Creating segmentation masks for cell segmentation using interactive segmentation model (SAM)

MEENAKSHI GIRISH NAIR, University of Twente, The Netherlands

Cell segmentation is a crucial process in the biomedical field as the size, shape or number of cells can provide a plethora of information for medical diagnosis of many diseases[7]. However, cell segmentation can be a difficult task to tackle owing to the irregular shapes and overlapping cells that lead to poor boundary distinction. In addition to the difficulty in the segmentation of cells, the availability of ground truth to train the model is a bottleneck due to the the resource intensive process of creating labeled segmentation masks. This paper implements an automatic cell segmentation model using Mask R-CNN trained on microscopic cell images dataset. The Segment Anything Model (SAM) will be used for interactive segmentation of images to produce the ground truth to train the model. The primary objective of this study is to analyse how SAM prompts can be used interactively in order to produce accurate segmentation masks to use as ground truth to train segmentation models like Mask R-CNN.

Additional Key Words and Phrases: Cell segmentation, instance segmentation, SAM, Mask R-CNN

1 INTRODUCTION

In recent times, the growth of AI has resulted in vast advancements in many fields. In the biomedical field, a rather groundbreaking use of AI is for cell segmentation. Cell segmentation is an essential step in bio medical research for quantitative analysis of cells, disease diagnosis, stem cell research and more[14]. Cell segmentation is the process by which each instance of the cells, i.e. the region of interest, is separated from the rest of the image. Using this, it is possible to analyze the biological features of the cell, which provides great insight into cellular functions and interactions. Owing to the complex nature of cell segmentation, caused by overlapping cells and the difficulty in accurately detecting cell boundaries, cell segmentation can be quite erroneous[4]. An error in cell segmentation can result in potential systematic errors in all the downstream functions [6]. Hence, the accuracy of the cell segmentation is of utmost importance.

There are multiple existing AI deep learning models like U-Net, Mask R-CNN, DeepLab etc., which when given an image, outputs the segmentation masks of the image. All these models require large amounts of annotated data to train so that it can perform with higher accuracy and precision on new data. The creation of annotated data is time consuming and expensive [15], monetarily and in terms of human resources. When training a segmentation model for cell segmentation, it needs to be provided with ground truth segmentation masks for the model to learn what it needs to predict. When creating the ground truth segmentation masks, there are multiple challenges faced. The microscopic images have overlapping cell boundaries, which poses challenges when creating segmentation masks for each cell instance.

The Segment Anything Model (SAM) is an interactive segmentation model, that lets the user create segmentation masks for images by specifying points and bounding boxes as prompts which guides the algorithm to segment the region of interest.

In this research, SAM will be used to generate segmentation masks for fluorescent microscopic cell images. The corresponding bright field images and the created segmentation masks will then be used as training and validation data to train the Mask R-CNN model. The main focus of this research is to see how well the segmentation masks created by SAM serves as ground truth to training a segmentation model on bright field images. A main contribution made by this paper is the dataset of segmentation masks and labels that is created in this research.

2 LITERATURE REVIEW

Deep learning models often require large datasets in order for it to achieve high performance. Large amount of data provides varied examples that would help the model generalise, preventing over fitting of data[5]. Owing to the requirement to have a large dataset, there has been research into creating them, such as the LIVECell dataset, which is a high-quality dataset of manually annotated, label-free images of cells [3]. This includes around 1.6 million cells from various cell types. However, even after training with more than 1.6 million cell instances, the accuracy sometimes fail to reach saturation[15]. Expanding these datasets would mean more manual annotation of cell images. However, manual creation of annotated datasets are very resource intensive. When dealing with images that contain a lot of cell instances, this process would take a lot of time. The manual annotations will also be prone to subjectivity and it can also be affected by fatigue and attention span of the annotator. This can be prominent especially in cases where there are a lot of overlapping cells and there are no distinct boundaries to segment each cell. In this case, the ground truth masks are created based on the annotator's interpretation of where the boundary is. These challenges makes the creation of accurate ground truth segmentation masks hard, especially when made at a very large scale.

A solution to tackle this issue would be to use a smaller training dataset. When using smaller datasets for training, manual creation of segmentation masks would be easier. However using smaller dataset could lead to over fitting due to the model memorising the training data rather than learning the patterns. This can be solved using transfer learning, in which the model is initially trained on a large, varied dataset. And once the training is done, the knowledge it infers from the large dataset can be fine-tuned to predict on the smaller dataset[13].

41st Twente Student Conference on IT, July 05, 2024, Enschede, The Netherlands

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/XXXXXXX.XXXXXXX

In addition to the dataset size, microscopic cell images poses the challenge of accurate segmentation. Cell images are usually made on different focal planes (z-stack levels), where some cells are clearly visible in some focal planes whereas others are blurry. This causes difficulties when creating segmentation masks for each cell instance as some cells do not have clear boundaries in any focal planes. When there are cells that are completely covering the other cell, the boundary contrast is not evident in any z-stack level causing inaccurate segmentation masks being created. This leads to the following problem statement.

