. Subsequently, through iterations of active learning facilitated by interactions with an expert user, we propose the optimization of (hyper-)parameter configurations tailored to each individual object. This methodology is particularly beneficial in bioimaging, where manual annotation poses inherent complexities. However, it is important to minimize the interaction time to encourage the adoption of this framework in bioimage analysis pipelines. At the core of this framework lies a data-driven regressor that maps object-morphology features to the specific segmentation algorithm and settings employed in the process. This mapping is updated during each active learning iteration based on user feedback. To expedite the convergence rate,

the most informative and uncertain samples are selected from various regions of the feature space for user input queries. To ensure diversity in the selected samples, an initial clustering function divides the feature space into subparts.

This work is an extended version of our previous research, in which we proposed a framework for weakly-supervised ML with interactive user input (Hajiabadi et al., 2022). Below is a list of improvements made between the current work and the previous version:

- In our previous work, we updated the settings of all objects in one cluster similarly during each user interaction. However, in this updated version, we have introduced a mathematical formulation that describes the mapping between each object and its associated segmentation setting. This formulation is explained in detail in Section 4.7 of the paper. As a result of this formulation, with each user input, all settings are updated through re-training the regressor. This process involves mapping each data sample to its associated setting, allowing for dynamic adjustments and refinement of the segmentation algorithm.
- Furthermore, we have improved the convergence rate by automatically determining the number of clusters and implementing three query strategies. These enhancements have led to a reduction in user interactions, as demonstrated in Section 4.5 and Section 4.6 of this paper.
- Additionally, we demonstrate the convergence of the framework through an additional experiment, as illustrated in Section 5.1 of this paper.

3. Background: Segmentation metrics

Metrics to evaluate image segmentation can be categorized based on different criteria. One criterion is whether a metric is considered subjective or objective (Chen et al., 2018). Subjective metrics are based on human visual assessments. These metrics are, by definition, not formally defined. To reach a generalizable result, the number of observers should be high.

Another criterion is whether the evaluation of a segmentation approach is analytical or empirical (Zhang, 1996). In empirical evaluations, the accuracy and precision are evaluated on the basis of example data in a supervised fashion. An analytical evaluation does not rely on example data but rather analyzes the theoretical properties of the segmentation (Wang et al., 2020).

As a third criterion, the evaluation of image segmentation can be supervised (with annotated images) or unsupervised (without annotated images). Supervised evaluation benefits from the direct comparison between the segmentation results and the reference annotated image. The evaluation can be based on the per-pixel comparison (True Positive (TP) and True Negative (TN), Matthews Correlation Coefficient (MCC) and F-measure,

region-based comparison (Bipartite Graph Matching (BGM) (Taha and Hanbury, 2015) and Segmentation Covering (SC) (Dey et al., 2018)) or distance-based evaluation (Hausdorff Distance, directional Hamming distance and Mahalanobis distance (Pont-Tuset and Marques, 2015)). Unsupervised metrics evaluate the quality of segmentation by directly calculating the feature parameters of the segmentation result without using the annotated reference image. These metrics are the ideal kinds of metrics, especially when there is no ground truth. These unsupervised metrics are mostly based on mathematical indicators showing the quality of segmentation results, such as Peak signal to noise ratio (SNR) (Zhang et al., 2008), calculating the inside and outside contrast of the segmentation area, or measuring the foreground and background variances (Zhang et al., 2008).

4. The proposed approach

In this section, we propose our framework (Algorithm 1) for data-driven image segmentation based on active learning. Fig. 2 illustrates the overall view with all steps included in the framework. As can be seen from the figure, the initial step ❶ is to prepare the object set. Preparation means that the object positions are spotted and a bounding box around the rigion of interest (ROI) is extracted (Section 4.1). As the objects are in various shapes, the segmentation settings applied to various objects are expected to be different. That is why the framework creates a feature space in step ❷ (Section 4.2), thereby being able to differentiate variation in object shapes.

The assumption is that there is no annotation, which is why we cannot certainly select the best algorithm and the setting resulting in the optimal prediction for each object. But through unsupervised iterations in step ❸ (Section 4.3), the framework obtains the initial estimate of the segmentation algorithm and the setting for each object. However, this function is only able to *approximate* the quality. Thus, to accurately adjust the segmentation setting associated with each object, active learning iterations afterward proceed in step ❺ (Section 4.5).

In active learning, the learning process proceeds by actively querying the users to provide feedback on the presented data. To decide on which objects should be selected for querying users, the framework combines three query strategies, which are (i) querying from diverse clusters, (ii) using the most informative objects, and (iii) using the most uncertain objects. Whereas the latter two strategies are both implemented in step ❺ (Section 4.5), the cluster determination takes place in step ❻ (Section 4.6). After presenting the selected objects to the user and receiving the user feedback, the segmentation settings associated with the presented objects are directly updated in step ❼ (Section 4.7).

To enable automatic adjustment of the associated settings with all other objects (objects that are not queried) according to the user feedback, a mathematical formulation expressing the mapping between the objects and the associated settings is required. To create this mapping function, the framework fits a regression model on the objects and the associated settings. The input of the regression model is the representation of the object set in feature space and the settings associated with objects is the output. In every active learning iteration, after updating the settings of the presented objects and their neighbors, the regressor model is re-trained and all the associated settings are accordingly updated (Section 4.7). The active learning iterations continue until convergence.

Algorithm 1 The proposed segmentation framework

INPUT:
Input images
Base unsupervised segmentation algorithms
OUTPUT:
Optimal setting assigned to each object

1: Preparing object dataset according to Section 4.1
2: Initial segmentation and finding the initial estimate of the settings per object (Section 4.3)
3: Creating the feature space according to Section 4.2
4: Finding different clusters according to Section 4.6
5: Fitting a regression model on the objects and the segmentation settings associated with the objects (Section 4.7)
6: **while** repeat until convergence not reached **do**
7: for every cluster, select two objects (the most informative and the most uncertain Section 4.5
8: Present the selected objects to the user according to Section 4.5
9: According to the user feedback, directly adjust the segmentation settings for presented objects (Section 4.7)
10: Re-train the regressor to update the associated settings for all other objects (Section 4.7)
11: **end while**
return the associated settings

