Table of Contents

Table of Contents
1. Introduction
    1.1 Overview
    1.2 Challenges of identifying Multiple Myeloma Plasma Cells
    1.3 Methods of Segmentation of Multiple Myeloma Plasma Cells
        Process of Automatic Segmentation
        Nucleus Segmentation
        Cytoplasm Segmentation
    1.4 Utilizing Pytorch and Github
2. Research Dataset
3. Project Statement
4. Project Timeline
        Project Timeline Outline
5. Image Annotation
    5.1 Labeling and Annotation via LabelMe
6. Neural Networking via Pytorch
    6.1 Utilizing Dataloaders for NN training
Checkpoint 1
7. Segmentation Process via Pytorch
Checkpoint 2
8. Results
Reflections:
References:

---

1. Introduction

1.1 Overview

Our research project involves exploring methods of cell segmentation in order to identify cells that may suffer certain illnesses/diseases such as cancer. The manual

process of identifying these cells is time-consuming. Many implementations of segmentation can be used to train a model to automatically identify the cells of interest. Utilizing PyTorch to train a Convolutional Neural Network, we would train a model by providing data of labeled microscopic images. The goal of this project is to create a model that automatically identifies the cells of interest for patients that suffer from Multiple Myeloma. This model will be tested, validated, and then analyzed in its ability to accurately label the cells of interest (malignant cells) in the provided dataset for our study.

1.2 Challenges of identifying Multiple Myeloma Plasma Cells

Multiple myeloma is a type of cancer that affects the plasma cells in the bone marrow. As a form of blood cancer, it is characterized by the uncontrollable growth of cells that spread to the surrounding tissue cells. The manual process of identifying cells of interest through a microscope without the aid of computational algorithms is labor-intensive, which has prompted scientists to seek alternative methods for automatically identifying Multiple Myeloma and other diseases. . An automatic segmentation of cells from microscope images involves creating a computational algorithm that is able to detect the cell of interest and segment and label the background, cell nucleus and cell cytoplasm. Using automatic segmentation methods, scientists are able to discern myeloma cells separated from regular plasma cells. Automatic segmentations are usually indicated with colored borders around the myeloma cells, this helps detect the presence of myeloma cells and diagnose the appropriate disease. This process involves the usage of pixel wise segmentation or object detection.

Myeloma cells often have irregular shapes, exhibit larger sizes, and have an abnormal nucleus. In order to analyze samples of plasma cells, many pathologists consider the PCP (Plasma Cell Percentage) as a major factor in multiple myeloma. The manual process of identifying and determining PCP is time-consuming, which has led to scientists relying on faster and automated methods of analyzing these cells. Cell

segmentation algorithms may allow for a quicker and more efficient way at analyzing the cell.

This leads to a major challenge that incorporates algorithms to identify cells for multiple myeloma. Firstly, the cell nucleus and cytoplasm have to be identified. This could be addressed through the usage of instance segmentation (object detection and pixel-wise detection). These methods then would require a scan or photo of the plasma cells, usually taken under a microscope through advanced imaging. It must also address the possible issue of certain cells having their appearances distributed by other structures nearby.

1.3 Existing methods of Segmentation

Process of Automatic Segmentation

Automatic segmentation methods utilize eight basic steps during image analysis:

A. *Image Acquisition*
   Involves the acquisition of microscopic images containing the nuclei, which usually includes the fluorescent staining of the cell using fluorescent dyes/proteins that absorb light at specific wavelengths and emit the light of a longer wavelength. This makes it easier to identify the structure that is being analyzed in microscopic imaging.
B. *Preprocessing*
   Preprocessing involves enhancing the quality of the image and reduce noise. As such, the most common methods of preprocessing involve noise reduction (unwanted noise shot under low light conditions), contrast enhancement (increasing or decreasing the contrast of the image) , and image normalization (changes to the range of pixel intensity values).
C. *Thresholding*
   In this step, thresholding is used to separate foreground objects from the background of the image. This is accomplished by defining a threshold value,

those above the threshold are classified as the foreground or object of interest, and those below are classified into the background.