3 PROBLEM STATEMENT

In order for the existing image segmentation models to perform cell segmentation, the models would need to be trained on cell images to learn the features. Even when leveraging transfer learning to have the ability to effectively train the model on a smaller dataset, the cell images poses challenges when creating ground truth segmentation masks. This results in the following research question:

3.1 Research question

Given the challenges of generating fully annotated training data for automated cell segmentation models, how can interactive segmentation be used to create accurate segmentation masks?

This can be answered with the following sub-questions:

(1) How can we implement interactive segmentation using SAM to create segmentation masks from fluorescent cell images?
(2) How well do these segmentation masks serve as ground truth when training basic segmentation models, like Mask R-CNN, on bright field cell images?

4 BACKGROUND INFORMATION

In this paper, we are tackling the concern of segmentation tasks and we will be using the Mask R-CNN model for this.

Segmentation tasks can be of two types, semantic segmentation and instance segmentation.

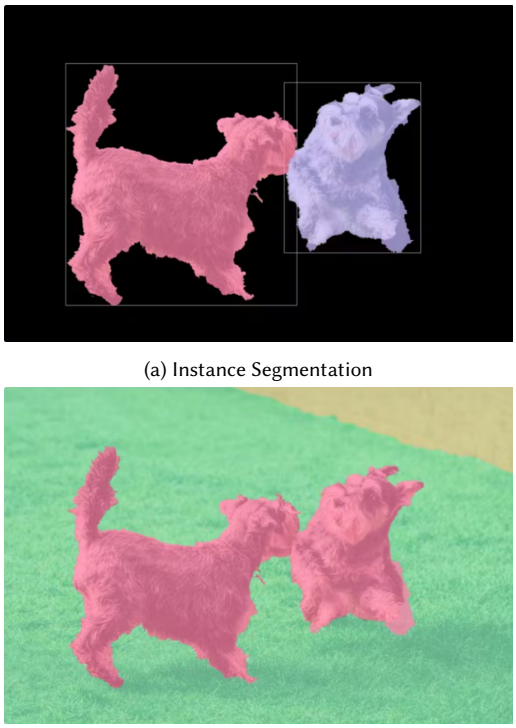
4.1 Semantic segmentation

Semantic segmentation is the process of inferring labels for each pixel in the image and classifying it to the right class[2]. The classes are the regions of interest and the background. If there are multiple occurrences of the same class in the given image, these will be all classified within the same mask in semantic segmentation. Semantic segmentation differs from object detection as it "allows the object of interest to span multiple areas in the image at a pixel level"[12].

4.2 Instance segmentation

Instance segmentation, on the other hand, separates each occurrence of a class and assigns a separate mask to each instance of the class[1]. Each pixel in the image is assigned a unique label depending on what class instance it belongs to. There are multiple techniques used for instance segmentation. One commonly used technique is detection-based instance segmentation where object detection is utilized first to detect the regions of interest, where bounding boxes are generated around the object. The masks are then generated for these regions. Instance segmentation is key in medical imaging as it allows for detailed study of medical images to diagnose various heath conditions[1].

An example of instance and semantic segmentation is shown in Figure 1.



(a) Instance Segmentation

(b) Semantic segmentation

Fig. 1. Instance and Semantic Segmentation [1]

In this paper, the Mask R-CNN model will be trained to perform automatic instance segmentation.

4.3 Mask R-CNN

The Mask R-CNN is an instance segmentation model. It is an extension of the Faster R-CNN model, which is a primarily used for object detection. Alongside the object detection branch, Mask R-CNN has a parallel segmentation branch, that produces segmentation masks for the region of interest. The backbone model of Mask R-CNN consists of the ResNet architecture to extract features of the image. It also contains a Region Proposal Network (RPN), which predicts the likelihood of an object in that region. The regions of interest obtained from RPN is then passed through pooling layers to convert all the regions to the same shape. These regions are then passed through a fully connected network to predict the class labels and bounding boxes. These layers are similar to the Faster R-CNN model. Additional to these layers, Mask R-CNN has a mask head, a fully convolutional network that predicts segmentation mask for each area of interest[10]. With this architecture, Mask R-CNN efficiently predicts labels, bounding boxes and segmentation masks for the objects in an image.

4.4 Other models

During the initial phases of this research, the U-Net model was also considered as a potential candidate for the model training. U-Net is a biomedical image segmentation model, that is designed to use convolutional networks for semantic segmentation of images[8]. While U-Net is more suitable for cell segmentation as it is a biomedical segmentation model, it is mainly for the purpose of semantic segmentation more than for instance segmentation. A sub-goal of this research is mainly to train a model for instance segmentation. Hence, if using the U-Net model, its architecture would need to be modified. Mask R-CNN is specifically for instance segmentation. Hence, by adapting Mask R-CNN on cell images, it can be trained to predict instance level segmentation masks for cell images. Therefore, Mask R-CNN was chosen to be the model that will be trained in this research.

