QUANTITATIVE ANALYSIS OF THP1 CELL CONFLUENCY AND PROLIFERATION UNDER TEMPORAL AND PHARMACOLOGICAL CONDITIONS USING DEEP LEARNING TECHNIQUES

A THESIS SUBMITTED TO
THE FACULTY OF ARCHITECTURE AND ENGINEERING
OF
EPOKA UNIVERSITY

By

ALBAN XHEPI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

JUNE, 2024

Approval sheet of the Thesis

This is to certify that we have read this thesis entitled **“Quantitative Analysis of THP1 Cell Confluency and Proliferation Under Temporal and Pharmacological Conditions Using Deep Learning Techniques”** and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

____

Assoc. Prof. Dr. Arban Uka
Head of Department
Date: June, 26, 2024

Examining Committee Members:

Assoc. Prof. Dr. Arban Uka (Computer Engineering) __________________

Prof. Dr. Betim Çiço (Computer Engineering) __________________

Dr. Florenc Skuka (Computer Engineering) __________________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name Surname: Alban Xhepi

Signature:

ABSTRACT

Quantitative Analysis of THP1 Cell Confluency and Proliferation Under Temporal and Pharmacological Conditions Using Deep Learning Techniques

Alban Xhepi

M.Sc., Department of Computer Engineering

Supervisor: Assoc. Prof. Dr. Arban Uka

The proliferation and behavior of THP1 cells, a human monocytic cell line, are critical in understanding various biomedical and pharmaceutical applications. This thesis presents a comprehensive analysis of THP1 cell images categorized into different states: 'D2_PAR30' treated with varying concentrations of the drug (5µg, 20µg, 50µg, and 500µg). The primary objectives are to develop and optimize UNet models for accurate cell segmentation, quantify cell confluency, and analyze cell health based on confluency metrics across these categories.

Initially, the THP1 dataset, comprising unique and newly labeled cell images, was preprocessed. Original images (1080x1024) were cropped into smaller sizes (128x128, 256x256, and 512x512) and augmented to enhance dataset diversity. These preprocessed images were then used to train a UNet model for cell segmentation, with the 256x256 dataset yielding the best performance. Hyperparameters, loss functions, batch sizes, and epochs were carefully experimented with to optimize the segmentation accuracy.

To optimize the model for edge devices, pruning and quantization techniques were employed. Pruning reduced the model size from 355 MB to 100 MB, while quantization further decreased it to 35 MB, making the model significantly more efficient without compromising accuracy.

A pipeline was developed to automate the analysis process. Original cell images were divided into 256x256 segments, each segment's cell confluency and area were predicted, and the results were aggregated to assess the overall confluency and cell area of the original image. This method facilitated the evaluation of cell proliferation and confluency changes over time and under different drug treatments, enabling differentiation between healthy and unhealthy cells based on confluency.

The analysis revealed distinct patterns of cell confluency and proliferation associated with temporal changes and drug treatments. By testing 10 images from each category, significant insights were gained into the cellular response under different conditions. These findings contribute to the broader understanding of THP1 cell behavior and provide a foundation for future research in cellular biology and pharmacological studies.

Keywords: *THP1, U-net, segmentation, area, SVD, confluency, pruning, quantization, classification.*

ABSTRAKT

Analiza Sasiore e Konfluencës dhe Proliferimit të Qelizave THP1 nën Kushte Temporale dhe Farmakologjike duke Përdorur Teknikat e Deep Learning

Alban Xhepi

Master Shkencor, Departamenti i Inxhinierisë Kompjuterike

Udhëheqësi: Assoc. Prof. Dr. Arban Uka

Proliferimi dhe sjellja e qelizave THP1, një linjë qelizore monoksite humane, janë të rëndësishme për të kuptuar aplikimet e ndryshme biomedikale dhe farmaceutike. Kjo disertacion paraqet një analizë të përmbajtshme të imazheve të qelizave THP1 të kategorizuara në gjendje të ndryshme: 'D2_PAR30' trajtuar me koncentrata të ndryshme të ilaçit (5µg, 20µg, 50µg, dhe 500µg). Objektivat kryesore janë zhvillimi dhe optimizimi i modeleve UNet për segmentimin e saktë të qelizave, vlerësimi i konfluencës së qelizave, dhe analiza e shëndetit të qelizave bazuar në metrikat e konfluencës në këto kategori.

Fillimisht, dataseti i THP1, i përbërë nga imazhe unike dhe të etiketuara së fundmi të qelizave, u parapërpunua. Imazhet origjinale (1080x1024) u prerën në madhësi më të vogla (128x128, 256x256, dhe 512x512) dhe u shtuan për të përmirësuar diversitetin e datasetit. Këto imazhe të parapërpunuara u përdorën për të trajnuar një model UNet për segmentimin e qelizave, me datasetin 256x256 që dha performancën më të mirë. Hiperparametrat, funksionet e humbjes, madhësitë e grupit dhe epokat u eksperimentuan me kujdes për të optimizuar saktësinë e segmentimit.

Për të optimizuar modelin për pajisje të kufizuar (edge devices), u përdorën teknika të prerjes dhe kvantizimit. Prerja reduktoi madhësinë e modelit nga 355 MB në 100 MB, ndërsa kvantizimi e reduktoi më tej në 35 MB, duke bërë modelin shumë më efikas pa kompromentuar saktësinë.

U zhvillua një pipeline për të automatizuar procesin e analizës. Imazhet origjinale të qelizave u ndanë në segmente 256x256, për çdo segment u parashikua konfluencën dhe zona e qelizave, dhe rezultatet u mbledhën për të vlerësuar konfluencën dhe zonën e përgjithshme të imazhit origjinal. Ky metodë lehtësoi vlerësimin e proliferimit dhe ndryshimeve të konfluencës së qelizave në kohë dhe nën trajtime të ndryshme me ilaç, duke mundësuar dallimin mes qelizave të shëndetshme dhe atyre të pa shëndetshme në bazë të konfluencës.

Analiza zbuloi modelet e dallueshme të konfluencës dhe proliferimit të qelizave në lidhje me ndryshimet kohore dhe trajtimet me ilaç. Duke testuar 10 imazhe nga çdo kategori, u fituan njohuri të rëndësishme për reagimin qelizor në kushte të ndryshme. Këto zbulime kontribuojnë në kuptimin më të gjerë të sjelljes së qelizave THP1 dhe ofrojnë një bazë për hulumtimet e ardhshme në biologjinë qelizore dhe studimet farmakologjike.

Fjalët kyçe: *THP1, U-net, segmentim, siperfaqe, SVD, konfluenc*ë, *pruning, kuantizim, klasifikim.*

Dedicated to everyone whom I loved and couldn’t spend time with

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to Assoc. Prof. Dr. Arban Uka for his invaluable patience, experience sharing and feedback that he has provided to me during the time that we have worked together. I am also grateful to my classmates for their editing help and moral support.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

CHAPTER 1

INTRODUCTION

1. Thesis Objective

The primary objective of this thesis is to conduct a comprehensive analysis of THP1 cell images to evaluate cell confluency and proliferation across different temporal states and under varying pharmacological treatments. This involves several specific goals: developing and optimizing UNet models for accurate cell segmentation, quantifying cell confluency, and analyzing cell health based on confluency metrics across different categories ('D1_Cells', 'D2_Cells', and 'D2_PAR30' treated with varying drug concentrations of 5µg, 20µg, 50µg, and 500µg).

To achieve these objectives, the THP1 dataset, comprising unique and newly labeled cell images, was preprocessed by cropping the original images (1080x1024) into smaller sizes (128x128, 256x256, and 512x512) and augmenting them to enhance dataset diversity. These preprocessed images were used to train a UNet model for cell segmentation, with extensive experimentation on hyperparameters, loss functions, batch sizes, and epochs to optimize segmentation accuracy.

For deployment on edge devices, pruning and quantization techniques were employed to reduce the model size from 355 MB to 100 MB and further down to 35 MB, respectively, without compromising accuracy.

An automated pipeline was developed to process original cell images by dividing them into 256x256 segments, predicting cell masks, confluency, and area for each segment, and aggregating the results to determine the overall confluency and cell area of the original image.

The analysis of cell confluency and proliferation across different categories was conducted by comparing changes over time and under different drug treatments. This process enabled differentiation between healthy and unhealthy cells based on confluency metrics. Validation of the pipeline and analysis results was achieved by testing 10 images from each category, providing significant insights into the cellular response to temporal changes and drug treatments. These findings contribute to a broader understanding of THP1 cell behavior and provide a valuable foundation for future research in cellular biology and pharmacological studies.

2. Scope of works

This thesis encompasses a comprehensive study of THP1 cell images, focusing on the evaluation of cell confluency and proliferation across different temporal states and under varying pharmacological treatments. The scope of work includes several key components, beginning with the development and preprocessing of the dataset. THP1 cell images are collected and categorized into 'D1_Cells' (Day 1), 'D2_Cells' (Day 2), and 'D2_PAR30' treated with drug concentrations of 5µg, 20µg, 50µg, and 500µg. The original images (1080x1024) undergo preprocessing, which involves cropping them into smaller sizes (128x128, 256x256, and 512x512) and applying data augmentation techniques to enhance the diversity and robustness of the dataset.