D. *Segmentation*
   After thresholding, an algorithm would be used to delineate cell structures from images. Machine learning-based approaches, especially deep learning models, are trained to learn from annotated training data.
E. *Post-processing*
   Post-processing is one of the crucial steps to improve the accuracy of the results. This may involve morphological filtering that would remove small objects and use contour smoothing techniques to refine the segmentation boundaries.
F. *Validation*
   The validation step is another crucial step to assess the accuracy of the segmentation process. Manually annotated ground truth data, recall, and F1-score would measure the segmentation algorithm's performance.
G. *Analysis*
   After all other steps, analysis is performed to extract quantitative data about the cell morphology and behavior.

Nucleus Segmentation

Nucleus Segmentation is the process of image analysis that aims to identify nuclei in microscopic images. This strategy aims to use the detected nucleus as a supervisory signal (aids of identification of individual cells) for the cytoplasm instance to handle overlapping areas. Scientists who utilize this method train a U-net model using a nucleus annotation mask, which is just the digital outline on the boundaries of cell nuclei.

The process of Nucleus Segmentation is critical to understanding more details of cellular activities and the underlying biology. Traditional methods usually use intensity-based thresholding, which leverages variations in pixel intensities to differentiate nucleus regions from the background. Several advanced approaches use machine learning models, such as deep neural networks, to automatically learn and detect the nucleus, greatly enhancing the segmentation accuracy. Nucleus segmentation

facilitates basic cell biology research and also possesses significant potential for applications in medical diagnosis.

Cytoplasm Segmentation

Similarly to Nucleus Segmentation, Cytoplasm Segmentation also involves outlining a specific portion of the image, namely the cytoplasm. The cytoplasm is identified by manual annotation or a trained model, highlighted and included as a layer in the training data. In the manual process, it is common practice to identify and then outline the cell's cytoplasm (the gel-like area surrounding the outside of the nucleus). This segmentation method provides precise identification and delineation of cytoplasmic areas within microscopic images. In certain applications of Cytoplasm Segmentation, the algorithm usually utilizes an input image of the cell, a GT mask (Ground Truth Mask) that reveals certain images and discerns them from each other, usually using different colored imaging. The three things that are discerned are: the cell nucleus, the cytoplasm, and the outlining region.

The cytoplasm segmentation method is pivotal to the field of image analysis and cellular biology. Algorithms such as convolutional neural networks (CNNs) play a crucial role, creating an automated feature extraction and pattern recognition for improved segmentation accuracy. Additionally, region based algorithms such as active contours and region growing, also contribute by incorporating spatial information to define boundaries of cytoplasmic regions. The choice of a specific method depends on the characteristics of the microscope images and the level of segmentation precision. Looking further into cytoplasm segmentation helps enhance scientists' understanding of cellular dynamics while providing various biomedical applications, such as detecting cancerous cells.

1.4 Utilizing Pytorch and Github

PyTorch allows a platform for implementing segmentation algorithms due to its ease of use, flexibility and computation capabilities. Image segmentation requires various advanced computational techniques which can be handled using a deep learning framework. There are various neural network modules - modules of neural networks that work together to train and develop neural network models - and optimization algorithms that are tailored to create segmentation models in relation to identifying the characteristics of multiple myeloma plasma cells.