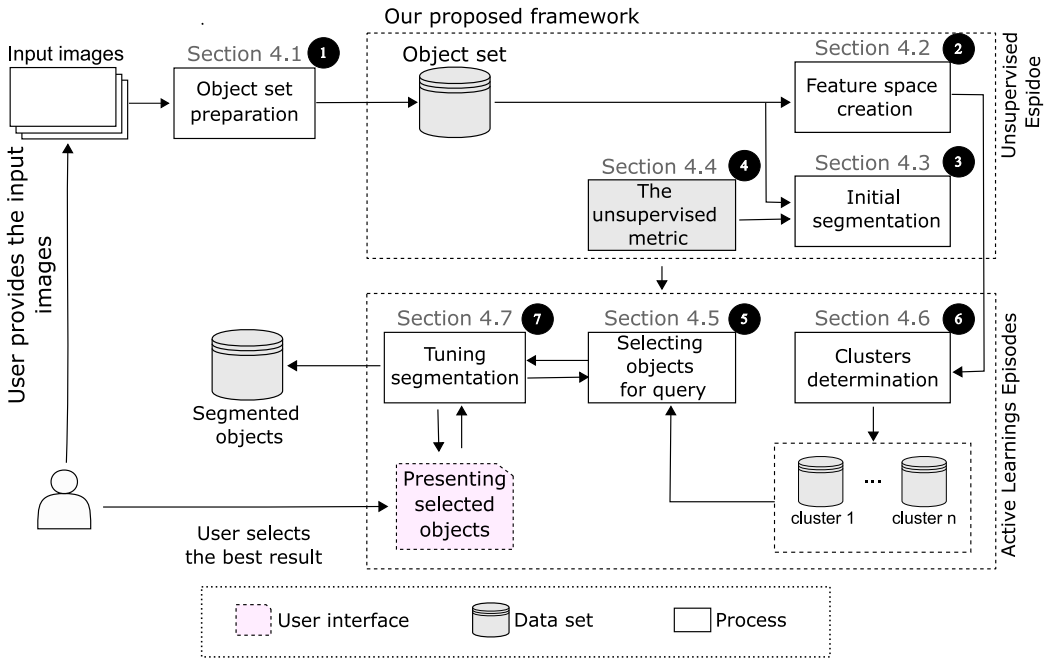

Fig. 2. The proposed framework consists of several automated steps. At the beginning, the user uploads the input raw images. With a few interactions, the segmented objects are given back to the user.

4.1. Object set preparation

In this step (step 1 in Fig. 2), the framework locates segmentation target objects (ROI) and extracts a bounding box surrounding these objects. Some of the ROIs are marked with red boxes in Fig. 1. Some simple thresholding algorithms can successfully detect the position of the target objects, however, depending on the experiments there might be some other bright subcellular components making directly spotting the ROI difficult (Fig. 1). Therefore, the framework at first employs a filter to make the image blurry and then proceeds with a thresholding technique to only spot the ROIs. Generally, any blurry filter can be used, however, in our particular application case, a Gaussian blur filter with a standard deviation of six pixels is applied to each nucleus, followed by a local thresholding method to roughly spot the ROIs. A region of size (60×60) is then cropped and saved. Some examples of the objects are illustrated in Fig. 3.

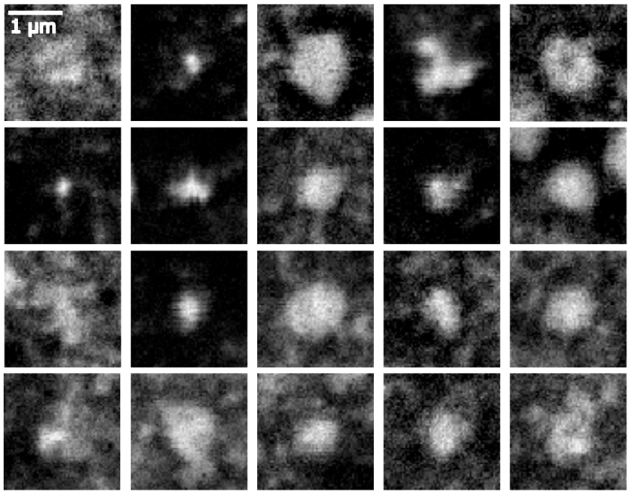

Fig. 3. Detail views of objects. Representative Pol II Ser5P clusters mid-plane obtained by STEDD super-resolution microscopy from a fixed sphere-stage zebrafish embryo. Each image is of size 60 × 60 pixels. The images are randomly selected from a dataset of size 148.

4.2. Feature space representation of objects

Every measurable (mostly numeric) property describing a characteristic of an object is generally called a feature (Sklansky, 1978). Each segmentation target object can be represented by a vector of features and the feature space is the space spanned by the feature vectors (Sklansky, 1978). Objects usually show up with different morphologies (Pancholi et al., 2021). For example in our particular application case, some objects appear similar to a dot, and some other objects are larger and relatively compact but still with clear boundaries. There are also some larger objects with unfolded shapes where there is no clear boundary with extensive morphological variety (Fig. 3). Because of these differences, we expect that different segmentation algorithms and parameter settings will be optimal for the different types of objects.

To enable the choice of different algorithms for different types of objects, the framework first establishes a representation of the overall object sets into a feature space capable of reflecting morphology variation (step 2 in Fig. 2). We consider two groups of features, some are extracted from the intensity histogram of the image and some are shape-related features.

- Intensity histogram-based features: the intensity histogram provides valuable information about the distribution and variation of pixel intensities in an image. These features can be useful for tasks such as object detection. As variations in intensity histogram peaks or distributions can indicate the presence of specific structures or objects of interest in microscopy images.
- Shape-related features: in microscopy images, the shape and morphology of objects can be crucial for analysis and interpretation. These features can provide insights into the morphology of objects such as size, area, perimeter, and compactness. They enable quantification and comparison of shapes, facilitating studies on morphology variations.

The features space that we consider includes the following features:

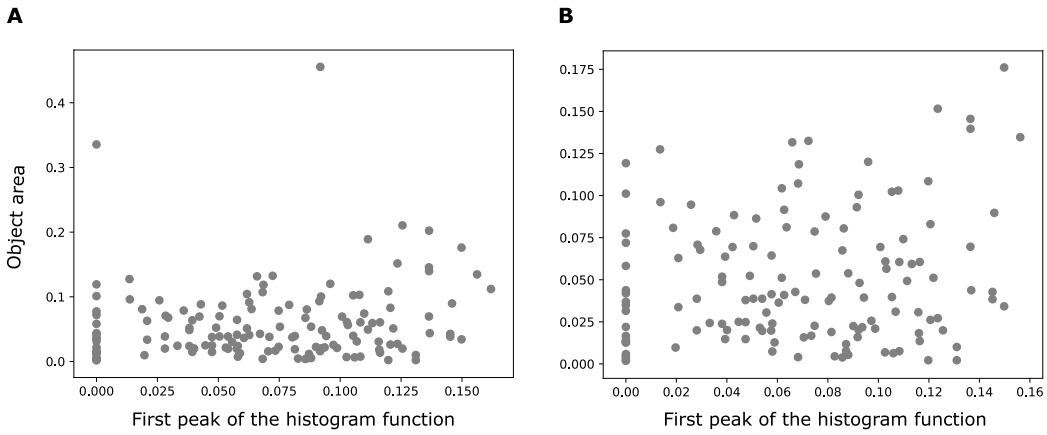

Fig. 4. (A) Two features (area and the first local maximum of the histogram intensity function) are selected and 2D representation of the objects based on the selected features is illustrated. (B) 2D representation of the object set in the feature space after outlier removal.

Table 1
Threshold-based segmentation algorithms used in our framework. Thresholding techniques aim to create a binary mask from a grayscale image, which segments objects from a background.

Algorithm	Short description
Isodata	A threshold is iteratively found based on the image histogram
Mean	Uses the mean value of pixel intensities as threshold value
Minimum	Takes a histogram of the image and smooths it repeatedly until there are only two local maxima (Prewitt and Mendelsohn, 1966).
Otsu	Based on maximizing the variance between two classes of pixels which are separated by the threshold (Otsu, 1979).
Triangle	A geometric method assumes maxima near one end of the histogram and searches towards the other end (Zack et al., 1977).
Yen	Based on Yen's thresholding method (Sezgin and Sankur, 2004)
Li	An iterative method based on Li's Minimum Cross-Entropy thresholding method (Li and Lee, 1993)
Local	Each threshold value for each pixel is the weighted mean of the local neighborhood minus an offset value.
Median	Uses the median value of pixel intensities as threshold value

- Local maxima and their values in the histogram function of the object.
- Local minima and their values in the histogram function of the object
- Area of the object, which means the total space taken up by the 2D shape of an object.
- Solidity of the object. A solidity value of 1 signifies a solid object and a value less than 1 represents an object having an irregular boundary, or containing holes.