The next component involves the development and optimization of a UNet model for cell segmentation. The preprocessed images are used to train the model, with extensive experimentation on hyperparameters, loss functions, batch sizes, and epochs to achieve optimal segmentation accuracy. To ensure the model's efficiency for deployment on edge devices, pruning and quantization techniques are applied, reducing the model size from 355 MB to 100 MB and further down to 35 MB without compromising its accuracy.

An automated analysis pipeline is then developed to streamline the evaluation process. This pipeline processes the original cell images by dividing them into 256x256 segments, predicting cell masks, confluency, and area for each segment, and aggregating the results to determine the overall confluency and cell area of the original image. This approach allows for a detailed analysis of cell confluency and proliferation across different categories, comparing changes over time and under different drug treatments.

The scope of work also includes validating the pipeline and analysis results by testing 10 images from each category. This validation provides significant insights into the cellular response to temporal changes and drug treatments, advancing the understanding of THP1 cell behavior and contributing valuable data for future research in cellular biology and pharmacological studies.

3. Organization of the thesis

This thesis is organized into several chapters, each detailing a specific aspect of the research and findings.

Chapter 1 provides an introduction to the study, outlining the significance of understanding the proliferation and behavior of THP1 cells in various biomedical and pharmaceutical applications. It presents the research problem, objectives, and the scope of work.

Chapter 2 reviews the relevant literature on cell segmentation, deep learning techniques for image analysis, and the application of UNet models in biomedical research. This chapter also discusses the optimization techniques for deploying models on edge devices.

Chapter 3 describes the methodology used in the study. It details the development and preprocessing of the THP1 dataset, including the categorization of images and the application of data augmentation techniques. The chapter also explains the training and optimization of the UNet model, including the experimentation with hyperparameters, loss functions, batch sizes, and epochs.

Chapter 4 focuses on the implementation of the automated analysis pipeline. It provides a step-by-step description of the pipeline development, from segmenting the original images to predicting cell masks, confluency, and area, and aggregating the results to assess overall confluency and cell area.

Chapter 5 presents the results of the analysis. This chapter discusses the patterns of cell confluency and proliferation observed across different temporal states and drug treatments. It includes a detailed comparison of changes in confluency and cell count, and differentiates between healthy and unhealthy cells based on confluency metrics.

Chapter 6 validates the pipeline and analysis results by testing images from each category. It highlights the significant insights gained into the cellular response to temporal changes and drug treatments, contributing to a broader understanding of THP1 cell behavior.

Chapter 7 concludes the thesis by summarizing the key findings and their implications for future research in cellular biology and pharmacological studies. It also discusses the limitations of the study and suggests potential areas for further investigation.

Finally, the appendices provide additional data, charts, and technical details that support the main text of the thesis, including detailed descriptions of the datasets, model parameters, and the algorithms used in the analysis.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The purpose of this literature review is to provide a comprehensive background for the study by examining existing research on cell segmentation methodologies, particularly focusing on deep learning techniques. Accurate cell segmentation is essential in biomedical research as it enables precise quantification of cell characteristics, which is crucial for understanding cellular processes and disease mechanisms.

This review is structured to cover the following topics:

- **Cell Segmentation in Biomedical Research**: This section discusses the importance of cell segmentation, traditional segmentation methods, and the challenges faced in accurately segmenting cells from microscopic images.
- **Deep Learning Techniques for Image Analysis:** Here, we explore the impact of deep learning on image analysis, highlighting its advantages over traditional methods and the specific relevance of convolutional neural networks (CNNs) in this field.
- **The UNet Model:** This section provides a detailed description of the UNet architecture, its key features, and its applications in biomedical image segmentation, emphasizing its effectiveness and versatility.
- **THP1 Cells and Their Significance:** This part introduces THP1 cells, their characteristics, and their importance in biomedical and pharmacological

research, supported by previous studies involving THP1 cell analysis and segmentation.

- **Data Augmentation:** Here we explain the role of data augmentation in enhancing model performance, covering common techniques used in image analysis and their specific applications in cell segmentation research.
- **Model Optimization Strategies:** This section explores model optimization methods such as pruning and quantization, discussing their benefits in terms of efficiency and deployment in biomedical research.
- **Comparison of Cell Segmentation Models:** A review of other cell segmentation models and their performance, comparing these with the UNet architecture to provide insights on model selection for cell segmentation tasks.
- **Challenges and Future Directions:** This final section highlights the current challenges in cell segmentation using deep learning, potential future advancements, and emerging trends in biomedical image analysis.

By systematically covering these topics, the literature review establishes a thorough context for the subsequent chapters, identifying gaps in current research and setting the stage for the contributions of this thesis

2.2 Cell Segmentation in Biomedical Research

2.2.1 Historical perspective on cell segmentation techniques.

Historically, cell segmentation techniques have come a long way to meet the growing demands of biomedical research. In the early days, manual segmentation was the norm, where researchers painstakingly drew cell boundaries by hand. While this method was often accurate, it was extremely time-consuming and prone to variability due to its subjective nature [1].

As digital imaging technology advanced, automated segmentation techniques began to emerge. One of the first automated methods was thresholding, which converts grayscale images into binary images by applying a global or local threshold. Although simple, thresholding struggled with variations in cell intensity and overlapping cells [2]. Edge detection methods, like the Sobel and Canny edge detectors, improved on thresholding by identifying cell boundaries based on intensity gradients. However, these methods were still challenged by noise and the complex shapes of cells [3].

Region-based methods, like the watershed algorithm and the active contour models, marked significant progress by focusing on the areas within cell boundaries. The watershed algorithm, for example, treats the grayscale image as a topographic surface and finds cell boundaries by simulating a flooding process. Despite their sophistication, these methods often required extensive preprocessing and parameter tuning to perform well [4].

The advent of machine learning algorithms brought further improvements. Traditional ML methods, like random forests and support vector, used handcrafted features extracted from the images. These models improved segmentation accuracy but were limited by the quality of the features and the need for extensive feature engineering [5].

The introduction of deep learning has revolutionized cell segmentation. Models like CNNs and FCNs can learn features directly from raw image data, greatly enhancing segmentation performance. The UNet model, a type of FCN, has become particularly popular due to its ability to perform precise segmentation with relatively small training datasets [6].

Cell segmentation techniques have evolved from manual methods to sophisticated deep learning models, each step addressing the limitations of its predecessors and contributing to more accurate and efficient segmentation in biomedical research.

2.2.2 Importance of accurate cell segmentation in biomedical research.

Accurate cell segmentation is crucial in biomedical research because it forms the foundation for many subsequent analyses and interpretations. Precise segmentation allows researchers to quantitatively analyze cell morphology, count cells, and measure various cellular features, which are essential for understanding cellular processes and disease mechanisms [1]. For example, in cancer research, accurately segmenting cells can help identify and quantify the morphological changes associated with malignancy, aiding in diagnosis and treatment planning [7].

Moreover, cell segmentation is vital in drug discovery and development. By accurately segmenting cells, researchers can assess the effects of pharmaceutical compounds on cell proliferation, apoptosis, and other cellular behaviors. This helps identify potential therapeutic targets and evaluate drug efficacy [8]. In tissue engineering and regenerative medicine, segmentation is used to analyze cell growth and tissue formation, which are critical for developing effective treatments and therapies [9].

Accurate cell segmentation also plays a significant role in high-throughput screening technologies, where large-scale automated image analysis is required. High-throughput screening relies on precise segmentation to process and analyze vast amounts of image data efficiently, facilitating the rapid identification of promising compounds or genetic modifications [10].

Advances in personalized medicine depend heavily on accurate cell segmentation. By analyzing patient-specific cell images, researchers can develop tailored treatments based on the unique cellular characteristics of an individual's disease, improving treatment outcomes and reducing adverse effects [6].

Accurate cell segmentation is essential for a wide range of biomedical applications, from basic research to clinical practice. It enables precise quantification and analysis of cellular features, advancing our understanding of health and disease, developing new therapies, and improving patient care.

2.2.3 Challenges and common issues faced in cell segmentation.

Despite significant advancements in cell segmentation techniques, numerous challenges and common issues persist in the field. One major challenge is the inherent variability in cell morphology. Cells can vary greatly in size, shape, and texture, even within the same tissue type or sample, making it difficult to develop a one-size-fits-all segmentation algorithm [1]. This variability necessitates the use of robust algorithms capable of handling a wide range of cell appearances.

Another common issue is the presence of noise and artifacts in microscopic images. Factors such as uneven illumination, staining variability, and imaging artifacts can obscure cell boundaries and complicate the segmentation process [11]. Preprocessing steps, such as noise reduction and contrast enhancement, are often required but may not fully eliminate these issues, leading to segmentation inaccuracies.