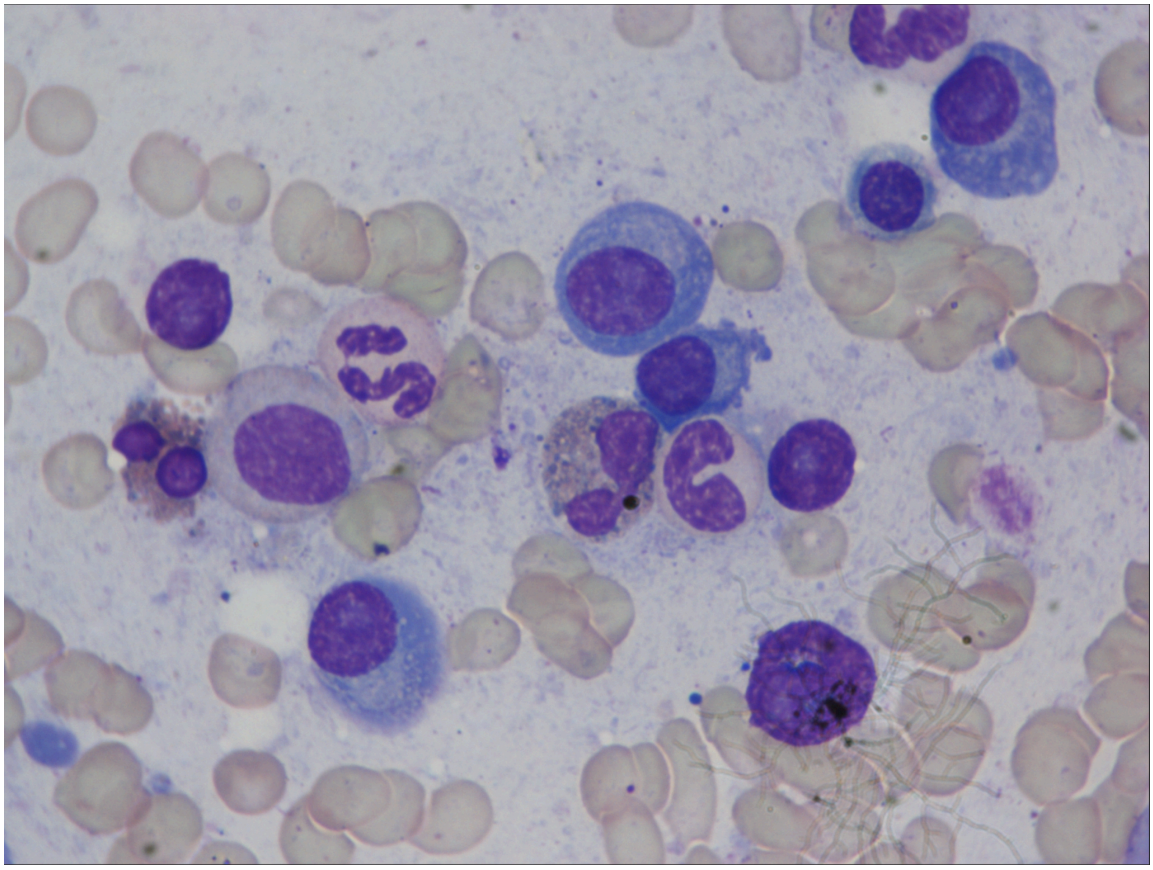
Through the usage of PyTorch, one can train a convolutional neural network (CNN) model that learns the features of multiple myeloma cells from any given annotated training data. There are various intuitive Application Programming interfaces (APis) to build and train a neural network, allowing for model optimization.

In order for our group to functionally work together to build this model, a shared repository where updates to the model can be tracked and pushed into the main branch should be utilized. GitHub, an example of a development platform that allows for this, will be used for this research project. This will give our team access to PyTorch via Github Codespaces, an IDE found on the repository (a place that stores code, files and revision history) of Github.

After the model has been trained, it can be used to automatically annotate multiple myeloma plasma cells. In the following section, we will discuss the specifics of our dataset for this project.

2. Dataset

The challenge presented for our research requires us to use images for the segmentation process aftermanual annotation. These images are in the bmp file format with a size of 2560 x 1920 pixels. The majority of the data consists of images in this format and size. An example image from the dataset is shown in Figure 1.



- Figure 1. Microscopic image captured from a bone marrow aspirate slide of patients with diagnosed multiple myeloma, a type of white blood cell cancer.