5 METHODOLOGY

This section will explain how the research question will be answered.

5.1 Segment Anything Model

Segment Anything Model (SAM) is an interactive segmentation model, in which the user can specify points, bounding boxes or a combination of both to segment the region of interest from the rest of the image. For each prompt, the user can also assign whether it is part of the foreground or background of the image. The prompts would be an array of coordinate points. For bounding boxes, these are the (x,y) coordinates of the top left corner and the bottom right corner. For the point prompts, it is an array of the (x,y) coordinates of each point and a corresponding array of 0s and 1s, specifying if the point is in the foreground or background of the image. An example is shown in (a) in Figure 2. Using these prompts the model will then produce segmentation masks, which can be further refined by the user by giving in more prompts or adjusting the existing prompts to guide the algorithm. Figure 2 shows how points can be used as prompts to segment an instance of the cell from the fluorescent image.

5.2 Training Mask R-CNN

Once the segmentation masks were created, they were used as the ground truth and Mask R-CNN model was trained with pre-trained weights, to create instance segmentation masks on bright field images. The pre-trained weights used were the weights trained on the COCO dataset. The COCO dataset, is a large scale object detection and segmentation dataset[16]. The masks created by Mask R-CNN model will be evaluated using evaluation metrics like IoU, f1 score, mean average precision and label accuracy.

The basic methodology is visualised in Figure 3.

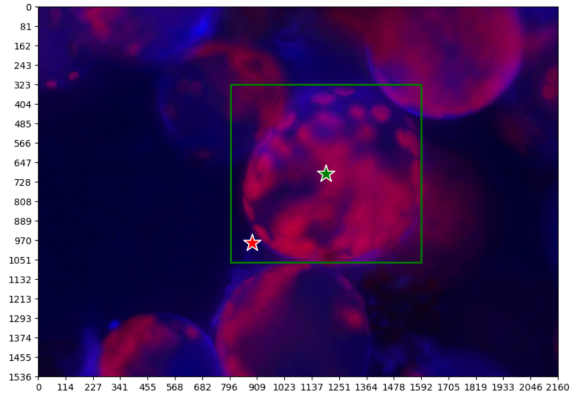
6 EXPERIMENT

6.1 Dataset

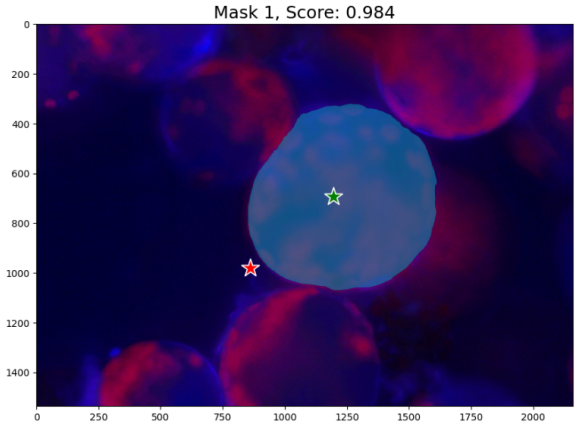
The dataset used in this research consists of plant cell microscopy images. It contains bright field images of plant cells. The data is spread over 5 days, there are 3 cell positions and there are 7 z-stack level for each position. Each bright field image has a corresponding fluorescent image. An example is shown in Figure 4. There are also

```
input_box = np.array([800, 323, 1592, 1060])
input_point = np.array([[1196, 692], [860, 980]])
input_label = np.array([1, 0])
```

(a) Segmentation prompt



(b) Points and boxes on Fluorescent images



(c) Generated segmentation mask

Fig. 2. Interactive Segmentation

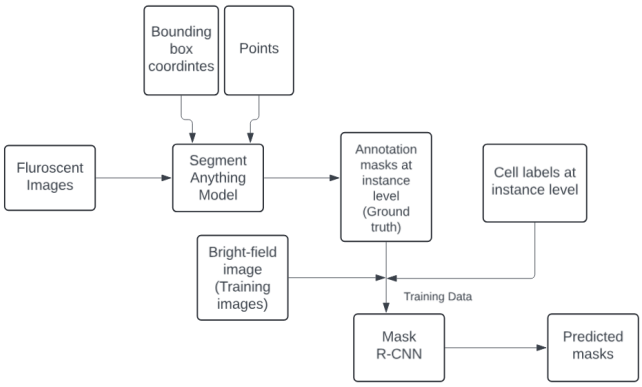
additional positions, which do not have data for all 5 days. These are used as unseen data for evaluation of the model.