After creating the feature space, all the features in each dimension are normalized (value between 0 and 1) then our model training is less sensitive to the scale of features (Fig. 4A). The outliers are next removed based on the *Z-score* (Fig. 4B). *Z-score* of 1 means that the data point distance to the mean is equal to 1 variance. Usually for data that follows a Gaussian distribution a Z-score ≥ 3 means that the data point is an outlier. Above the Z-score of 3 we consider a data point to be an outlier Fig. 4 provides a 2D representation of the objects in the feature space before outlier removal and after.

4.3. On the initial choice of segmentation algorithm

For the initial segmentation (step 3 in Fig. 2), the framework carries out an unsupervised iteration by comparing the results of several segmentation algorithms per object (Table 1). In principle, any segmentation algorithm can be used here. For simplicity, in our application case, the framework only considers segmentation methods based on thresholding techniques (Sahoo et al., 1988). As there is not any annotated object enabling accurate evaluation of the resulting quality, we cannot accurately decide which segmentation algorithm and settings give the optimal results for each object. Instead, an unsupervised metric (Section 4.4) is used to approximate the quality of segmentation and help the framework compare the results and converge to the initial estimate of the segmentation algorithm and the settings. The obtained results are then adjusted through the additional training by user input in the following active learning iterations (Fig. 5).

4.4. Unsupervised metric for estimation of the segmentation quality

For the unsupervised iteration where the framework suggests the initial estimate of the segmentation algorithm and the setting per object, an unsupervised metric (step 4 in Fig. 2) is needed to roughly compare the results of different segmentations. In principle, every unsupervised metric that can roughly evaluate the segmentation results can be used here. We particularly use a metric based on two criteria; cross-entropy (Csiszár et al., 2004) and "busyness".

$$\text{Cross-Entropy} = -\sum_{i=1}^{n} P(i) \times log(Q(i/m))$$

The variable i represents the intensity value of pixels, while n and m indicate the number of pixels located inside and outside the segmented line, respectively. $P(i)$ and $Q(i/m)$ are probability distributions associated with the pixel intensity values inside and outside the segmented line, respectively. The segmentation line should be on the pixels separating the background signal which is mostly noise from the object signal. A good segmentation line will result in the maximized cross-entropy between these two discrete sets (object and background).

The other criterion is *busy* which is based on the measure of "busyness" (Weszka and Rosenfeld, 1978; Zhang et al., 2008) in the image, with the assumption that the ideal objects and background are not strongly textured and have simple compact shapes.

$$Busyness = \sum |\nabla^2 I(x, y)|$$

∇^2 represents the Laplacian operator, which computes the second derivative of the image intensity concerning both x and y coordinates. $I(x, y)$ represents the intensity value of the image at coordinates (x, y). By taking the absolute value of the Laplacian, we are interested in measuring the overall magnitude of these intensity changes or variations, which can be indicative of texture or busyness. The sum of absolute values of a Laplacian is used to measure busyness (Weszka and Rosenfeld, 1978) and it is meant to be minimized.

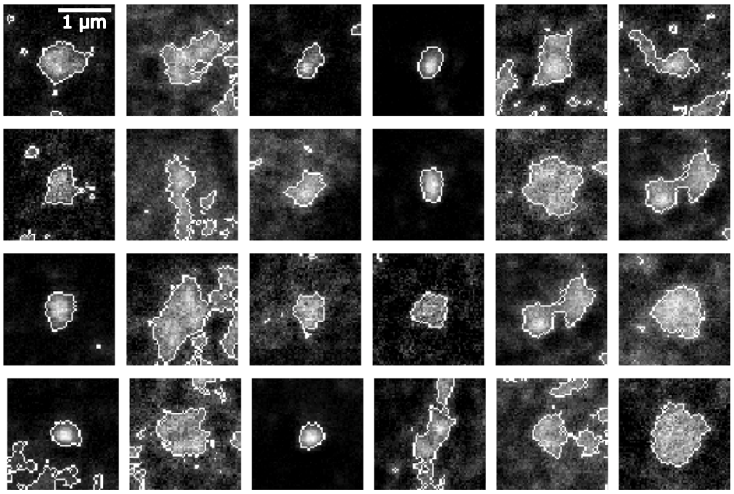

Fig. 5. Different objects annotated with the initial estimate of the segmentation algorithm and the settings, which is obtained by the framework through unsupervised iterations using an unsupervised metric defined in Section 4.4. Representative Pol II Ser5P clusters mid-plane obtained by STEDD super-resolution microscopy from a fixed sphere-stage zebrafish embryo. Each image is of size 60 × 60 pixels. The images are randomly selected from a dataset of size 148.

4.5. Active learning iterations and query strategies

The metric described above can just approximate the quality of segmentation and thus cannot be used for fully-automated training. At the same time, asking for user input on every object is very cumbersome and not practical. Thus, we propose the combination of active learning and weakly supervised training to reduce the number of interactions by objective selections of samples and querying users. To selectively choose the samples later presented to the user, the framework needs a query strategy (step 5 in Fig. 2). In our framework, the three following query strategies are combined.

- **Querying from diverse clusters:** To comprehensively explore the feature space, the framework guarantees that the selected objects are drawn from various regions of the feature space. This is achieved by clustering all samples into different groups and then selecting samples from diverse groups within the feature space. (Section 4.6 and Fig. 7).
- **Querying the most informative objects:** We consider samples having the highest number of neighbors, located within a specified radius, the most informative sample (Fig. 6B). Usually, similar objects need similar segmentation algorithms and settings. By adjusting these informative objects, many other objects are indirectly adjusted and therefore the active learning iterations converge faster.
- **Querying the most uncertain objects:** We consider samples with the largest distance to the center of the cluster as the most uncertain samples (Fig. 6C).

In every active learning interaction (each user interaction), we pick two samples from every cluster, the most informative and the most uncertain ones, and present them to the user. Fig. 6 shows examples of selected most informative and most uncertain objects. The framework then directly adjusted the segmentation settings associated with the queried objects according to the user feedback. Afterward, the settings associated with all other objects are also updated through an extra learning procedure (Section 4.7). A detailed explanation of the update procedure is included in Section 4.7.

4.6. Automatic determination of the clusters

As described above, our query strategy includes picking samples from diverse clusters. In this section, we describe how these clusters are determined (step 6 in Fig. 2) Dividing the feature space into different clusters enables querying from diverse choices of samples. Every clustering algorithm can be used. However, we used the k-Means clustering (Likas et al., 2003). To determine the optimal number of clusters, the framework uses silhouette analysis (Thinsungnoena et al., 2015; Kodinariya et al., 2013). This measure has a value in the range of $[-1, 1]$. Values near $+1$ indicate that the sample is far away from the neighboring clusters. A value of 0 indicates that the sample is on or very close to the decision boundary between two neighboring clusters and negative values indicate that those samples are assigned to the wrong cluster. The optimal number of clusters should result in the highest silhouette values over all samples. After performing a standard clustering with the optimal number of clusters obtained via silhouette coefficient (Fig. 7A), the framework also separates the samples which are considered as being close or close to the borderline (Fig. 7B) as another cluster.