The test dataset consists of a total of 277 images that are all from patients that suffer from multiple myeloma. Due to time constraints, a short sample (10 bmp files) of these images will be considered in creating our model. Since this is the *final test phase* of the dataset provided, we will be tasked with creating labels that will help us identify cells with multiple myeloma. First, we will have to investigate methods that are capable of detecting three distinct features within each figure: the Background, Cytoplasm and Nucleus. To do this we will need to create labels for the dataset with 0 as the background, 1 as the cytoplasm and 2 as the nucleus. This will be one of our first steps in the segmentation process.

3. Project Statement

Our research is all about generating new methods within the cell segmentation process and figuring out how well we can use the cell segmentation method to spot indications of multiple myeloma in plasma cells. The main goal is to identify parts of the microscopic images, labeling the background, cytoplasm and nucleus. Only then could we be able to segment the cell of interest in the dataset provided to us. Using medical imaging of plasma cells, we will use concepts such as object detection and pixel-wise detection to achieve this. We will produce a model that identifies these areas of the cells through cell nuclei segmentation and cytoplasm segmentation. We will also incorporate deep learning tools such as PyTorch to help identify the labels for each microscopic image. This model will be used to submit for our final project and results will be demonstrated through a presentation.

Why this matters in the world of computational science is because of the cool things we can do with object detection and different ways of breaking down images. Thanks to these techniques, scientists have been able to pick up on oddities and problems in scans and other images that might not be obvious to the naked eye. These algorithms have been a game-changer in the medical field, helping us solve problems that were a lot trickier in the past.

4. Project Timeline

This project lasts our full semester, spanning a total of 12 weeks, and some of our findings will be used to contribute to our final project in “Data Science meets Health

Science” under Professor Olmo Zavala Romero. In the first week, our plan is to dive into research, trying to get a good grip on the data, the problem and expectations that come with the project. As we move into the second week, we'll be gathering with teammates, brainstorming and figuring out the initial topics for our presentation. Once we hit the third week, we will be solely focusing on our first project presentation. We'll be getting into the root of our challenge and the data, downloading the dataset, and making sure everyone understands what we are dealing with. Moving on to the fourth week, we will go into the real work, researching and figuring out how to tackle the challenge within our group. Week five is when we are starting to put the segmentation into action for the challenge. This rolls into the sixth week, where we fine-tune our segmentation efforts and start brainstorming our second project presentation. Week seven will consist of refining our project and making sure our second presentation is polished, showcasing our skills in handling the data, and explaining how we're measuring our success. Taking a break in week eight is unavoidable due to prior planning outside of academia. On week nine, we will be presenting our progress to Professor Olmo, getting some feedback, and making any necessary tweaks. Week ten, that's when we take the first steps into the segmentation process. Week eleven is where we will turn our attention to creating the proposed model for our research project. In the last week before the final project presentation, we will work on our third and final presentation, breaking down the details of our proposed model. The final product will be ready for presenting, in which we will detail our model and show our results of the research challenge.

Project Timeline Outline

Project	Project	Task Name	Planned	Planned	Planned	Percent

Name	Status		Start Date	Finish Data	Duration	Complete
Practicum	Open	Topic and Challenge Research	Jan 22, 2024	Jan 26,2024	1 week	100%
Practicum	Open	Initial Presentation Topics	Jan 29, 2024	Feb 2, 2024	1 week	10%
Practicum	Open	Presentation and Dataset Collection	Feb 5, 2024	Feb 9, 2024	1 week	0
Practicum	Open	Brainstorm and Research Strategies	Feb 12, 2024	Feb 16, 2024	1 week	0
Practicum	Open	Start Implementation/ Report Draft 1	Feb 19. 2024	Feb 23, 2024	1 week	0
Practicum	Open	Continue Implementation/ Start Presentation 2	Feb 26, 2024	Mar 1, 2024	1 week	0
Practicum	Open	Finalize Presentation 2	Mar 4, 2024	Mar 15, 2024	2 weeks	0
Practicum	Open	Presentation 2 and Report Draft 2	Mar 18, 2024	Mar 22, 2024	1 week	0
Practicum	Open	Initial Segmentation Process	Mar 25, 2024	Mar 29, 2024	1 week	0
Practicum	Open	Generate Proposed Model/Finalize Segmentation	Apr 2, 2024	Apr 12, 2024	2 week	0
Practicum	Open	Final Report/Oral	Apr 15,	Apr 19,	1 week	0