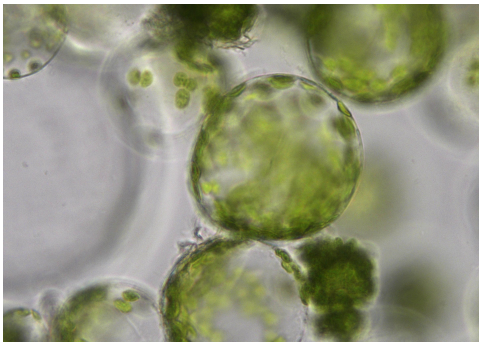
6.2 Creating segmentation masks

When creating the segmentation masks, there were two possible ways in which the z-stack levels can be used to generate the segmentation masks.

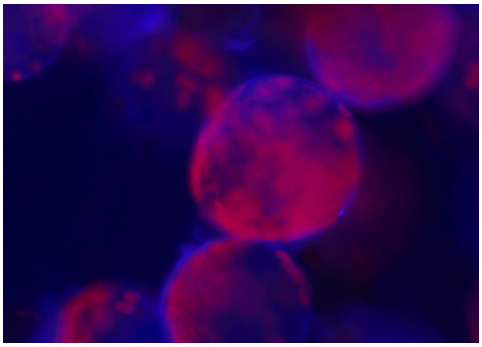
- Median of all the z-stacks: The median of each pixel intensity across different z-stack levels will be taken to combine all the z-stack levels. The segmentation masks will be created for each cell instance in the combined z-stack image.
- Creating segmentation masks for each cell visible in each z-stack: Segmentation mask will be created for each cell instance in each z-stack level image. The visible cell in the certain z-stack will be annotated. Once this is done, each

Fig. 3. Methodology





(a) Bright field image



(b) Fluorescent image

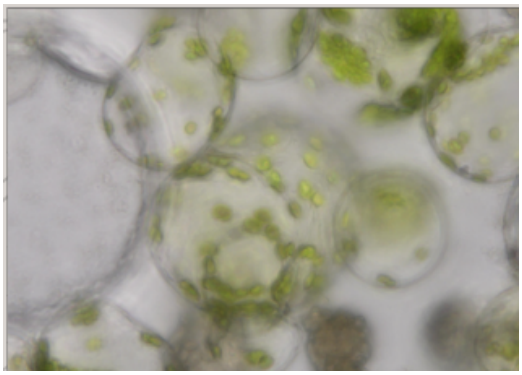
Fig. 4. Dataset

instance mask can then be combined to be a single mask dictionary, where the bounding box of each cell is mapped to its corresponding mask.

In order to decide which of these methods is more suitable for the research, a small experiment was conducted in which both the options were tried out. The dataset contained cell positions xy01, xy14 and xy20. For the experiment, the images of Day 1 were used. From the experiment, it was seen that for position xy14 and most images of xy20, creating segmentation mask for the combined median pixel image was more accurate. Hence, initially this was the chosen method for creating segmentation masks for all the images.

6.2.1 Challenge of the segmentation. During later stages of mask annotation, it was seen that this method did not work that well for images that had a lot of overlapping cells. This was especially seen in the position xy01. As shown in Figure 5, when taking the combined median pixel image of position xy01, the boundaries of the cells were not clear. Some images in position xy20 also had the same issue. Hence, it was very difficult to create segmentation masks for the cells in these images. Therefore, for these positions, the method was changed to the other option, where each z-stack level was inspected and the cells visible in that z-stack was annotated. Since the z-stack shows the boundary of the cells more clearly, more accurate segmentation masks could be created.

Fig. 5. Combined pixel image



Examples of generated segmentation masks for each position is shown in Figure 6.

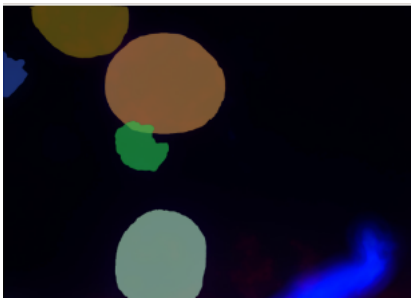
6.3 Label and bounding box generation

Since the Mask R-CNN model does object classification and segmentation, the ground truth should consist of segmentation masks, bounding boxes and labels for each cell. The labels would be 'Alive' or 'Dead' to denote alive and dead cells. These were classified based on whether the cells were visible in the fluorescent images or not. Cells that were visible in the bright field images but not in the fluorescent ones, were labelled dead and the rest were labelled alive. The bounding boxes were generated from the masks. Once the masks were generated by SAM, opencv-python library was used to get the bounding rectangle of the masks.

6.4 Training the model

Once all the segmentation masks were made, the next step in the research was to train the Mask R-CNN model. In order to train the Mask R-CNN model, the dataset was first split into two sets, the train set and the validation set. The images in position xy01 and xy20 were used as the train set and the images in the position xy14 was used as the validation set. The image details, the segmentation masks, bounding boxes and the labels were all stored in a JSON file in the COCO format. The segmentation section in the annotation

(a) xy01 Mask



(b) xy14 Mask

(c) xy20 Mask

Fig. 6. Generated masks

part contains the size of the mask, which is 512 x 512, the size of the mask and it also contains the segmentation mask encoded in RLE format. In order to evaluate the model, the model's prediction on unseen data, which includes unseen positions, will be visualised. Additionally, the model will also be evaluated on the following evaluation metrics:

- **Intersection over Union (IoU)**:

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|}$$

  where A is the predicted bounding box/mask and B is the ground truth bounding box/mask. IoU gives a good idea of the overlap between the ground truth and the predicted boxes and masks.