Overlapping cells pose another significant challenge. In many biological samples, cells are densely packed and often overlap, making it difficult to distinguish individual cell boundaries. Traditional segmentation methods, such as thresholding and edge detection, struggle with this issue, while more advanced methods like watershed algorithms and active contour models require careful parameter tuning to perform effectively [4]. An example of overlapping cells is shown in fig.1

Segmentation accuracy is also affected by the presence of different cell types within the same image. In heterogeneous samples, different cell types may have distinct morphological features, making it challenging to develop a universal segmentation approach. This issue often necessitates the use of machine learning and deep learning techniques that can learn to differentiate between various cell types based on training data [12].

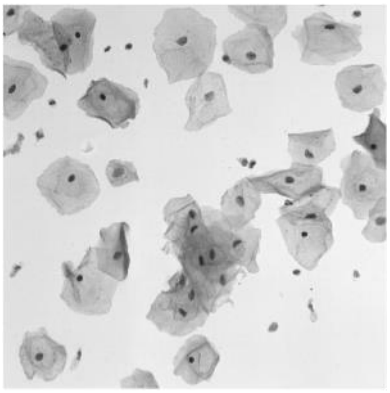


Figure 1 Overlapping cells example

Additionally, manual annotation of cell images for training and validation of segmentation algorithms is time-consuming and prone to human error. Creating large annotated datasets requires significant effort and expertise, and inconsistencies in manual annotations can affect the performance of machine learning models [13].

The challenges in cell segmentation stem from the inherent variability in cell morphology, noise and artifacts in images, overlapping cells, heterogeneous cell types, and the labor-intensive nature of manual annotation. Addressing these challenges requires the development of sophisticated algorithms and the use of advanced ML and deep learning techniques to achieve accurate cell segmentation in biomedical research.

2.3 Deep Learning Techniques for Image Analysis

2.3.1 Overview of deep learning and its application in image analysis

Deep learning has significantly changed the landscape of image analysis, offering notable improvements over traditional methods. At its heart, deep learning involves training artificial neural networks on big datasets to recognize features in the data. These networks, especially convolutional neural networks (CNNs), have shown exceptional success in various image analysis tasks and this is done because of their ability to understand and represent complex data patterns [14].

One of the major benefits of deep learning in image analysis is its capability for automatic feature extraction. Traditional image processing techniques often rely on handcrafted features, which can be time-consuming to design and may not capture all the relevant information in an image. Unlike traditional methods, deep learning models extract features directly from the raw pixel data, enabling them to identify complex patterns and relationships that might be missed by human-designed features [15].

Convolutional neural networks are particularly effective for image analysis. CNNs consist of multiple layers, including convolutional layers that apply filters to the input image, pooling layers that reduce dimensionality, and fully connected layers that perform classification. This layered structure allows CNNs to capture spatial hierarchies in images, making them highly effective for tasks such as object detection, image classification, and segmentation [16].

In biomedical image analysis, deep learning has been applied to a wide range of problems, from cell segmentation and classification to disease diagnosis and prognosis. For instance, CNNs have been used to accurately segment cells in microscopy images, identify cancerous tissues in histopathology slides, and detect abnormalities in medical imaging scans such as MRI and CT [17]. The ability of deep learning models to handle large, complex datasets and learn from vast amounts of data has made them indispensable tools in modern biomedical research.

Furthermore, advancements in deep learning frameworks and the availability of powerful computational resources have facilitated the development and deployment of deep learning models. Libraries such as TensorFlow and PyTorch provide robust tools for building and training neural networks, while advances in GPU technology have significantly reduced the time required to train large models [18].

Deep learning has transformed image analysis by providing powerful, automated tools for feature extraction and pattern recognition. Its application in biomedical image analysis has led to significant advancements in cell segmentation, disease detection, and other critical tasks, making it an essential technology in the field.

2.3.2 Comparison of traditional image processing methods with deep learning approaches

Traditional image processing methods and deep learning approaches differ significantly in their methodologies, capabilities, and outcomes. Traditional image processing relies on manually designed algorithms and handcrafted features to analyze images. Techniques such as thresholding, edge detection, and morphological operations have been commonly used for tasks like image segmentation and object detection [10]. These methods often require significant expertise to design and fine-tune for specific applications, and their performance can be limited by the quality of the manually extracted features and the variability in image data [20].

Thresholding is a straightforward method where pixel values are divided into classes, usually based on intensity levels. While simple, this method struggles with variations in lighting and noise, often resulting in poor performance in complex images [2]. Edge detection techniques, like the Canny and Sobel filters, focus on identifying the boundaries of objects within an image. Although effective for certain applications, they can be sensitive to noise and may fail to capture subtle or complex boundaries [3]. Morphological operations, which include dilation, erosion, opening, and closing, are used to refine segmented regions but are heavily dependent on the initial segmentation quality and the chosen structuring elements [21].

In contrast, Deep learning techniques, especially CNNs, have transformed image analysis by learning to automatically extract relevant features from raw image data. This process eliminates the need for manual feature engineering, allowing the models to adapt to the data and learn intricate patterns that might be overlooked by traditional methods [15]. CNNs consist of multiple layers that apply filters to the input images, capturing spatial hierarchies and complex features at various levels of abstraction. This hierarchical feature learning enables CNNs to achieve superior performance in main tasks like image classification, object detection, and image segmentation [16].

The key advantages of deep learning over traditional methods is its ability to handle large and diverse datasets. Deep learning models improve with more data, learning more robust and generalizable features. This is particularly important in biomedical image analysis, where variability in tissue types, imaging modalities, and pathological conditions can be high [17]. Additionally, deep learning models can be fine-tuned on specific tasks through transfer learning, leveraging pre-trained models to adapt to new, smaller datasets with reduced training time and improved performance [22].

Moreover, deep learning models can incorporate contextual information and learn complex spatial relationships within images, which is challenging for traditional methods. For instance, in cell segmentation, CNNs can learn to distinguish overlapping cells and varying cell morphologies more effectively than traditional algorithms [12]. However, deep learning approaches require substantial computational resources and large annotated datasets for training, which can be a limitation in some scenarios [23].

While traditional image processing methods are useful for simple and well-defined tasks, deep learning approaches offer greater flexibility, accuracy, and scalability for complex image analysis tasks. The ability of deep learning models to automatically learn from data and adapt to various challenges makes them a powerful tool in modern image analysis.

2.3.3 Advantages of using deep learning for cell segmentation

Deep learning offers several significant advantages for cell segmentation, making it a preferred approach over traditional methods. One of the primary benefits is its capability to automatically extract and learn different features from raw image data. Unlike traditional methods that rely on handcrafted features, deep learning models, particularly convolutional neural networks (CNNs), learn to identify important patterns and structures in the data through training [14]. This automatic feature learning leads to more accurate and robust segmentation results.

Another major advantage of deep learning is its capacity to handle large and complex datasets. In biomedical research, cell images can vary widely in terms of scale, texture, and morphology. Deep learning models are well-suited to manage this variability as they can generalize from large amounts of training data, learning to recognize cells in diverse conditions and environments [15]. This capability is particularly valuable for segmenting cells in heterogeneous tissues or under different imaging conditions.

Deep learning models, such as the UNet, have shown exceptional performance in segmenting overlapping and densely packed cells. Traditional methods often struggle with these challenges, as they typically rely on predefined rules that may not account for the complexity of biological structures. In contrast, deep learning models can learn to distinguish individual cells even in crowded and overlapping scenarios by leveraging their ability to understand spatial hierarchies and contextual information [6].

The flexibility and adaptability of deep learning are also notable advantages. They can be customized for specific tasks or datasets by employing transfer learning. This process involves adjusting a pre-trained model to new provided data, requiring only minimal additional training. This approach not only saves time and computational resources but also enhances the model's performance on specific segmentation tasks [22]. For example, a model pre-trained on general cell segmentation can be fine-tuned to accurately segment a particular type of cell or tissue.

Additionally, deep learning frameworks provide powerful tools for integrating multiple sources of data. In cell segmentation, integrating multimodal data, such as combining fluorescence and phase-contrast images, can significantly improve segmentation accuracy. Deep learning models can effectively learn from these different data types, combining them to produce more accurate and comprehensive segmentation results [24].

Lastly, deep learning models have been shown to be highly effective in minimizing human intervention and reducing the time required for analysis. Once trained, these models can process and segment large volumes of images rapidly and consistently, which is crucial for high-throughput screening and large-scale biomedical studies [17]. This efficiency not only accelerates the research process but also reduces the potential for human error and variability in manual segmentation.

The advantages of using deep learning for cell segmentation include automatic feature extraction, the ability to handle large and complex datasets, superior performance in challenging scenarios such as overlapping cells, flexibility through transfer learning, integration of multimodal data, and significant time savings and consistency in image analysis. These benefits make deep learning an indispensable tool in modern cell segmentation and biomedical research.

2.4 The UNet Model

2.4.1 Detailed description of the UNet architecture.

The UNet model, developed by Ronneberger et al. in 2015, is a convolutional neural network (CNN) architecture tailored specifically for biomedical image segmentation. Its distinctive U-shaped architecture allows it to capture both detailed local features and broad contextual information from images, making it highly effective for tasks requiring precise localization [6].