4.7. Tuning the segmentation in each active learning iteration

In this section, the last step (step 7) in Fig. 2 is explained. In every active learning iteration, the framework first presents some objects to the user and then the settings associated with the queried objects are adjusted according to the user feedback. To be able to adjust the other, unlabeled objects as well, a mathematical formulation of the mapping between every object and the setting associated with it is needed. By updating the mathematical formulation, the settings are all automatically updated too.

To implement this mathematical formulation, the framework uses a random forest regressor (Ho, 1995) mapping every object to the setting associated with it. When an object is chosen and presented to the user, the setting of the presented object is updated. The regressor with the updated label is then re-trained and this is where the settings associated with others are also updated.

Once a sample is chosen, it is marked as visited and it will not be presented to the user again. The process of choosing samples, presenting them to the user, adjusting the setting, and updating the regressor is repeated until convergence. The framework is considered as converged either when the user does not request any further adjustment or all the samples are marked.

5. Evaluation

In this section, we present a biological case study, which allowed us to evaluate our framework based on the following three research questions:

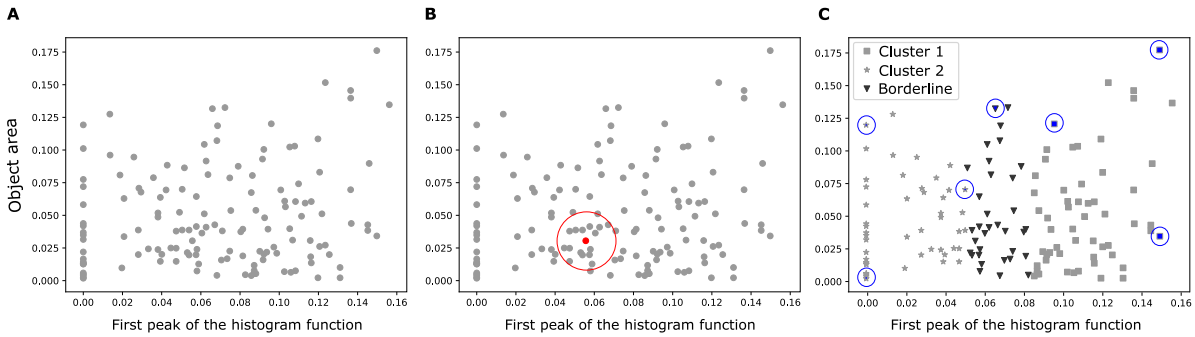

Fig. 6. Example results of query strategies. (A) Representative 2D visualization of all samples in the feature space. (B) The most informative sample with the largest number of neighbors specified with red. (C) Uncertain samples with the farthest distance to the mean sample of clusters specified with blue (clusters are explained in Section 4.6). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

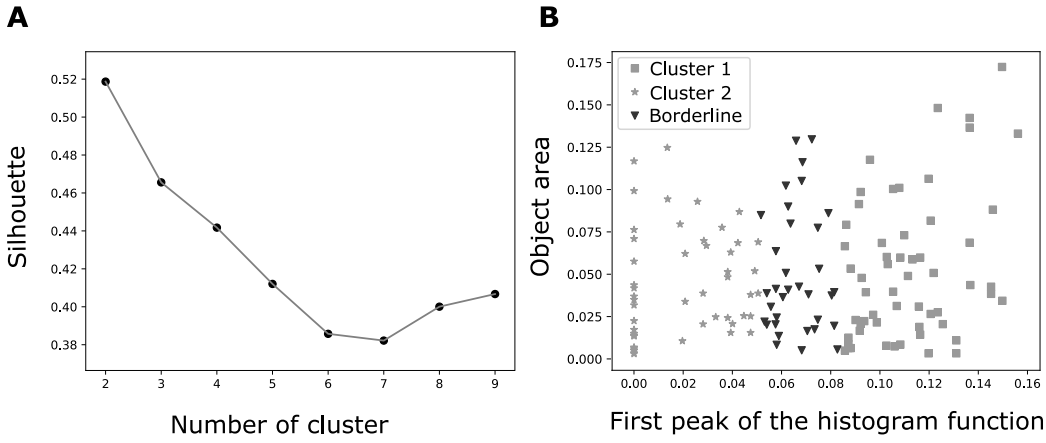

Fig. 7. (A) Silhouette coefficient for a different number of clusters. The maximum value indicates the optimal number of clusters which in this case the optimal number is 2. (B) Representative 2D visualized data samples, where only 2 features (Area and local maxima) are selected. Three Different clusters (cluster 1, cluster 2, and borderline) with different marks based on the optimal number of clusters (2).

- RQ1: Is the framework, from the end-user perspective, converging to the desired results?
- RQ2: How satisfying is the quality of the segmentation results from the end-user perspective?
- RQ3: How effective is our framework in helping end-users with their segmentation tasks?

We conduct all procedures on a laptop with Intel(R) Core(TM) i7-8665U CPU 1.90 GHz 2.11 GHz processor and 32.0 GB RAM, using the Microsoft Windows 11 operating system. The code is written in Python 3, uses the "Scikit-learn" package, and is publicly released on Hajiabadi (2022a).

5.1. Experimental setup

We used input images recorded by STEDD microscopy in a previous study (Pancholi et al., 2021; Hilbert, 2021). The images are maximum-intensity projections of cell nuclei in fixed zebrafish embryos, where Pol II Serine 5 phosphorylation was labeled by indirect immunofluorescence (Fig. 1). The dataset contained images of 60 nuclei. We extract an object set based on cropped ROI images of 60 × 60 pixel size, which are centered on prominent groups of Pol II. We obtained a set of 148 objects that were used for the further application of our framework.

To address the research questions we ask five biologists to use our script for the adjustment of object segmentation (second scenario in Fig. 9). In line with variety in object morphologies mentioned in Pancholi et al. (2021), a feature space (as introduced in Section 4.2) is built and objects are categorized in three clusters (Section 4.2, Section 4.6). Initially, through an unsupervised iteration, segmentation algorithms listed in Table 1 are applied to the objects (Section 4.3) and the preliminary estimate of the segmentation algorithm and the settings are obtained by optimization of the unsupervised metric introduced in Section 4.4 (Fig. 5). This whole unsupervised iteration is conducted without any query from the user.

We then mathematically formulate the mapping function between objects and the associated settings by training a random forest regressor on the samples. The input of the regressor is the object feature vectors and the output is the associated settings. Then through every active learning iteration, six objects (according to the criteria explained in Section 4.5) are selected and presented to the user. With the user input, the settings associated with the presented objects and the neighbors are directly adjusted. Afterward, the regressor is re-trained using the updated input and output, and the settings associated with all other objects are also updated (Section 4.7). The active learning iteration continues until convergence. We consider the framework as converged if there is no further adjustment request from the user or all the objects are marked as visited.