- **Mean Average Precision (mAP)**:

$$\text{mAP} = \frac{1}{N} \sum_{i=1}^{N} \text{AP}_i$$

  where N is the number of classes and AP_i is the average precision for the i-th class. mAP is calculated over different IoU thresholds between 0.5 and 1. mAP gives a good idea on the accuracy of the model.

- **F1 Score**:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

  where

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

- **Label Accuracy**:

$$\text{Label Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}$$

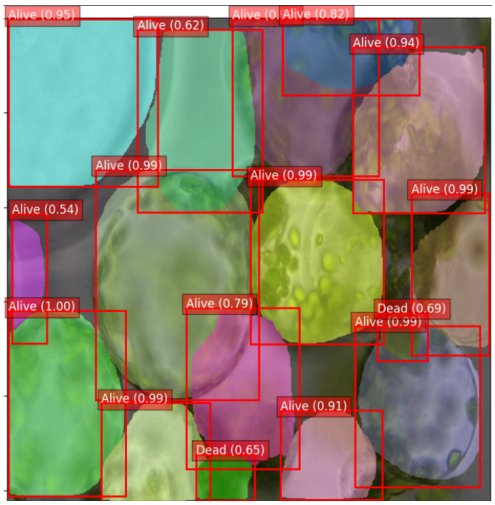
6.4.1 Data pre-processing. The images, in its original format, is too large to use as training data as it takes up a lot of GPU memory and it also slows down the training process. Hence, the images were resized to the dimension (512x512). Along with resizing the images, the masks and bounding boxes were also resized correspondingly. The images were then randomly flipped horizontally and vertically. The transformations were also applied to the ground truth bounding boxes and segmentation masks. The data that was initially stored as a json file was transformed to a CoCo Dataset and DataLoader was used in order to split the data into batches.

6.4.2 Training. Once the training data is defined, the model is initiated with pre-trained COCO weights and the ResNet50 fpn backbone. The ResNet50 is a deep residual network that consists of residual blocks, convolutional layers, pooling layers and fully connected layers that allows for training deeper networks for object detection[9]. Feature Pyramid Network (FPN) helps boost the capability of the ResNet50 backbone by creating a multi-scale feature maps that help detecting objects of different scales[11]. Combining ResNet50 with FPN helps in creating multi-feature maps that can then be used by the head, the region proposal network (RPN). The RPN takes in the extracted features and predicts whether an object is present in a region or not. ROI align is then used to extract fixed-size feature maps that are then fed to the object detection head and the segmentation head. The Adam optimizer was used in order to have a stochastic gradient descent that adapts the learning rates for each parameter. The initial learning rate was set to 0.0001. In order to adjust the learning rate during training, ReduceLROnPlateau learning rate scheduler was used. GradScaler was also used for mixed precision training to achieve performance speed up and maintain the accuracy. The loss function used is the default loss function for Mask R-CNN model, which is a multi-class loss function that includes classification loss, bounding box regression loss and mask loss. Classification loss is calculated using the cross-entropy loss, bounding box regression loss uses the smooth L1 loss function and the mask loss is calculated using the binary cross-entropy loss.

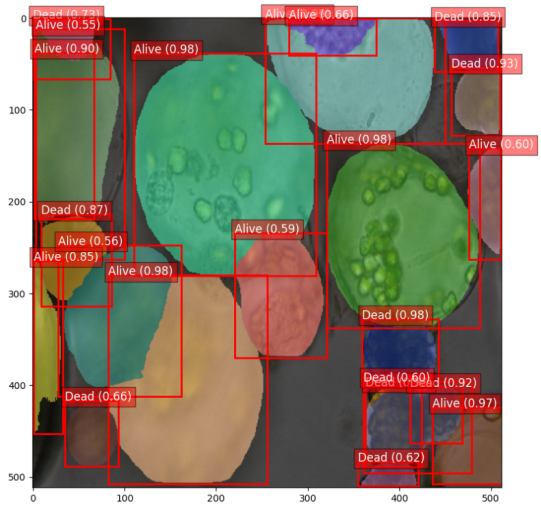
7 RESULTS

7.1 Prediction results

Once the model was trained, the trained model was used to predict some segmentation masks on unseen data and unseen cell positions. The results are shown in Figure 7.