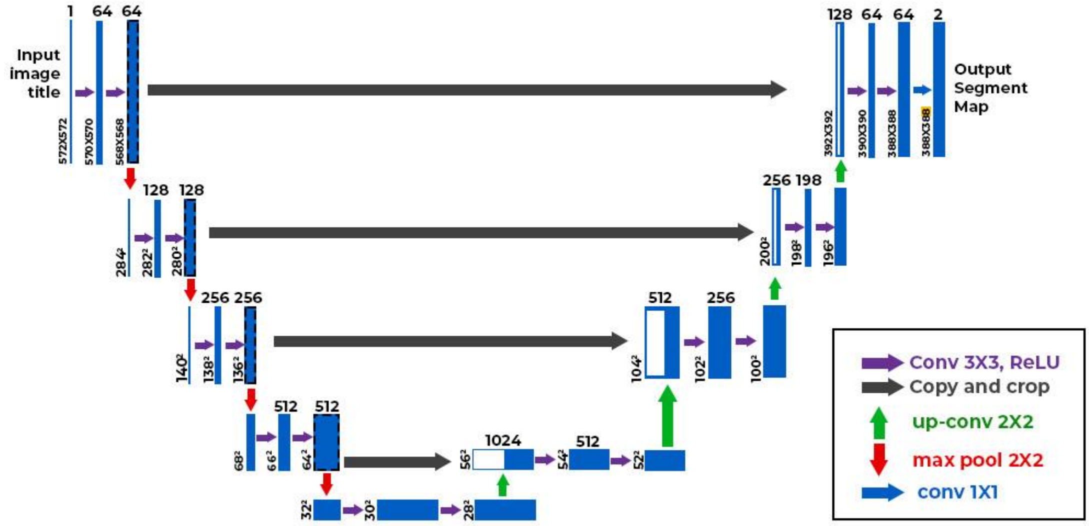


Figure 2 The Unet architecture example

The UNet architecture consists of two main parts: the contracting path (encoder) and the expanding path (decoder).

1. **Contracting Path (Encoder)**:
   - The contracting path is responsible for capturing the context of the input image. It comprises a series of convolutional layers, each followed by a rectified linear unit (ReLU) activation function and a max-pooling operation.
   - Each convolutional layer applies filters to the input image, generating feature maps that highlight different aspects of the image. The max-pooling layers reduce the spatial dimensions of these feature maps, effectively downsampling the image while retaining the most important features.
   - Typically, the contracting path includes several stages, with each stage doubling the number of feature channels. This allows the network to learn increasingly complex features at multiple scales [15].
2. **Bottleneck**:
   - At the bottom of the U-shape, between the contracting and expanding paths, is the bottleneck. This part of the network captures the most abstract features of the image. It consists of two convolutional layers

followed by ReLU activations, without any downsampling or upsampling operations.

3. **Expanding Path (Decoder)**:
    - The expanding path is designed to produce a high-resolution segmentation map. It includes a series of upsampling operations, each followed by a convolutional layer and a ReLU activation function.
    - Each upsampling step in the expanding path is accompanied by a concatenation with the corresponding feature map from the contracting path. These concatenations, known as skip connections, help the network retain spatial information that might have been lost during downsampling [25].
    - These skip connections allow the network to combine high-level features from the contracting path with high-resolution features from the expanding path, resulting in precise and accurate segmentations.
4. **Output Layer**:
    - The final layer of the UNet is a 1x1 convolution that reduces the number of feature channels to the desired number of output classes. This layer is followed by a softmax activation function (for multi-class segmentation) or a sigmoid activation function (for binary segmentation) to produce the final segmentation map [26].

The strength of the UNet architecture lies in its ability to learn both global context and local details simultaneously. This makes it particularly well-suited for tasks where precise localization is crucial, such as biomedical image segmentation. Since its introduction, the UNet model has been widely adopted and extended, demonstrating robust performance across a variety of biomedical imaging tasks [27].

2.4.2 Key features and components of the UNet model.

The UNet model is renowned for its robust performance in biomedical image segmentation, thanks to its unique design and key features. Here are the main components and features that contribute to its effectiveness:

1. **Symmetric U-shaped Architecture**:
    - The hallmark of the UNet model is its U-shaped architecture, featuring a symmetric design with a contracting path (encoder) and an expanding path (decoder). This configuration enables the model to capture both high-level contextual information and detailed features, which are crucial for accurate image segmentation [6].
2. **Contracting Path (Encoder)**:
    - The contracting path consists of multiple convolutional layers, each followed by a rectified linear unit (ReLU) activation function and a max-pooling operation. This segment of the network is tasked with downsampling the input image, capturing crucial features while decreasing its spatial dimensions. Each layer in the contracting path doubles the number of feature channels, enabling the network to learn progressively complex features at various scales [15].
3. **Expanding Path (Decoder)**:
    - The expanding path mirrors the contracting path and is designed to reconstruct the image resolution. It includes upsampling operations that the expanding path increases the spatial dimensions of the feature maps through upsampling, followed by convolutional layers and ReLU activations. The decoder merges high-level features from the encoder with the fine details recovered during upsampling, resulting in a precise segmentation map [25].
4. **Skip Connections**:
    - One of the key features of the UNet model is the incorporation of skip connections, which connect corresponding layers in the encoder and decoder paths. These connections concatenate feature maps from the contracting path to the expanding path, helping the network preserve

spatial information that might be lost during downsampling. This ensures that the final segmentation map is both precise and detailed [26].

5. **1x1 Convolutional Layers**:
   - The final layer of the UNet model is a 1x1 convolutional layer that reduces the number of feature channels to the desired number of output classes. This layer is followed by a softmax activation function (for multi-class segmentation) or a sigmoid activation function (for binary segmentation), which produces the final segmentation map [27].
6. **High Capacity for Learning Features**:
   - The UNet model's architecture, with its multiple layers and feature channels, provides a high capacity for learning complex features. This enables the network to effectively handle diverse and challenging segmentation tasks, such as segmenting overlapping or densely packed cells in biomedical images [28].
7. **Efficient Training with Data Augmentation**:
   - UNet models are particularly effective when combined with data augmentation techniques. Data augmentation helps to increase the diversity of the training dataset by applying transformations such as rotations, flips, and shifts. This improves the model's robustness and generalization capability, making it more effective in real-world applications [29].
8. **Flexibility and Adaptability**:
   - The UNet architecture is highly flexible and can be adapted to various image segmentation tasks. It has been extended to 3D versions for volumetric data and nested versions (like UNet++) for improved performance. This adaptability makes UNet a versatile tool for a wide range of biomedical imaging applications [27].

The key features and components of the UNet model, including its symmetric U-shaped architecture, skip connections, and high capacity for learning features, contribute to its outstanding performance in biomedical image segmentation.

2.4.3 Previous applications of the UNet model in biomedical image segmentation.

Since its introduction, the UNet model has been widely adopted and applied in numerous biomedical image segmentation tasks, demonstrating its versatility and robustness. Here are some notable applications:

The original application of the UNet model was for segmenting neuronal structures in electron microscopy stacks. Its success in this domain led to widespread adoption in other types of cell and nucleus segmentation. For instance, the UNet model has been used to segment cell nuclei in histopathological images, enabling precise quantification and analysis of cellular structures in cancer diagnosis and research [6]. UNet has also been applied to segment lung nodules in CT scans, aiding in the early detection of lung cancer. By accurately delineating the boundaries of lung nodules, the model assists radiologists in identifying potential malignancies, improving diagnostic accuracy and patient outcomes [30].

Additionally, the model has been employed in brain tumor segmentation from MRI scans. Accurate segmentation of brain tumors is crucial for treatment planning and monitoring disease progression. The UNet model's capability to capture fine details and contextual information makes it exceptionally effective in identifying and segmenting various tumor types and regions [31]. In ophthalmology, the UNet model has been used to segment retinal images, including tasks such as optic disc and cup segmentation, and the detection of retinal lesions. These applications are essential for diagnosing and managing conditions like glaucoma and diabetic retinopathy, where early detection and monitoring are critical [32].

UNet has also been applied to segment various organs in medical imaging, such as the liver in CT and MRI scans. This application is important for surgical planning, radiation therapy, and volumetric analysis of organs. The model's precision in segmenting organ boundaries helps clinicians in making informed decisions [33]. Moreover, the UNet model has been extensively used in the analysis of histopathology images, including tasks such as tissue classification and the segmentation of cancerous regions. Its ability to handle the high variability and complexity of histopathological data makes it a valuable tool for pathologists, aiding in accurate diagnosis and research [34].

In cardiology, UNet has been employed to segment cardiac structures from MRI and echocardiography images. Accurate segmentation of the heart's chambers and vessels is crucial for assessing cardiac function, diagnosing heart diseases, and planning treatments [35]. The model has also been adapted for multimodal image segmentation, where it combines information from different imaging modalities to improve segmentation accuracy. For example, combining MRI and PET scans can enhance the detection and segmentation of brain tumors by leveraging the complementary information provided by each modality [36].

The UNet model has proven to be a highly effective and adaptable tool for a wide range of biomedical image segmentation tasks. Its applications span across various medical fields, contributing to improved diagnostic accuracy, treatment planning, and research outcomes.