To address the RQ1, we randomly select 10 objects and randomly alternated the results of the last five iterations to assignments of choices [1..5] to prevent bias in choice. Fig. 8 demonstrates the segmented results for some randomly selected objects in the last five iterations. We provided the segmented objects to the expert biologists and ask them to, based on their visual assessment, sort the segmented images from good to bad. We also provided a choice if an object in some iterations looks almost unchanged. Table 2 reports the evaluation results, indicating that in 90% of the cases, the last iteration contains the best results by visual assessment of the expert. In the 40% and 10% of cases, the segmented objects look almost unchanged in the last two and three

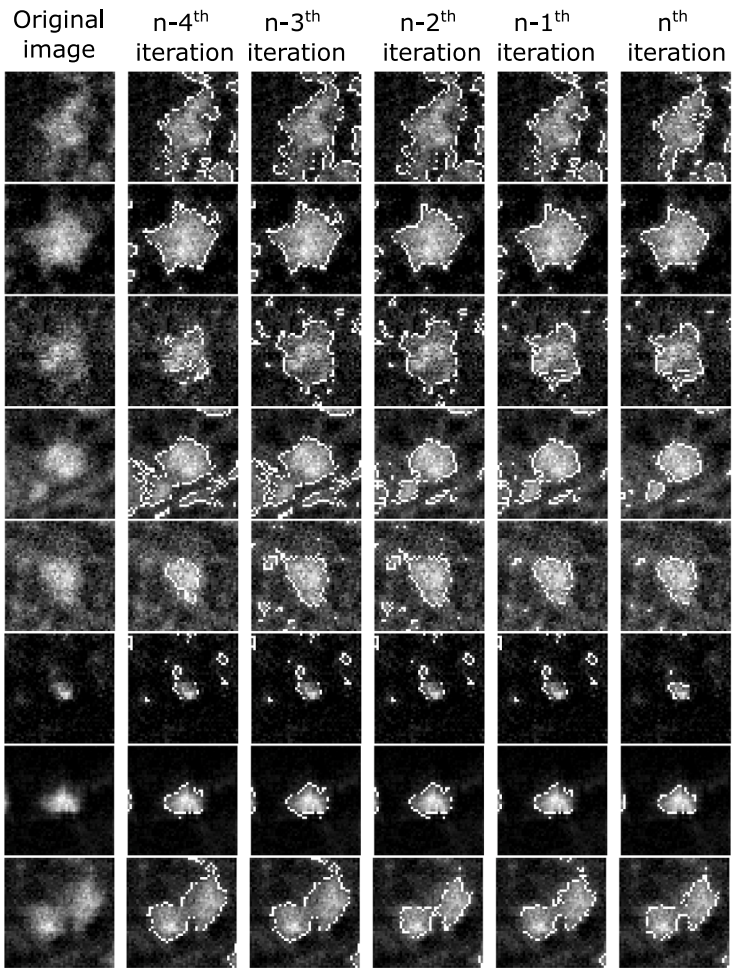

Fig. 8. Different objects annotated with the segmentation line obtained by the framework in the last 5 iterations while n_{th} iteration indicates the last iteration. Each image is of size 60 × 60 pixels, The images are randomly selected from a dataset of size 148.

Table 2
Evaluations results obtained by the experts.

Choice	Percentage
The last iteration contains the best results	90%
Segmented objects in the last two iterations remain unchanged	40%
Segmented objects in the last three iterations remain unchanged	10%

Table 3
Evaluation results performed by the experts.

Choice	Percentage
Results obtained by our framework outperform	80%
Results obtained after expert tuning outperform	0%
Both results look almost similar	10%
It depends on ...	10%

iterations respectively. The results indicate that the learning procedure is converging.

To address RQ2, we need a reference to make the comparison. To establish a reference for the evaluation of our training results, we asked one expert biologist with extended knowledge of ML to choose a segmentation algorithm based on the object set, and manually tune the settings of the algorithm (first scenario in Fig. 9). This expert had a wide knowledge of segmentation algorithms and was very familiar with the application data. The adjustment for this expert took around 130 min. The results that we obtained from the expert can be considered the best results that the use of the existing resource can achieve. We then present 10 randomly selected pairs of reference objects and the segmented results obtained by our script to the expert biologist to evaluate the quality of segmentation by visual assessment. A sample questionnaire presented to the biologists is available at Hajiabadi (2022b).

Using a human to visually assess the result is a kind of subjective metric. We provided four choices for each comparison pair: (i) segmented image A is better, (ii) segmented image B is better, (iii) both segmentation results are almost equal and (iv) the evaluation of segmentation depends on an additional consideration formulated by the expert. We randomly alternated the assignment to choices A and B to prevent bias in choice. The evaluation indicates that We provided four choices for each comparison pair: (i) segmented image A is better, (ii) segmented image B is better, (iii) both segmentation results are almost equal and (iv) the evaluation of segmentation depends on an additional consideration formulated by the expert. We randomly alternated the assignment to choices A and B to prevent bias in choice. The evaluation indicates that, in the eyes of an expert, in 90% of cases the segmentations obtained by our framework are better than or almost equal to the segmentations obtained after manual tuning by an expert (Table 3). an expert, in 90% of cases the segmentations obtained by our framework are better than or almost equal to the segmentations obtained after manual tuning by an expert (Table 3).

To address the RQ3, we ask five biologists to use our script and segment objects with our framework assistance (second scenario of Fig. 9). These five biologists are among the final target groups whom the framework eventually assists. We then record the time spent and the

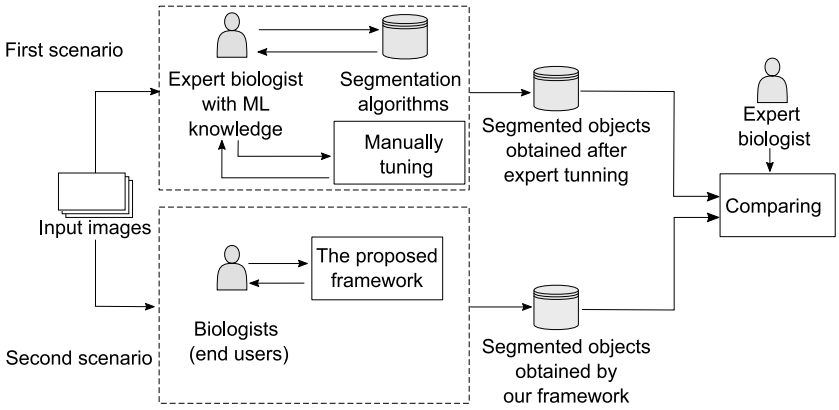

Fig. 9. A subjective metric is used for evaluation. The results obtained by our framework are compared to the results obtained after tuning the algorithm by an expert. This evaluation is based on a human visual assessment.

Table 4
Number of interactions and the dedicated time required for convergence.

	Total number of interactions	Dedicated time	Simplicity 1: not simple 7: maximally simple
Case study 1	5	235 s	7
Case study 2	7	365 s	5
Case study 3	6	250 s	5
Case study 4	4	165 s	7
Case study 5	5	250 s	7
Average	5.4	253 s	6.2

number of interactions required for tuning the script. Every interaction is one iteration of active learning. In Table 4, We refer to the experiment associated with every biologist as a case study. Table 4 reports the obtained results, indicating that the framework on average, converges within 5.4 user interactions. The convergence takes an average of 253 s. This time includes only the time when the framework is working. The time that biologists take to decide on which result is better is not calculated. We then asked the biologists to rate the simplicity of our framework from 1 (not simple) to 7 (maximally simple) and the simplicity of our proposed framework was rated high 6.2 out of 7.