		Presentation	2024	2024		

---

5. Image Annotation

5.1 Labeling and Annotation via LabelMe

After obtaining our dataset of 10 microscopic images depicting multiple myeloma cells, our first step is to meticulously annotate and label each image to provide ground truth data for training our model. To accomplish this task, we employ the labeling software LabelMe, which offers advanced features and flexibility in annotating complex structures such as cellular components.

Using LabelMe, we outlined and annotated each individual myeloma cell within the microscopic images, ensuring precise delineation of boundaries and accurate labeling of cell types. This annotation process is crucial for training our model to accurately segment and identify myeloma cells in unseen images.

6. Neural Networking via Pytorch

6.1 Utilizing Dataloaders for NN training

As we progress from annotation to model creation, we leverage the power of PyTorch, a popular deep learning framework, to develop our segmentation model. PyTorch provides a flexible and efficient platform for building and training neural network models, making it an ideal choice for our project. With PyTorch, we load our annotated dataset and construct custom data loaders to efficiently feed the data into our segmentation model during training. These data loaders handle tasks such as data augmentation, batching, and shuffling, ensuring smooth and effective training of our model on the annotated dataset.

Diving deeper into the model development process, our focus goes towards implementing the segmentation techniques within PyTorch. We will explore various neural network architectures, such as U-Net, FCN, and DeepLab, to find the most suitable approach for segmenting myeloma cells with high accuracy and efficiency. Additionally, we will experiment with different loss functions, optimization algorithms, and hyperparameters to fine-tune our model and optimize its performance on the task of myeloma cell segmentation.

In the subsequent phases of our project, we plan to evaluate the performance of our segmentation model on unseen data and assess its generalization capabilities. We want to be able to have our trained model help real-world scenarios to assist medical professionals in accurately identifying and analyzing myeloma cells from microscopic images.

7. Segmentation Process via Pytorch

Since the original format for the images in the dataset were bmp files, we needed a way to convert the annotated png files back into the appropriate files. Not only that, but we needed to be able to create folders for both the annotated png files and bmp files. The

next part was trickier, which was trying to replicate the black squares back onto the bmp files. We saved the positions of each of the black squares in the png images while converting it over to the bmp file.

8. Results

After setting up and processing the dataset, we proceeded to define our model and training set. Originally, we planned to train our data using nine images. However, we decided to reduce this to just three images for training, while the remaining nine were allocated for validation. Unfortunately, we couldn't generate images from the trained model into this report due to time constraints. Nevertheless, the provided depiction gives a basic idea of what the segmentation should resemble. This image is one of the three used for training the model.