2.5 THP1 Cells and Their Significance

2.5.1 Introduction to THP1 cells and their characteristics

THP1 cells are a human monocytic cell line originating from a patient with acute monocytic leukemia.Established in 1980, these cells have become a staple in immunological and pharmaceutical research due to their versatility and human origin [37]. They offer a range of characteristics that make them incredibly valuable for studying various aspects of the immune system, drug responses, and cellular processes.

Characteristics of THP1 Cells:

1. **Monocytic Origin**:

THP1 cells are derived from monocytes, a type of white blood cell crucial for the immune response. Monocytes can differentiate into macrophages and dendritic cells, making THP1 cells an excellent model for studying these processes [38].

2. **Differentiation Capability**:

A key feature of THP1 cells is their ability to differentiate into macrophage-like cells when treated with phorbol 12-myristate 13-acetate (PMA) or other agents. This differentiation process changes the cells' morphology, adhesion properties, and the expression of specific markers, allowing researchers to study the changes associated with macrophage differentiation and activation [39].

3. **Human Origin**:

Being of human origin, THP1 cells provide a more relevant biological context compared to animal models. This is particularly important for drug testing and immunological studies, as it allows for more accurate predictions of human responses [40].

4. **Ease of Culturing**:

THP1 cells are relatively easy to culture and maintain in the lab. They grow in suspension, which makes them suitable for large-scale production and high-throughput screening. Their robust growth and ease of handling make them a popular choice in many laboratories [41].

5. **Expression of Key Immune Receptors**:

THP1 cells express a diverse array of immune receptors, including Toll-like receptors (TLRs), which are crucial for recognizing pathogens and triggering immune responses. This makes them an excellent model for studying innate immunity, inflammation, and pathogen-host interactions [42].

6. **Versatility in Research Applications**:

THP1 cells are used in various research areas, including immunology, cancer biology, toxicology, and drug development. Their ability to model different aspects of the human immune response makes them particularly useful for studying disease mechanisms and evaluating new therapeutic agents [43].

THP1 cells are a versatile and invaluable tool in biomedical research due to their human origin, ability to differentiate into macrophages, ease of culturing, and expression of key immune receptors. Their widespread use in various research applications highlights their significance in advancing our understanding of immune responses, disease mechanisms, and drug development.

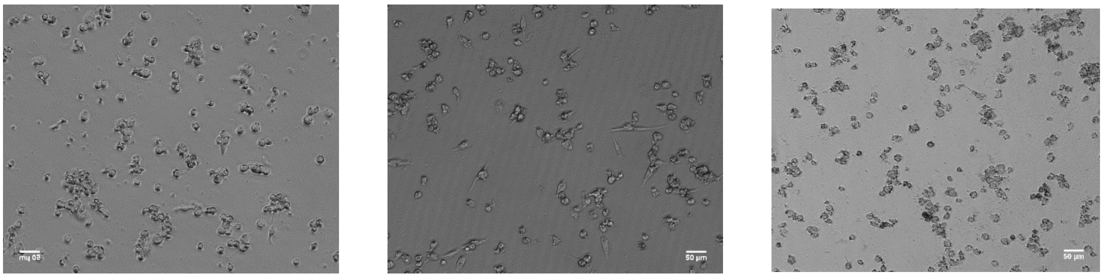


Figure 3 Exampels of THP1 cell images

4.5.2 Previous studies involving THP1 cell analysis and segmentation

THP1 cells have been extensively used in various research areas, showcasing their utility in biomedical studies. Here are some notable examples of how they have been applied:

1. **Inflammatory Response Studies**:

Researchers have used THP1 cells to study mechanisms of inflammation. For example, Chanput et al. (2014) investigated how different compounds affect immune responses by differentiating THP1 cells into macrophages and analyzing changes in cytokine production and other immune markers. This work provides valuable insights into inflammatory processes and potential therapeutic targets [38].

2. **Pathogen Interaction Analysis**:

THP1 cells are also used to study how pathogens interact with human cells. Arrouchi et al. (2014) utilized THP1 cells to examine the impact of human cytomegalovirus (HCMV) on immune cells. By segmenting the cells, they were able to observe viral entry, replication, and immune response, contributing to the development of treatments against HCMV [44].

3. **Cancer Research and Drug Testing**:

In cancer research, THP1 cells help evaluate the efficacy of new anti-cancer drugs. Savary et al. (2018) used THP1 cells to test novel compounds targeting leukemia. By segmenting and analyzing the treated cells, they assessed the drugs' effectiveness in inducing cell death and inhibiting proliferation, crucial steps in developing new cancer therapies [45].

4. **Macrophage Differentiation and Function Studies**:

THP1 cells are widely used to study macrophage differentiation and function. Daigneault et al. (2010) identified specific markers of macrophage differentiation in PMA-stimulated THP1 cells. By segmenting the differentiated cells, they analyzed marker expression, enhancing our understanding of macrophage biology and their role in immune responses [42].

5. **Nanoparticle Toxicity Assessments**:

THP1 cells are employed to assess the toxicity of nanoparticles. Franchi et al. (2017) investigated how different nanoparticles affect THP1 cells. By segmenting the exposed cells, they measured cytotoxicity and reactive oxygen species (ROS) production, which are critical for evaluating the safety of nanoparticles for medical use [46].

6. **Automated Image Analysis and High-Throughput Screening**:

Advances in automated image analysis have made high-throughput screening using THP1 cells more efficient. Beumer et al. (2014) developed an automated pipeline to segment and analyze THP1 cells in drug screening assays, enabling rapid and accurate evaluation of large numbers of compounds, thereby accelerating the drug discovery process [47].

7. **Gene Expression and Epigenetic Studies**:

THP1 cells are also used in gene expression and epigenetic research. Nishizawa et al. (2016) studied epigenetic changes during macrophage differentiation using THP1 cells. By segmenting the cells, they could isolate and analyze specific populations, providing insights into the regulatory mechanisms of macrophage differentiation [48].

Studies involving THP1 cell analysis and segmentation have greatly advanced our understanding of immune responses, pathogen interactions, cancer biology, nanoparticle toxicity, and gene expression. These applications highlight the versatility and importance of THP1 cells in biomedical research.

2.6 Model Optimization Strategies

2.6.1 Overview of model optimization techniques (pruning and quantization).

Model optimization techniques, such as pruning and quantization, are essential for enhancing the efficiency and performance of deep learning models, particularly for deployment on resource-constrained devices. These techniques reduce the computational and memory demands of neural networks without significantly compromising their accuracy.

Pruning:

Pruning is a technique that involves removing unnecessary or less important weights from a neural network. This process reduces the number of parameters, thereby decreasing the model's size and computational complexity. Pruning can be applied at various levels, including individual weights, neurons, or even entire layers [49].

There are several methods of pruning. Magnitude-based pruning involves removing weights with the smallest absolute values, on the assumption that these weights have the least impact on the model's performance. This approach is straightforward and widely used [50]. Structured pruning, on the other hand, focuses on removing entire filters or channels in convolutional networks rather than individual weights. This method is advantageous for reducing latency and improving hardware efficiency, as it results in more regular and dense computations [51]. Additionally, iterative pruning and fine-tuning involve performing pruning in stages and then fine-tuning the model to recover any accuracy loss. This approach helps maintain the model's performance while progressively reducing its size [52].

The primary benefits of pruning include a significant reduction in the memory footprint and computational requirements of neural networks. This facilitates the deployment of models on edge devices, such as smartphones and embedded systems, which have limited resources. Furthermore, pruned models can achieve faster inference times, making them more practical for real-time applications [53].

Quantization:

Quantization is a technique aimed at reducing the precision of the numbers used to represent a model's parameters and activations. Instead of utilizing 32-bit floating-point numbers, quantized models employ lower precision formats, such as 8-bit integers. This adjustment leads to decreased memory usage and faster computations [54].

There are several approaches to quantization. Post-training quantization involves converting a pre-trained model to a lower precision format after training has been completed. This method is straightforward but may cause some degradation in the model's accuracy [55]. On the other hand, quantization-aware training integrates quantization into the training process, allowing the model to adapt to the lower precision representation and generally resulting in better accuracy retention post-quantization [56]. Furthermore, dynamic and static quantization are two types of quantization strategies. Dynamic quantization applies to weights during inference, offering flexibility, while static quantization involves quantizing both weights and activations during training for optimized performance [57].

The primary benefits of quantization include a significant reduction in the size of the neural network, which enhances storage and processing efficiency. Additionally, it improves inference speed due to decreased computational complexity. Quantized models are especially advantageous for deploying deep learning applications on mobile and embedded devices, where computational and storage resources are limited [58].

Together with pruning, quantization forms a crucial part of model optimization. These techniques collectively enhance the efficiency and deployability of deep learning models, making it possible to run complex neural networks on devices with limited resources. This enables broader application and integration of AI technologies across various fields.