(a) Prediction for day1c1t1xy15z20



(b) Prediction for day5c1t6xy37z14

Fig. 7. Predicted results

As seen in the image, most of the cells in the image are detected and the labels for these are predicted correctly. However, there are also false defections. There are some masks in areas on the image, where there are no cells and some masks where only part of the cell is included in the mask. This trend is generally seen in the cells that overlap. One of the reasons for there to be segmentation masks for cells not in the specific z-stack could be because of how the training data is structured. In the training data, for each z-stack, the corresponding segmentation mask was made to the combined z-stack pixel image. These would also include cells that are not clearly seen in the specific z-stack. Hence, this could be why the predicted results show masks for cells not in the specific z-stack level. The performance of the model can be better assessed using evaluation metrics.

7.2 Evaluation metrics

The average of all the evaluation metrics over all the epochs are shown in Table 1 for the bounding boxes and labels. The results for the masks are shown in Table 2.

Table 1. Bounding box and label metrics

mAP (%)	f1 Score (%)	IoU score (%)	label accuracy (%)
38.2	27.3	85.5	84.8

Table 2. Segmentation masks metrics

mAP (%)	f1 Score (%)	IoU score (%)
0.098	0.86	87.02

The train and validation loss was also plotted and the result is shown in Figure 8. The decreasing trend in the training loss indicates that the model is learning well. However, the fluctuating validation loss suggests that the model is not predicting well for unseen data. The validation loss also increases at the end, where the training loss keeps decreasing, which suggests that the model is over fitting to the dataset. The visualised predicted results also corresponds to this pattern as there are more cells segmentation masks being predicted than cells in the images.

Fig. 8. Train and validation loss

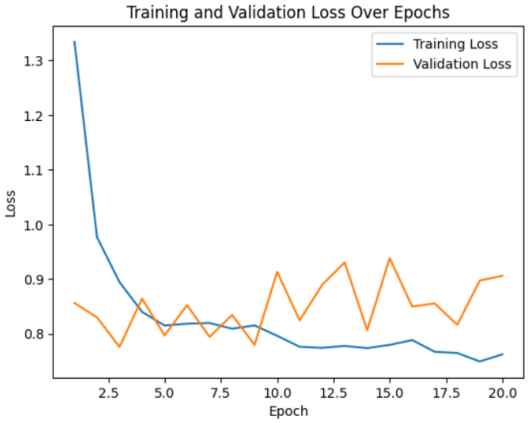


Table 3. Model Performance Comparison Across Z-levels

Z-level	mAP (%)		f1 (%)		IoU (%)		label accuracy (%)
	bbox	masks	bbox	masks	bbox	masks	
z12	31.5	0.007	23.5	0.53	83.3	84.4	83.4
z14	35.2	0.004	25.8	0.30	84.51	85.40	82.3
z16	41.0	0.310	30.5	2.70	85.98	86.75	820
z18	39.7	0.106	35.3	1.40	85.43	87.64	80.6
z20	37.6	0.166	40.0	3.00	85.30	87.54	81.0
z22	32.3	0.050	36.5	1.60	84.93	86,26	83.0
z24	25.2	0.181	37.1	2.10	84.24	84.41	84.3

7.3 Evaluation metrics across z-stack levels

Across the different z-stack levels the bounding box metric values shows a pattern similar to a bell curve. The initial z-stack levels and the last z-stacks have a comparatively lower mAP and f1 score whereas the z-stack levels in between have a higher value in these metrics. This pattern is expected as the blur level tends to decrease as we move towards the z-stack levels 18, 20 and it starts to again move out of focus at z-stack level 22,24. It will be easier for the model to predict the bounding boxes at z-18 and z-20 as the contours of the cell will be more apparent in these focus levels. The peak at z-16 might be due to the fact that only very few cells are visible at this z-stack level and hence not a lot of bounding boxes or masks will be made at this focal length. At z-12 and z-14, almost none of the cells are visible and at z-22 and z-24, cells are slightly visible, however, the focus starts fading resulting in the pattern observed.

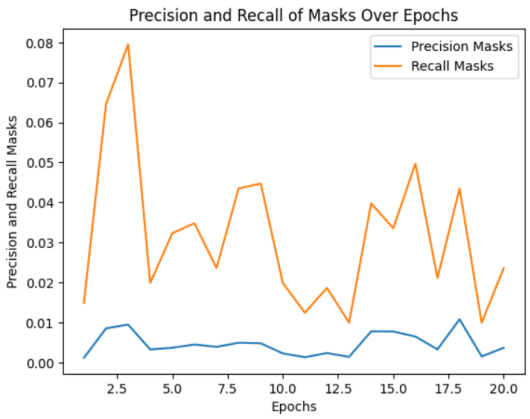
The performance of the masks across the different z-stack level also shows a peak at z-16 and z-20, especially for f1 score. The mAP for the masks also suggests a similar trend with peaks at z-16 and z-20. This also corresponds to the pattern seen with the bounding boxes. For the masks, there is a significant dip at z-18 and this might because at z-18 we start seeing some boundaries, however, these are not fully clear and hence when the model tries to create segmentation masks for these cells, it produces inaccurate masks.