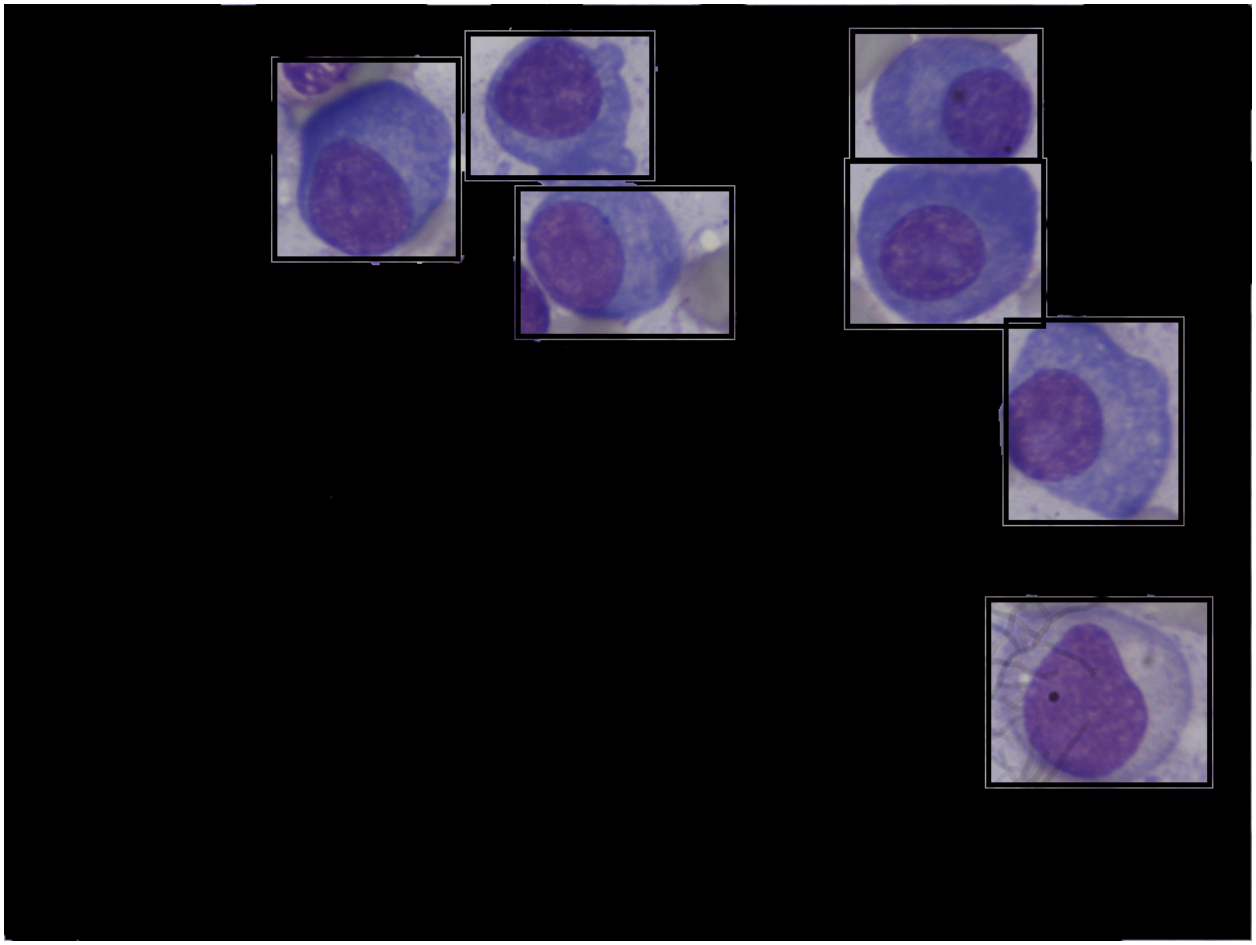


Figure 8. A glance at a partially segmented image of multiple myeloma cells giving a look into the training set, in comparison with the expected output of the validation set.