2.6.2 Examples of optimization strategies applied in biomedical research

Model optimization techniques, such as pruning and quantization, have demonstrated significant value in biomedical research by improving the efficiency and performance of deep learning models. Here are some notable examples:

In the field of medical imaging, Molchanov et al. (2017) applied pruning to convolutional neural networks (CNNs) used for analyzing medical images. By selectively removing less important filters, they managed to significantly reduce the model's size and computational demands without compromising accuracy. This approach was especially useful for deploying models on resource-constrained devices in clinical settings [53].

For ultrasound image analysis, Jacob et al. (2018) showcased the effectiveness of quantization. They applied post-training quantization to a CNN model, converting it from 32-bit floating-point to 8-bit integer precision. This optimization not only reduced the model size but also improved inference speed, facilitating real-time analysis of ultrasound images on portable devices [54].

Wu et al. (2016) applied both quantization and pruning techniques to a deep learning model for lung nodule detection in CT scans. By quantizing model weights and pruning redundant connections, they achieved a significant reduction in model size and computational load. This optimization enabled faster and more efficient processing of CT images, aiding in the early detection of lung cancer [57].

In cardiac image segmentation, Courbariaux et al. (2016) utilized binarized neural networks (BNNs). BNNs constrain weights and activations to +1 or -1, which drastically reduces memory and computational requirements. This method was applied to segment cardiac structures in MRI scans, resulting in efficient and accurate segmentation suitable for real-time applications [56].

Gupta et al. (2015) used limited numerical precision techniques for histopathological image classification. By employing reduced precision arithmetic, they optimized the model to operate efficiently on standard CPUs, making it accessible for various healthcare settings. This approach enabled quick and accurate classification of tissue samples, supporting pathologists in diagnostic processes [55].

Han et al. (2015) implemented pruning techniques in a deep learning model for brain tumor segmentation from MRI images. By pruning less significant weights, they reduced the model size and improved its inference speed, which made it feasible to deploy the model in clinical environments with limited computational resources. This optimization enhanced the model’s usability for real-time tumor detection and monitoring [49].

For diabetic retinopathy detection, LeCun et al. (1990) applied optimal brain damage pruning techniques to CNNs. This method involved removing weights that had minimal impact on the model’s performance, thereby reducing the network's overall complexity. The optimized model was then used to analyze retinal images, enabling early and accurate detection of diabetic retinopathy [50].

The application of optimization strategies such as pruning and quantization has greatly improved the efficiency and practicality of deep learning models in biomedical research. These techniques facilitate the deployment of complex models on resource-constrained devices, enabling real-time analysis and enhancing diagnostic accuracy across various medical domains.

2.7 Deep Learning Approaches for Cell Classification

Deep learning has significantly advanced cell classification by automating feature extraction and classification processes. Convolutional Neural Networks (CNNs) have become central to this transformation, offering the capability to learn and identify complex patterns in image data. This advancement has addressed many limitations of traditional cell classification methods, such as manual feature extraction and reliance on heuristic approaches.

Various deep learning architectures have been explored for cell classification, each bringing unique strengths to the table. VGGNet, known for its straightforward architecture with small receptive fields, is effective in image classification tasks but can suffer from longer training times and higher computational costs due to its depth. InceptionNet, on the other hand, employs multiple convolutional filters of different sizes in parallel, enabling it to capture features at various scales. However, this complexity can lead to significant computational demands.

Among these, ResNet (Residual Network) has emerged as a particularly effective model for cell classification. Introduced by He et al. (2015), ResNet incorporates residual blocks with skip connections to overcome the challenges associated with training very deep networks. The ResNet50 variant, which includes 50 layers, strikes a balance between depth and computational efficiency, making it well-suited for biomedical image analysis tasks [60].

ResNet50's architecture benefits from its residual learning framework, which addresses issues such as vanishing gradients and facilitates the training of deep networks. This allows the model to learn intricate features from complex cell images effectively. The use of residual connections helps maintain high performance even in deeper networks, which is crucial for accurately distinguishing between various cell types and states [61][62].

In practice, ResNet50 has demonstrated exceptional performance in cell classification tasks. For instance, it has been employed in cancer detection and cellular morphology analysis, where its deep architecture enables the model to capture subtle differences between cell categories. Comparative studies have shown that ResNet50 often outperforms other models like VGGNet and InceptionNet in terms of accuracy and robustness [63][6].

Despite its advantages, ResNet50 is not without challenges. It requires large annotated datasets for training and can struggle with class imbalances. Future research is focused on overcoming these challenges through techniques such as transfer learning, fine-tuning, and the integration of domain-specific knowledge. These approaches aim to further enhance the model’s performance and applicability in cell classification tasks.

CHAPTER 3

METHODOLOGY

3.1 Dataset Preparation

3.1.1 Data Collection

The dataset used in this study consists of 116 images of THP1 cells, which are human monocytic cells derived from a patient with acute monocytic leukemia. These images were specifically created for researchs at Epoka University. The images were captured using high-resolution microscopy, ensuring detailed visualization of cell structures. The dataset includes images from different experimental conditions, including untreated cells (D1_Cells and D2_Cells) and cells treated with varying concentrations of PAR30 (5 µg, 20 µg, 50 µg, and 500 µg).

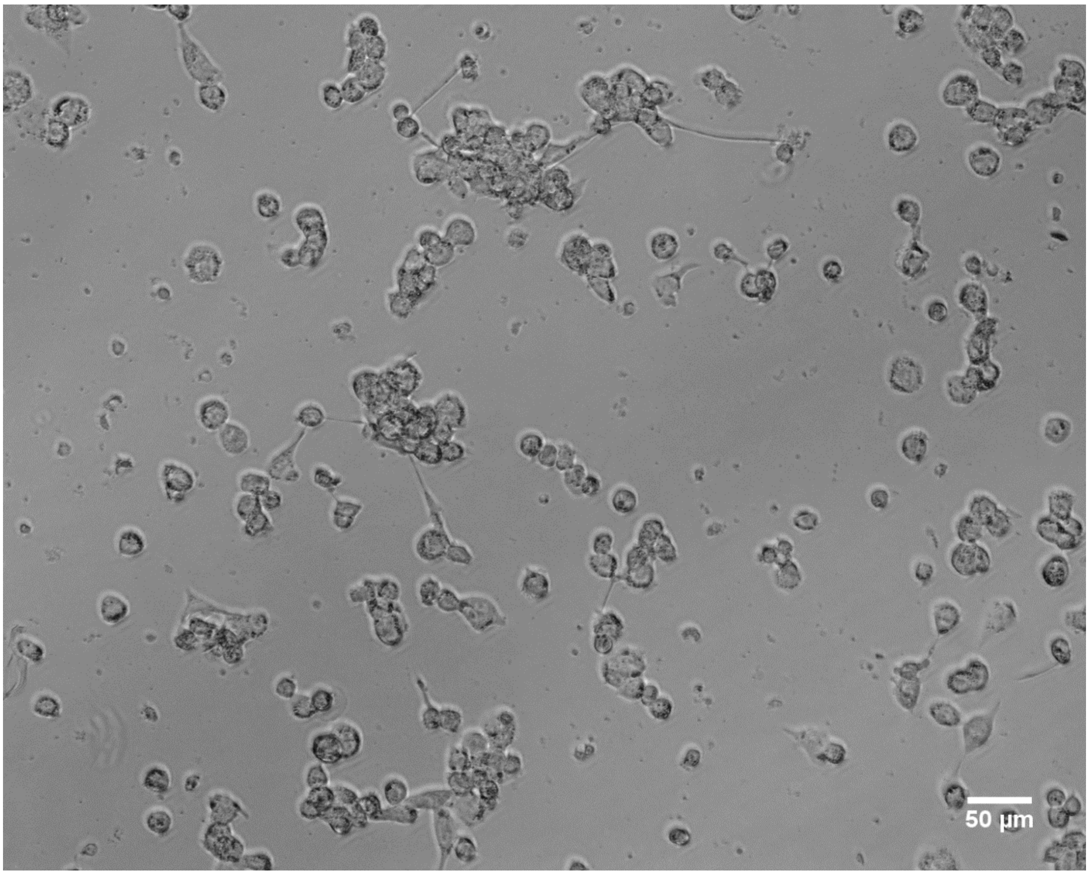
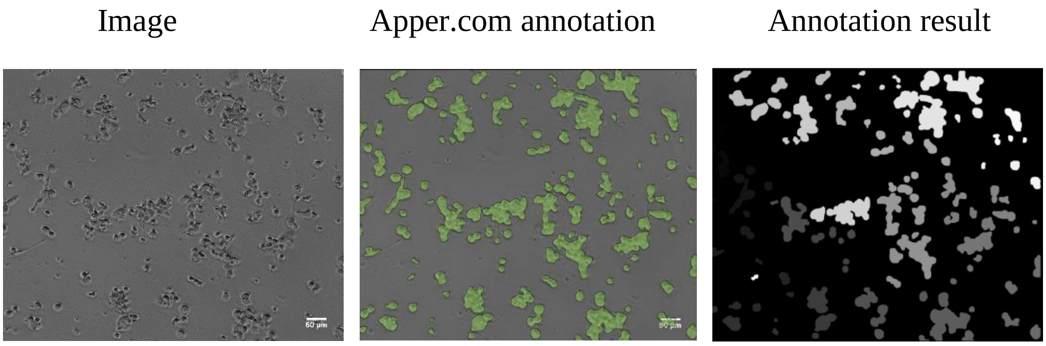


Figure 4 Exampe of Day 1 THP1 cells

3.1.2 Data Annotation

Out of the 116 images, 62 images were manually annotated to create ground truth labels for training, validation, and testing purposes. The annotation process was carried out using the Apeer.com platform, a specialized tool for precise and efficient image annotation. Expert annotators meticulously outlined the boundaries of individual cells within each image. These annotations were reviewed and validated to ensure accuracy and consistency. This split ensured that the model had sufficient data to learn from, while also allowing for effective evaluation of its performance.