7.4 Explanation of the evaluation metrics

The precision and recall for the bounding boxes shows that the recall is higher than the precision. This indicates that the model is detecting cells, however a lot of false cells are also being detected. The precision and recall for the masks, as seen in Figure 9, also suggests the same. From visualising the predictions, it is also seen that more masks are being created than the number of cells. In some cases, non-cell objects are also being identified as cells. This also results in the mAP of the masks being very low.

One of the reasons for the poor evaluation results for the masks could be the manual segmentation masks created. Since the ground truth was manually created, there is a possible chance that there might be differences in the segmentation mask's precision. For some of the cells, especially in the case of overlapping boundaries, it was not possible to segment it accurately and in these cases, the best possible mask was made. As also seen in the model's performance over the different z-stack levels, the level of blur causes the contrast

Fig. 9. Precision and recall of



between the background and the cells to reduce, which leads to the model not being able to segment the cells properly. This can also cause the model to not be able to segment adjacent cells accurately in out-of-focus images. The varying blur level can interfere with the model's ability to learn consistent features. The visualisation of the prediction does suggest that the model predicts the cells well. However the mask mAP is low, this could be due to the validation set used. The validation set used to evaluate the model were the images at the position xy14. As mentioned in the methodology, for position xy14, the ground truth masks were created on the combined cell images. The fluorescent images in this position had poor visibility of any cells. Only very few cells were visible and hence only these ones were segmented during the creation of the ground truth data. However, when looking at each z-stack level, there are more cells in the images than there exists the segmentation masks in the ground truth. The model is trained on position xy01 and xy20 and for these positions, the ground truth was created by going through each z-stack level as the combined images for these positions included all the cells and the cells were visible in the combined image but the cell walls were not clearly distinguishable in the combined image. Due to the model being trained on these positions, the model predicts even the less visible cells in position xy14, when it is fed in as the validation set. This causes there to be more predicted cell segmentation masks than the number of segmentation masks in the ground truth. Hence, the precision reduces as the number of FP increases. This causes the mAP value for the mask to be very low. The comparatively higher recall suggests that out of all the ground truth masks, most of the masks are predicted by the model. This is also substantiated by the IoU score as it suggests that the masks that are being created do overlap to a great extent with the ground truth masks and bounding boxes.

7.5 Comparison to baseline model

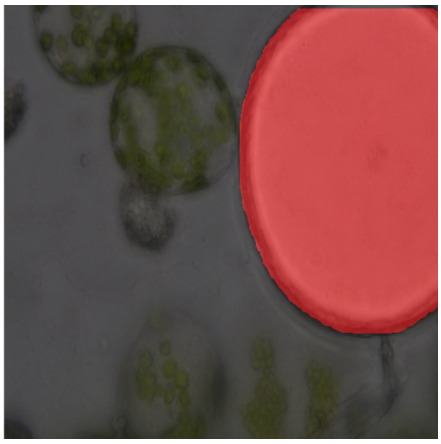
In order to further evaluate the performance of the model, we can compare it with Yolov5-Seg. Yolov5-Seg is an instance segmentation model that has been built on top of the standard Yolov5 architecture, which is primarily used for object detection. When predicting the validation set using Yolov5-Seg, we get the following evaluation

metrics results as shown in Table 4. As seen with the IoU score, the model performs poorly in detecting and segmenting the cells. The f1 score and accuracy metrics do have a higher value than the metrics of the Mask R-CNN model. However, this is due to the difference in the number of predictions made by Yolov5. Since both these metrics leverage the number of predictions to a great extent, there being less predictions in Yolov5, as seen in the image, leads to the seen performance. The IoU scores indicates that the predictions made by this baseline model does not segment the cells well as there is minimal overlap with the ground truth masks. This is also substantiated by Figure 10, where the detected component in the image is not a cell. Comparatively, our model performs better segmentation as seen in Figure 11. The trained Mask R-CNN model has a segmentation IoU score of 87.02%. The label accuracy of YoLov5 is 0% as the elements in the image were never identified as cells, due to the model not having seen cell images in its training data. This comparison between the two models provides great insight into the need for instance segmentation models trained specifically on microscopic cell images for effective cell segmentation.

Table 4. Evaluation metrics for Yolov5 on microscopic cell images

IoU score (%)	f1 Score (%)	accuracy (%)	label accuracy (%)
5.8	6.5	10.4	0

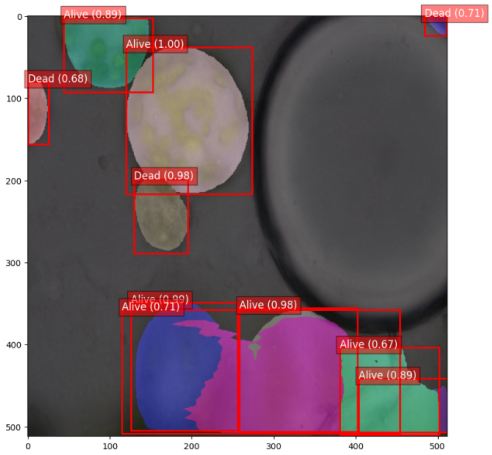
Fig. 10. Yolov5 prediction - day3c1t3xy14