6. Threats to validity

The validity of our conclusions could be mainly compromised by the example application and evaluation setting. We use only example data from one type of biological sample, recorded on one type of microscope. The experiment contains different samples to account for day-to-day variability in sample quality. However, image characteristics, e.g., intensity distribution and image resolution, can differ fundamentally between different microscopes. Adding data from other types of samples and microscopes would strengthen the generalizability of our findings, thereby representing a threat to external validity.

The biologists working in the same laboratory tested the framework by tuning segmentation on the object set. Thereby, the segmented objects obtained by our framework could be biased by the individual users interacting with it. Accordingly, user interaction is another threat to external validity, as it is unclear whether different users would obtain different segmentation results. Bias could be further reduced by increasing the number of users, and also recruiting users from different research groups.

The third threat to our experiment design is due to the number of expert biologists assessing the segmentation quality. We used only one expert biologist to visually assess the quality of segmentation results. Accordingly, it is unclear whether our measure of segmentation quality is a suitable indicator for the satisfaction of end users, as other experts might assess the quality differently. Assessment by a larger panel of experts would ensure a more unbiased result that increases the construct validity of our study.

To promote the reliability of our results, we enable the reproduction or replication of our study by making the underlying artifacts publicly available. These artifacts include the framework implementation (Hajiabadi, 2022a), the questionaries (Hajiabadi, 2022b), and the dataset reused from a previous study (Hilbert, 2021).

7. Related work

Our work primarily relates to two categories: (1) interactive approaches with user-guided parameter selection, and (2) weakly supervised segmentation methods. While these two groups can be merged, we will review them separately.

It is almost a decade since interaction object segmentation first emerged. The early approaches are mostly formulated as energy minimization tasks on a graph. An image is first represented as an undirected graph, which can be defined by a joint probability p. The goal is to assign a label to each pixel (0 represents the background and 1 for the objects) in a way that the probability p is minimized (Bai and Sapiro, 2009; Boykov and Jolly, 2001; Gulshan et al., 2010; Cagnoni et al., 1999). There have been some other interactive segmentation works that require users to annotate the edges (extreme-left, extreme-right, extreme-top, and extreme-bottom) of objects. The connection lines between edges represent the boundary line. Training on the annotated objects, the method can hopefully later detect the boundaries of unseen objects (Kwatra et al., 2003; Papadopoulos et al., 2017). The region inside the boundaries is considered as foreground (object) and the outside area is labeled as background. There are several major challenges regarding the transfer of these methods in the biological subcellular segmentation tasks. Firstly, some subcellular components cannot be inherently represented by edges and secondly, the whole region inside the boundaries sometimes is not the object, for example, objects with the shape of doughnuts.

Recently, DL methods have been extensively used in interactive segmentation tasks (Xu et al., 2016; Wang et al., 2019). Some deep learning tasks are based on detecting the four edges, where for a part of the dataset, users annotate the edges, and the network is later trained to detect the edges for unseen input (Maninis et al., 2018). There are some other weakly-supervised DL methods for image segmentation (Castrejon et al., 2017; Ling et al., 2019), where users should provide bounding boxes for a part of the dataset, and a network is learned to detect the bounding box for the rest of the dataset. These approaches are sometimes combined with reinforcement learning which asks users to mark the foreground. These approaches have been successfully used in CT/MRI image segmentation, where the organ appears in similar shapes in all images. In some studies (Caselles et al., 1997; Acuna et al.,

2019), the user inputs (bounding box annotations) have been combined with the graph convolutional networks (GCN). In the training phase, the manually annotated data is used and the GCN learns to represent an object with a polygon around it. To refine the trained model, the predicted polygon can be further adjusted by the user interaction.

Some researchers are investigating different conditions and proposing methods that leverage weak supervision for image segmentation tasks, such as image-level labels and self-attention mechanisms, to generate pseudo-instance-level annotations (Nishimura et al., 2021). These approaches effectively address the challenges of limited annotations and enable accurate instance segmentation. For example, Box2Mask (Chibane et al., 2022) is a method for weakly supervised 3D semantic instance segmentation using bounding boxes. The proposed approach leverages bounding box annotations to generate instance masks in a weakly supervised manner. It achieves accurate segmentation results by incorporating a 3D Mask-RCNN model and a self-supervised learning framework, demonstrating the effectiveness of weak supervision for 3D instance segmentation tasks (Chibane et al., 2022). Another study, Snorkel (Ratner et al., 2017), leverages heuristics, rules, and other noisy sources to generate training labels, bypassing the need for hand-labeled data. The framework incorporates a generative model that estimates the accuracies of the labeling functions, allowing for the creation of large training datasets quickly and efficiently. Segment Anything Model (SAM) (Chen et al., 2023) enhances pseudo labels for weakly supervised semantic segmentation. SAM utilizes multiple cues, including object boundaries and spatial consistency, to refine and improve the quality of pseudo labels. Experimental results demonstrate that SAM effectively boosts the performance of weakly supervised semantic segmentation methods, achieving more accurate and precise segmentations in various settings.

However, there is a general limitation in the use of methods based on DL, which is the requirement of annotated images. While some researchers aim to reduce the amount of annotated data needed, the process of creating training annotations manually is time-consuming and requires domain expertise. The same limitation arises when a network needs to be re-trained for new datasets, which typically also requires additional annotation by experts (Hesamian et al., 2019). There are specific challenges associated with transferring these approaches to biological image segmentation, especially when dealing with a large number of objects within a single image. For instance, it can be practically infeasible to annotate the bounding boxes or edges of all the objects in an image that contains, for example, 200 cells. Another limitation of biological image segmentation is the high degree of variability in object shapes. In particular, when the object of interest appears in complex shapes with unclear edges and boundaries, which is a common occurrence in biological subcellular components (Jahanifar et al., 2019). After reviewing the literature, we have come to realize that an effective approach for biological segmentation minimizes user interaction while allowing for easy customization with unseen data.

8. Conclusion

Detecting the different components that constitute a biological cell is a crucial task in the analysis of biomedical images. While there are automated approaches available for object detection, applying these algorithms to unknown datasets presents a significant challenge in practical applications. The primary hurdle lies in the requirement for annotated objects, which typically necessitates the involvement of domain specialists, making it a labor-intensive process. In this paper, we propose a framework that facilitates the customization of the segmentation algorithm for specific object types, even in the presence of unseen data, with minimal user interactions. This framework relies on a mapping function that correlates object morphology features with specific algorithm settings. We applied this framework to a biological case study and validated its performance and simplicity through the aforementioned study.

In future work, we plan to investigate the impact of object morphology feature space on the subsequent steps of our framework. We aim to enhance the representation of object sets within the feature space by incorporating an interactive approach rather than relying solely on unsupervised methods.

CRediT authorship contribution statement

Hamideh Hajiabadi: Writing – review & editing, Writing – original draft, Validation, Software, Project administration, Methodology, Conceptualization. **Christopher Gerking:** Writing – original draft, Conceptualization. **Lennart Hilbert:** Supervision, Methodology, Data curation, Conceptualization. **Anne Koziolek:** Validation, Supervision, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work is supported by the Helmholtz Association (HGF) under the joint research school “HIDSS4Health – Helmholtz Information and Data Science School for Health” and the Helmholtz program Natural Artificial and Cognitive Information Processing (NACIP). HH was partially supported by funding from HIDSS4Health. LH was supported by the Priority Program Molecular Mechanisms of Functional Phase Separation of the German Science Foundation (DFG-SPP2191). CG and AK were supported by funding from the topic “Engineering Secure Systems” of the HGF and by KASTEL Security Research Labs. We thank Irina Mamontova, Roshan Prizak, Agnieszka Pancholi and Saeed Nasseri and his research team for their help in conducting the case study. We also express our gratitude to Nils Friederich for providing feedback on our manuscript.