Image

Apper.com annotation

Annotation result

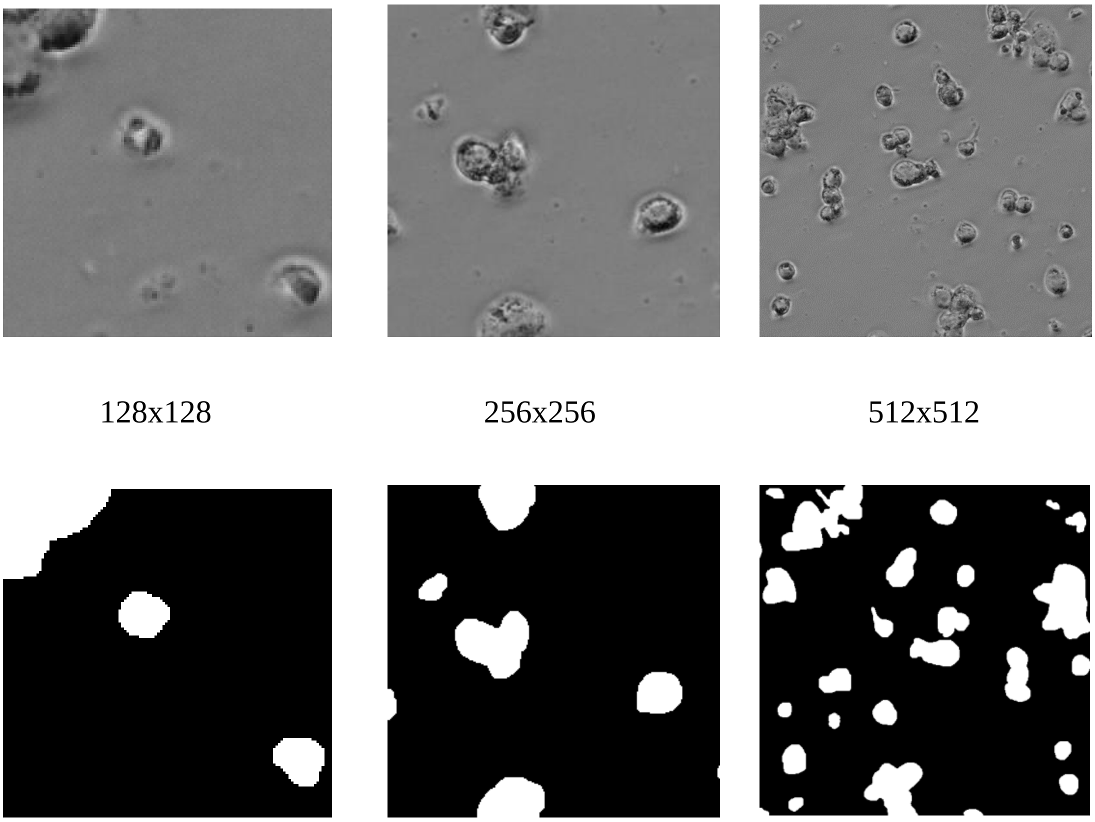
Figure 5 Making cell annotation using apeer.com

3.1.3 Data Preprocessing

Before feeding the images into the neural network, several preprocessing steps were undertaken to ensure the data was in the optimal format for model training:

Resizing:

Before feeding the images into the neural network, several preprocessing steps were undertaken to ensure the data was in the optimal format for model training. The original images, which were 1080x1024 pixels in size, were cropped into smaller patches of 128x128, 256x256, and 512x512 pixels. This resizing helped in standardizing the input dimensions and provided the model with varied perspectives of the cell structures.



128x128

256x256

512x512

Figure 6 Cropped images with their masks

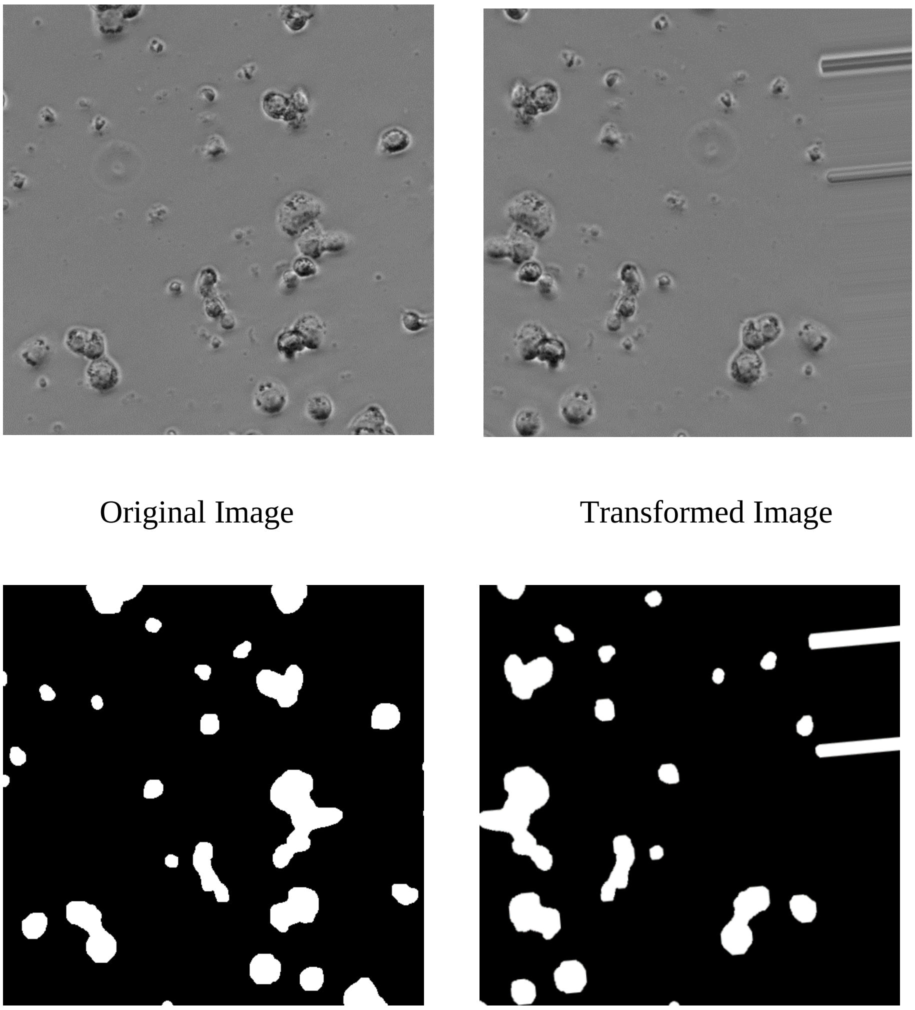
Normalization:

Image pixel values were normalized to a range of 0 to 1. This normalization step is crucial as it helps in stabilizing the training process and ensures that the model learns effectively without being affected by variations in pixel intensity.

Data Augmentation:

To enhance the model's accuracy, data augmentation techniques were applied. These included random rotations of up to 20 degrees to simulate different orientations, horizontal and vertical flipping to introduce variations in cell positioning, zooming in and out up to 20% to simulate different magnifications, and applying shear transformations up to 20% to introduce slight distortions and variations in the images.

We saw the need to use data augmentation only on the 512x512 dataset due to the limited number of images, which were only 248. This was insufficient for the effective training of the model. Our initial results showed very poor predictions before data augmentation, while the predictions improved significantly after applying data augmentation. Figure 7 illustrates an example of the transformed image using these data augmentation techniques.



Original Image

Transformed Image

Figure 7 Example of the output image after data augmentation

3.2 Image Segmentation Techniques

3.2.1 Choice of Model

The UNet model was chosen for this study due to its exceptional performance and suitability for biomedical image segmentation tasks. Several key factors influenced this decision. The UNet model has a symmetric U-shaped architecture with an encoder and a decoder, enabling it to capture high-level detailed features essential for precise cell segmentation. Skip connections between the contracting and expanding paths allow the model to integrate spatial information from various resolutions, improving its ability to generate accurate segmentation maps.

The UNet model has shown robustness in handling variations in cell shapes, sizes, and densities, which is particularly important for segmenting THP1 cells that can exhibit significant morphological diversity under different experimental conditions. The model's ability to learn and generalize from diverse training data makes it well-suited for this study. One significant advantage of the UNet model is its ability to achieve high performance even with limited training data. Given that only 62 annotated images were available for training in this study, the UNet model's efficiency in learning from small datasets was a critical factor. Its architecture allows it to leverage data augmentation effectively, improving the model's robustness and accuracy.