8 CONCLUSION

In conclusion, answering the first research question involves using SAM to effectively generate segmentation masks for the images. This process requires a great amount of manual work as we need to specify prompts for each cell in an image. These prompts would need to be further adjusted to accurately segment each cell. Some cells might just take one point coordinate and bounding box coordinates to accurately segment the cell, whereas some might take multiple point coordinates to segment the boundaries correctly. Additionally, in the cases of overlapping cell boundaries, the creation

Fig. 11. Mask R-CNN prediction - day3c1t3xy14



of segmentation masks were challenging. Hence, SAM can be effectively used to create segmentation masks to serve as ground truth to cell instance segmentation models, given that the z-stack images distinctly outlines the cell walls of each cell. However, it is not an efficient method due to its resource intensive nature.

Answering the second research question, the generated segmentation masks does serve as an basis for training a cell segmentation model. However, it needs to be developed further to train a model that predicts very accurate segmentation masks. The model detects cells and it also creates segmentation masks for it. However, there are also cells detected that are not in the image provided. Additionally, some of the created masks have inaccurate cell wall boundaries detected. The reasons for these inaccuracies are discussed in the explanation of the evaluation metrics. The main reason for this could be the quality of the ground truth and the size of the dataset. The dataset might not be large enough for the model to generalise the segmentation mask patterns. Hence, there would need to be further experiment to answer this question better.

9 FUTURE WORK

In order to continue this research, the main aim would be to improve the performance of the model. One way to do this would be to work on the ground truth segmentation masks. The already existing ones can be reviewed by an expert and made better. In addition to this, expanding the dataset would enhance the performance of the model. Hence, more ground truth segmentation masks need to be created. Although this is a time consuming task, more data would be beneficial for the performance of the model, especially to prevent over fitting. There could be a team of people that work on creating the segmentation masks, which would make the task more feasible. Additionally, other models could also be trained to do the same. U-Net architecture could be modified for it to perform instance segmentation. Since it is a biomedical segmentation model, it could perform better on microscopic cell images.

REFERENCES

[1] Encord. 2024. The Ultimate Guide to Instance Segmentation in Computer Vision. https://encord.com/blog/instance-segmentation-guide-computer-vision/. Accessed: 2024-06-20.

[2] Alberto Garcia-Garcia et al. 2018. A survey on deep learning techniques for image and video semantic segmentation. (2018).

[3] Christoffer Edlund et al. 2021. LIVECell—A large-scale dataset for label-free live cell segmentation. (2021).

[4] Juan C. Caicedo et al. 2019. Evaluation of Deep Learning Strategies for NucleusSegmentation in Fluorescence Images. *Cytometry: Journal of Quantitative Cell Science* (2019).

[5] Jifan Zhang et al. 2024. Algorithm Selection for Deep Active Learning with Imbalanced Datasets. (2024).

[6] Noah F.GreenWald et al. 2021. Whole-cell segmentation of tissue images with human-level performance using large-scale data annotation and deep learning. *nature biotechnology* (2021).

[7] Tingxi Wen et al. 2023. Review of research on the instance segmentation of cell images. *communications biology* (2023).

[8] Lucia Zheng Grace Kim, Will Song. 2022. Deep Learning for Cell Segmentation. (2022).

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. *arXiv preprint arXiv:1512.03385* (2015). https://arxiv.org/pdf/1512.03385

[10] Piotr Dollar Ross Girshick Kaiming He, Georgia Gkioxari. 2018. Mask R-CNN. (2018).

[11] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. 2017. Feature Pyramid Networks for Object Detection. *arXiv preprint arXiv:1612.03144* (2017). https://arxiv.org/pdf/1612.03144

[12] MathWorks. 2024. Semantic Segmentation. https://nl.mathworks.com/solutions/image-video-processing/semantic-segmentation.html. Accessed: 2024-06-20.

[13] Timothy Solberg Miguel Romero, Yannet Interian and Gilmer Valdes. 2021. Training Deep Learning models with small datasets. (2021).

[14] Mohammad Owasis, R Nithya, Ghaffar Nia Nafiseh, Kaplanoglu Erkan, and Nasab Ahad. 2021. Artificial intelligence in disease diagnosis: a systematic literature review, synthesizing framework and future research agenda. *Journal of Ambient Intelligence and Humanized Computing* 12, 12 (2021), 9945–9974. https://doi.org/10.1007/s12652-021-03612-z

[15] Ji Yu Prem Shreshta, Nicholas Kuang. 2023. Efficient end-to-end learning for cell segmentation with machine generated weak annotations. *communications biology* (2023).

[16] Karen Simonyan and Andrew Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR* abs/1405.0312 (2014). arXiv:1405.0312 http://arxiv.org/abs/1405.0312

A ADDITIONAL REMARKS

During the preparation of this work the author(s) used no artificial intelligence tools.