References

Acuna, D., Kar, A., Fidler, S., 2019. Devil is in the edges: Learning semantic boundaries from noisy annotations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11075–11083.

Al-Kofahi, Y., Zaltsman, A., Graves, R., Marshall, W., Rusu, M., 2018. A deep learning-based algorithm for 2-D cell segmentation in microscopy images. BMC Bioinf. 19 (1), 1–11.

Alemi Koohbanani, N., Jahanifar, M., Zamani Tajadin, N., Rajpoot, N., 2020. NuClick: A deep learning framework for interactive segmentation of microscopic images. Med. Image Anal. 65, 101771.

Apthorpe, N., Riordan, A., Aguilar, R., Homann, J., Gu, Y., Tank, D., Seung, H.S., 2016. Automatic neuron detection in calcium imaging data using convolutional networks. Adv. Neural Inf. Process. Syst. 29, 3270–3278.

Arganda-Carreras, I., Kaynig, V., Rueden, C., et al., 2017. Trainable Weka Segmentation: a machine learning tool for microscopy pixel classification. Bioinformatics 33 (15), 2424–2426.

Bai, X., Sapiro, G., 2009. Geodesic matting: A framework for fast interactive image and video segmentation and matting. Int. J. Comput. Vision 82 (2), 113–132.

Berg, S., Kutra, D., Kroeger, T., et al., 2019. Ilastik: interactive machine learning for (bio) image analysis. Nature Methods 16 (12), 1226–1232.

Boykov, Y.Y., Jolly, M.-P., 2001. Interactive graph cuts for optimal boundary & region segmentation of objects in ND images. In: Proceedings Eighth IEEE International Conference on Computer Vision, Vol. 1. ICCV 2001, IEEE, pp. 105–112.

Cagnoni, S., Dobrzeniecki, A.B., Poli, R., Yanch, J.C., 1999. Genetic algorithm-based interactive segmentation of 3D medical images. Image Vis. Comput. 17 (12), 881–895.

Cantaloube, S., Romeo, K., Le Baccon, P., et al., 2012. Characterization of chromatin domains by 3D fluorescence microscopy: an automated methodology for quantitative analysis and nuclei screening. BioEssays 34 (6), 509–517.

Carpenter, A.E., Jones, T.R., Lamprecht, M.R., Clarke, C., Kang, I.H., Friman, O., et al., 2006. CellProfiler: image analysis software for identifying and quantifying cell phenotypes. Genome Biol. 7 (10), 1–11.

Caselles, V., Kimmel, R., Sapiro, G., 1997. Geodesic active contours. Int. J. Comput. Vision 22 (1), 61–79.

Castrejon, L., Kundu, K., Urtasun, R., Fidler, S., 2017. Annotating object instances with a polygon-rnn. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5230–5238.

Chen, J., Ding, L., Viana, M.P., Lee, H., Sluezwski, M.F., Morris, B., Hendershott, M.C., Yang, R., Mueller, I.A., Rafelski, S.M., 2020. The Allen Cell and Structure Segmenter: a new open source toolkit for segmenting 3D intracellular structures in fluorescence microscopy images. Cold Spring Harbor Laboratory, http://dx.doi.org/10.1101/491035.

Chen, T., Mai, Z., Li, R., Chao, W.-l., 2023. Segment anything model (SAM) enhanced pseudo labels for weakly supervised semantic segmentation. arXiv preprint arXiv:2305.05803.

Chen, Y., Ming, D., Zhao, L., et al., 2018. Review on high spatial resolution remote sensing image segmentation evaluation. Photogramm. Eng. Remote Sens. 84 (10), 629–646.

Chibane, J., Engelmann, F., Anh Tran, T., Pons-Moll, G., 2022. Box2Mask: Weakly supervised 3D semantic instance segmentation using bounding boxes. In: Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXI. Springer, pp. 681–699.

Csiszár, I., Shields, P.C., et al., 2004. Information theory and statistics: A tutorial. Found. Trends Commun. Inf. Theory 1 (4), 417–528.

Dey, N., Rajinikanth, V., Ashour, A.S., Tavares, J.M.R., 2018. Social group optimization supported segmentation and evaluation of skin melanoma images. Symmetry 10 (2), 51.

Dougherty, E., 2018. Mathematical Morphology in Image Processing, vol. 1, CRC Press.

Guerrero-Pena, F.A., Fernandez, P.D.M., Ren, T.I., Yui, M., et al., 2018. Multiclass weighted loss for instance segmentation of cluttered cells. In: 2018 25th IEEE International Conference on Image Processing. ICIP, IEEE, pp. 2451–2455.

Gulshan, V., Rother, C., Criminisi, A., Blake, A., Zisserman, A., 2010. Geodesic star convexity for interactive image segmentation. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE, pp. 3129–3136.

Hajiabadi, H., 2022a. Interactive Segmentation. Zenodo, http://dx.doi.org/10.5281/zenodo.6504500.

Hajiabadi, H., 2022b. Questionaries. Zenodo, http://dx.doi.org/10.5281/zenodo.7682591.

Hajiabadi, H., Hilbert, L., Koziolek, A., 2022. Easing the reuse of ML solutions by interactive clustering-based autotuning in scientific applications. In: 2022 48th Euromicro Conference on Software Engineering and Advanced Applications. SEAA, pp. 5–12.

Hesamian, M.H., Jia, W., He, X., Kennedy, P., 2019. Deep learning techniques for medical image segmentation: achievements and challenges. J. Dig. Imaging 32 (4), 582–596.

Hilbert, L., 2021. Analysis of RNA Polymerase II Phosphorylation in Stimulated Emission Double Depletion (STEDD) Microscopy Images. Zenodo, http://dx.doi.org/10.5281/zenodo.4973062.

Ho, T.K., 1995. Random decision forests. In: Proceedings of 3rd International Conference on Document Analysis and Recognition, Vol. 1. IEEE, pp. 278–282.

Ivashkevich, A.N., Martin, O.A., Smith, A.J., et al., 2011. γH2AX foci as a measure of DNA damage: a computational approach to automatic analysis. Mutat. Res. Fund. Mol. Mech. Mut. 711 (1–2), 49–60.

Jahanifar, M., Koohbabnani, N.A., Rajpoot, N., 2019. Nuclick: From clicks in the nuclei to nuclear boundaries. In: MICCAI 2019 Computational Pathology Workshop COMPAY.

Jakhar, D., Kaur, I., 2020. Artificial intelligence, machine learning and deep learning: definitions and differences. Clin. Exp. Dermatol. 45 (1), 131–132.

Kodinariya, T.M., Makwana, P.R., et al., 2013. Review on determining number of Cluster in K-means Clustering. Int. J. 1 (6), 90–95.