Our model underwent ten epochs during training, each lasting approximately one minute to complete. Throughout these epochs, we observed a consistent decline in the loss function, indicating that the model was effectively learning and improving over time. This gradual decrease in the loss value is indicative of the model's training progress, demonstrating its ability to optimize and refine its predictions iteratively.

```
Epoch [1/10], Loss: 0.6901
2024-04-22 22:09:33.620439: W tensorflow/core/framework/local_rendezvous.cc:404] Local rendezvous is aborting with status: OUT_OF_RANGE: End of sequence
Epoch [2/10], Loss: 0.6768
2024-04-22 22:09:43.016220: W tensorflow/core/framework/local_rendezvous.cc:404] Local rendezvous is aborting with status: OUT_OF_RANGE: End of sequence
Epoch [3/10], Loss: 0.6550
2024-04-22 22:09:52.452138: W tensorflow/core/framework/local_rendezvous.cc:404] Local rendezvous is aborting with status: OUT_OF_RANGE: End of sequence
Epoch [4/10], Loss: 0.6193
2024-04-22 22:10:01.756181: W tensorflow/core/framework/local_rendezvous.cc:404] Local rendezvous is aborting with status: OUT_OF_RANGE: End of sequence
Epoch [5/10], Loss: 0.5613
2024-04-22 22:10:11.092409: W tensorflow/core/framework/local_rendezvous.cc:404] Local rendezvous is aborting with status: OUT_OF_RANGE: End of sequence
Epoch [6/10], Loss: 0.4733
2024-04-22 22:10:20.524797: W tensorflow/core/framework/local_rendezvous.cc:404] Local rendezvous is aborting with status: OUT_OF_RANGE: End of sequence
Epoch [7/10], Loss: 0.3551
2024-04-22 22:10:29.902745: W tensorflow/core/framework/local_rendezvous.cc:404] Local rendezvous is aborting with status: OUT_OF_RANGE: End of sequence
Epoch [8/10], Loss: 0.2324
2024-04-22 22:10:39.371195: W tensorflow/core/framework/local_rendezvous.cc:404] Local rendezvous is aborting with status: OUT_OF_RANGE: End of sequence
Epoch [9/10], Loss: 0.1477
2024-04-22 22:10:48.778745: W tensorflow/core/framework/local_rendezvous.cc:404] Local rendezvous is aborting with status: OUT_OF_RANGE: End of sequence
Epoch [10/10], Loss: 0.1112
Training complete!
```

Figure 9. The results of the training through ten epochs. The general loss function trend can be observed as well as it decreases with each epoch signifying the model learning.

Reflections:

In short, the model shows potential for effectively training data to accurately segment areas of medical images for cell detection. However, several challenges emerged during the project's execution. A limitation for the project was the relatively short time frame of three to four months compared to other researchers who had a full two-year span to tackle these problems. Additionally, our model encountered difficulties in generating images that displayed the segmented areas, hindering our ability to visualize and validate its performance effectively.

Moreover, a significant amount of our allocated time was dedicated to manual image annotation and with the implementation of our segmentation. Moving forward, addressing these challenges is critical. Priority should be given to further working on the segmentation approach rather than spending time on manual annotation tasks. Additionally, we need to judge our model for potential biases that might have influenced the interpretation of the loss function, potentially masking areas where the model failed to perform as expected. These reflections outline the importance of utilizing a more strategic approach in future, focusing on optimizing methodology, mitigating biases, and maximizing the utility of available resources.

References:

1. Afshin Bozorgpour, Reza Azad, Eman Showkatian, Alaa Sulaiman, "Multi-scale Regional Attention Deeplab3+: Multiple Myeloma Plasma Cells Segmentation in Microscopic Images," 2021.

2. Salvi M, Michielli N, Meiburger KM, et al. "cyto-Knet: An instance segmentation approach for multiple myeloma plasma cells using conditional kernels," International Journal of Imaging Systems and Technology, 2024; 34(1):e22984. doi:10.1002/ima.22984.

3. Wang A., Zhang Q., Han Y. et al. "A novel deep learning-based 3D cell segmentation framework for future image-based disease detection," Scientific Reports, 2022; 12:342. doi:10.1038/s41598-021-04048-3.

4. Chen, L. C., Papandreou, G., Schroff, F., & Adam, H. (2018). Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 801-818). doi: 10.1007/978-3-030-01264-9_49.

5. Isensee, F., Jaeger, P. F., Kohl, S. A. A., Petersen, J., & Maier-Hein, K. H. (2021). nnU-Net: A self-configuring method for deep learning-based biomedical image segmentation. Nature Methods, 18(2), 203-211. doi: 10.1038/s41592-020-01008-z.

6. SEGPC-2021 - Grand challenge for Cell Segmentation for Multiple Myeloma Plasma Cells. Grand Challenge. Available at: https://segpc-2021.grand-challenge.org/.

7. CSAILVision, "LabelMeAnnotationTool," GitHub, https://github.com/CSAILVision/LabelMeAnnotationTool.

8. PyTorch, "PyTorch," https://pytorch.org/.