Kwatra, V., Schödl, A., Essa, I., Turk, G., Bobick, A., 2003. Graphcut textures: Image and video synthesis using graph cuts. Acm Trans. Graph. (tog) 22 (3), 277–286.

Li, C.H., Lee, C., 1993. Minimum cross entropy thresholding. Pattern Recognit. 26 (4), 617–625.

Li, G., Liu, T., Nie, J., Guo, L., et al., 2008. Segmentation of touching cell nuclei using gradient flow tracking. J. Microsc. 231 (1), 47–58.

Likas, A., Vlassis, N., Verbeek, J.J., 2003. The global k-means clustering algorithm. Pattern Recognit. 36 (2), 451–461.

Ling, H., Gao, J., Kar, A., Chen, W., Fidler, S., 2019. Fast interactive object annotation with curve-gcn. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5257–5266.

Maninis, K.-K., Caelles, S., Pont-Tuset, J., Van Gool, L., 2018. Deep extreme cut: From extreme points to object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 616–625.

Matula, P., Verissimo, F., Wörz, S., et al., 2010. Quantification of fluorescent spots in time series of 3D confocal microscopy images of endoplasmic reticulum exit sites based on the HMAX transform. In: Medical Imaging 2010: Biomedical Applications in Molecular, Structural, and Functional Imaging, Vol. 7626. International Society for Optics and Photonics, p. 76261H.

McQuin, C., Goodman, A., Chernyshev, V., et al., 2018. CellProfiler 3.0: Next-generation image processing for biology. PLoS Biol. 16 (7), e2005970.

Nishimura, K., Wang, C., Watanabe, K., Bise, R., et al., 2021. Weakly supervised cell instance segmentation under various conditions. Med. Image Anal. 73, 102182.

Osterwald, S., Deeg, K.I., Chung, I., Parisotto, D., et al., 2015. PML induces compaction, TRF2 depletion and DNA damage signaling at telomeres and promotes their alternative lengthening. J. Cell Sci. 128 (10), 1887–1900.

Otsu, N., 1979. A threshold selection method from gray-level histograms. IEEE Trans. Syst. Man Cybern. 9 (1), 62–66.

Pancholi, A., Klingberg, T., Zhang, W., Prizak, R., Mamontova, I., Noa, A., Sobucki, M., Kobitski, A.Y., Nienhaus, G.U., Zaburdaev, V., et al., 2021. RNA polymerase II clusters form in line with surface condensation on regulatory chromatin. Mol. Syst. Biol. 17 (9), e10272.

Papadopoulos, D.P., Uijlings, J.R., Keller, F., Ferrari, V., 2017. Extreme clicking for efficient object annotation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4930–4939.

Pont-Tuset, J., Marques, F., 2015. Supervised evaluation of image segmentation and object proposal techniques. IEEE Trans. Pattern Anal. Mach. Intell. 38 (7), 1465–1478.

Prewitt, J.M., Mendelsohn, M.L., 1966. The analysis of cell images. Ann. New York Acad. Sci. 128 (3), 1035–1053.

Ratner, A., Bach, S.H., Ehrenberg, H., Fries, J., Wu, S., Ré, C., 2017. Snorkel: Rapid training data creation with weak supervision. In: Proceedings of the VLDB Endowment. International Conference on Very Large Data Bases, Vol. 11, No. 3. NIH Public Access, p. 269.

Sahoo, P.K., Soltani, S., Wong, A.K., 1988. A survey of thresholding techniques. Comput. Vis. Graph. Image Process. 41 (2), 233–260.

Salem, N., Sobhy, N.M., El Dosoky, M., 2016. A comparative study of white blood cells segmentation using otsu threshold and watershed transformation. J. Biomed. Eng. Med. Imag. 3 (3), 15.

Sezgin, M., Sankur, B., 2004. Survey over image thresholding techniques and quantitative performance evaluation. J. Electron. Imaging 13 (1), 146–165.

Sklansky, J., 1978. Image segmentation and feature extraction. IEEE Trans. Syst. Man Cybern. 8 (4), 237–247.

Sommer, C., Straehle, C., Koethe, U., Hamprecht, F.A., 2011. Ilastik: Interactive learning and segmentation toolkit. In: 2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro. IEEE, pp. 230–233.

Taha, A.A., Hanbury, A., 2015. Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool. BMC Med. Imaging 15 (1), 1–28.

Thinsungnoena, T., Kaoungkub, N., Durongdumronchaib, P., Kerdprasopb, K., Kerdprasopb, N., et al., 2015. The clustering validity with silhouette and sum of squared errors. Learning 3 (7).

Wang, Z., Acuna, D., Ling, H., Kar, A., Fidler, S., 2019. Object instance annotation with deep extreme level set evolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7500–7508.

Wang, Z., Wang, E., Zhu, Y., 2020. Image segmentation evaluation: a survey of methods. Artif. Intell. Rev. 53 (8), 5637–5674.

Weszka, J.S., Rosenfeld, A., 1978. Threshold evaluation techniques. IEEE Trans. Syst. Man Cybern. 8 (8), 622–629.

Wu, H., Barba, J., Gil, J., 2000. Iterative thresholding for segmentation of cells from noisy images. J. Microsc. 197 (Pt 3), 296–304.

Xu, N., Price, B., Cohen, S., Yang, J., Huang, T.S., 2016. Deep interactive object selection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 373–381.

Yi, J., Tang, H., Wu, P., et al., 2020. Object-guided instance segmentation for biological images. In: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, No. 07. pp. 12677–12684.

Zack, G.W., Rogers, W.E., Latt, S.A., 1977. Automatic measurement of sister chromatid exchange frequency. J. Histochem. Cytochem. 25 (7), 741–753.

Zhang, Y.J., 1996. A survey on evaluation methods for image segmentation. Pattern Recognit. 29 (8), 1335–1346.

Zhang, H., Fritts, J.E., Goldman, S.A., 2008. Image segmentation evaluation: A survey of unsupervised methods. Comput. Vis. Image Underst. 110 (2), 260–280.

Hamideh Hajiabadi is currently a doctoral researcher at Karlsruhe Institute of Technology (KIT) in Germany. Her research focuses on the intersection of software engineering and biology, encompassing a multidisciplinary approach. Specifically, she explores the applications of software engineering principles and methodologies in the field of biology.

Christopher Gerking is a postdoctoral researcher on software engineering at Karlsruhe Institute of Technology (KIT), Germany. He received his Ph.D. degree from Paderborn University in 2020. His research interest is the intersection of software engineering and IT security, focusing on formal methods for provable security of mobility systems.

Lennart Hilbert holds the position of Tenure-Track Junior-Professor of Systems Biology/Bioinformatics at Karlsruhe Institute of Technology (KIT) in Karlsruhe, Germany.

He received his Ph.D. degree in 2014 from McGill University, Montreal, Quebec, Canada. From 2014 to 2018, he served as a postdoctoral fellow at the Center for Systems Biology Dresden, the Max–Planck Institute of Molecular Cell Biology and Genetics, and the MaxPlanck Institute for the Physics of Complex Systems, all located in Dresden, Germany.

Anne Koziolek is a full professor of software engineering at Karlsruhe Institute of Technology (KIT), Germany. She received her Ph.D. degree from KIT in 2011 and was a Postdoc at University of Zurich until 2013. Her current research interest is how to reconcile agile, code-centric software development with model-based software engineering, especially regarding models for quality prediction as well as design models.