The UNet model has a proven track record in various biomedical image segmentation tasks, including cell segmentation, tumor detection, and organ segmentation. This extensive validation in the biomedical field provided confidence in its applicability and reliability for segmenting THP1 cells in this study. The UNet model is highly flexible and can be adapted to different image segmentation tasks. It supports various input and output sizes, making it suitable for segmenting images of different resolutions. Additionally, the model can be easily modified and fine-tuned to accommodate specific requirements of the study, such as incorporating additional layers or adjusting hyperparameters.

To enhance the segmentation performance further, four different UNet architectures were utilized:

1. **Basic U-Net Model**: The standard UNet architecture used as a baseline.
2. **U-Net Model with Dropout**: Incorporating dropout layers to prevent overfitting and improve generalization.
3. **U-Net Model with VGG16 Encoder**: Leveraging a pre-trained VGG16 model as the encoder to utilize transfer learning for better feature extraction.
4. **U-Net Model with Attention**: Integrating attention mechanisms to allow the model to focus on the most relevant parts of the image, enhancing segmentation accuracy.

These variations allowed for comprehensive experimentation to identify the most effective architecture for the specific requirements of the study.

The UNet model can be seamlessly integrated with modern deep learning frameworks like TensorFlow and PyTorch. These frameworks provide robust tools for model training, optimization, and deployment, facilitating the development and implementation of the segmentation pipeline. The availability of pre-built UNet implementations and community support further streamlined the process. Collectively, the UNet model's U-shaped architecture, robustness to variations, high performance with limited data, proven success in biomedical applications, flexibility, and seamless integration with modern deep learning frameworks make it an ideal choice for accurate and efficient segmentation of THP1 cells.

3.2.2 Model Architecture (UNET)

The UNet model is used to for this task. It is designed to efficiently segment images by capturing both high-level and low-level features. The UNet architecture is also shown in Fig.3.

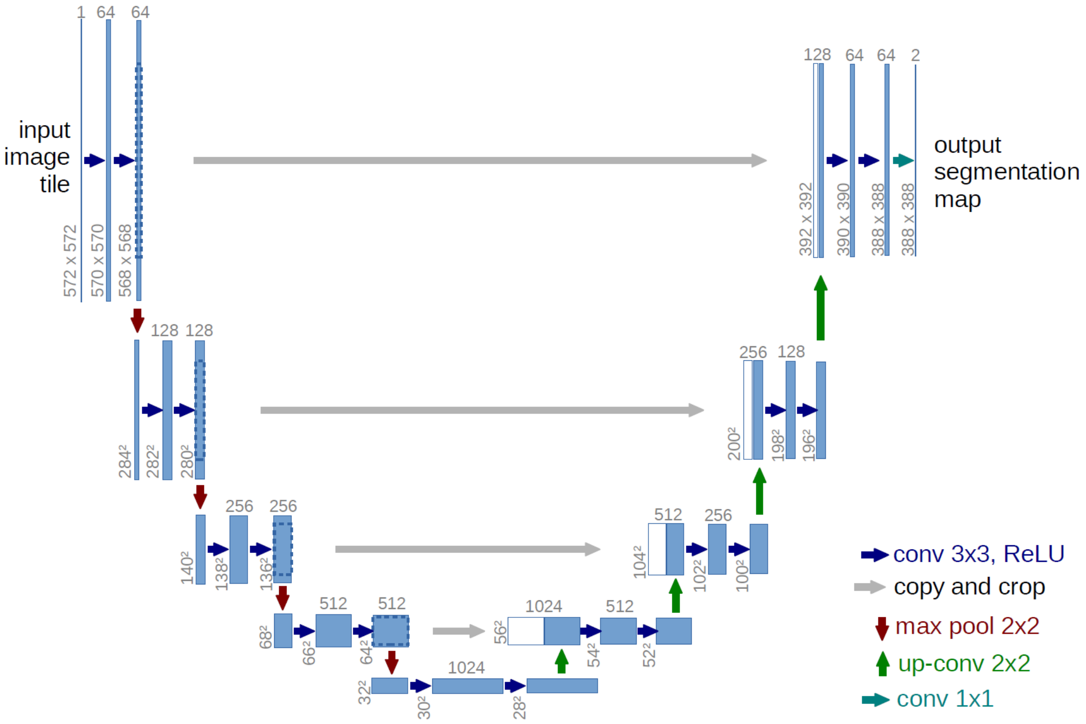


Figure 8 The UNet architecture used

1. **Basic U-Net Model:** The Basic U-Net model follows the standard UNet architecture, which consists of an encoder (contracting path), a bottleneck, a decoder (expanding path), and an output layer. The encoder captures contextual information through convolutional and max-pooling layers, while the decoder reconstructs the segmentation map through transposed convolutions and skip connections that integrate features from the encoder. This model serves as a baseline for comparison with other variants.
2. **U-Net Model with Dropout:** This variant incorporates dropout layers into the standard UNet architecture to prevent overfitting and improve generalization. Dropout layers are added after the convolutional layers in both the encoder and decoder paths. By randomly dropping units during training, the model becomes more robust and less sensitive to the noise in the training data, leading to better performance on unseen data.
3. **U-Net Model with VGG16 Encoder:** The U-Net model with a VGG16 encoder leverages a pre-trained VGG16 network as the encoder part of the

UNet architecture. VGG16 is a deep convolutional network known for its strong feature extraction capabilities. By utilizing transfer learning, the pre-trained VGG16 layers provide a rich set of features that improve the segmentation performance. The decoder part remains similar to the standard UNet, reconstructing the segmentation map using transposed convolutions and skip connections.

4. **U-Net Model with Attention:** This variant integrates attention mechanisms into the UNet architecture to enhance segmentation accuracy. Attention layers are added in the decoder path to allow the model to focus on the most relevant parts of the feature maps. These layers assign different weights to different parts of the input, enabling the model to give more importance to critical regions while downplaying irrelevant areas. This focus improves the model's ability to accurately segment cells, especially in complex and cluttered images.

Each of these architectures brings unique enhancements to the standard UNet model, aiming to improve segmentation performance and robustness for the specific task of THP1 cell segmentation.

3.2.3 Training Procedure

The training procedure for the UNet models involved several key steps to ensure effective learning and accurate segmentation of THP1 cells. Here is a detailed description of the training process:

1. **Data Preprocessing:**
   - **Data Augmentation:** To increase the diversity of the training data and prevent overfitting, various augmentation techniques were applied,

such as random rotations, flips, scaling, and translations. This helps the model generalize better to unseen data.

- **Normalization:** The input images were normalized to have zero mean and unit variance. This ensures that the model training is not biased by the varying intensity values of the images.
- **Splitting Data:** The dataset was divided into training, validation, and test sets. The training set was used to train the model, the validation set to tune hyperparameters and monitor overfitting, and the test set to evaluate the final model performance.

2. **Model Initialization:**
   - **Weights Initialization:** The weights of the convolutional layers were initialized using the He normal initialization method, which helps in maintaining the gradient flow during training.
   - **Pre-trained Weights:** For the U-Net model with VGG16 encoder, pre-trained weights on the ImageNet dataset were used for the encoder part to leverage transfer learning.
3. **Training Configuration:**
   - **Loss Function:** The binary cross-entropy loss function was used to measure the discrepancy between the predicted and actual segmentation maps. This loss function is suitable for binary segmentation tasks where the goal is to classify each pixel as either cell or background.
   - **Optimizer:** The Adam optimizer was employed for training the models due to its adaptive learning rate and efficient handling of sparse gradients. Learning rates of 0.001 and 0.0001 were tested to identify the optimal learning rate.
   - **Batch Size:** Batch sizes of 4 and 8 were chosen to balance between computational efficiency and the stability of gradient updates.
   - **Epochs:** The models were trained for 5, 10, 15, and 20 epochs, with early stopping implemented to halt training if the validation loss did not

improve for a set number of epochs. This prevents overfitting and saves computational resources.

4. **Model Compilation:**
    - The UNet model was compiled with the following settings:
        - **Loss Function:** Binary cross-entropy.
        - **Optimizer:** Adam optimizer with the specified learning rate.
        - **Metrics:** Dice coefficient, Intersection over Union (IoU), Recall, Precision, and Accuracy.
5. **Training Loop:**
    - **Forward Pass:** In each epoch, a forward pass was performed on the training batch to compute the predicted segmentation map.
    - **Loss Computation:** The binary cross-entropy loss was calculated based on the predicted and actual segmentation maps.
    - **Backward Pass and Optimization:** A backward pass was performed to compute the gradients of the loss with respect to the model parameters, followed by an update of the model parameters using the Adam optimizer.
    - **Validation:** After each epoch, the model was evaluated on the validation set to monitor its performance and adjust hyperparameters if necessary.
6. **Model Evaluation:**
    - **Performance Metrics:** The trained models were evaluated using various metrics, including the Dice coefficient, Intersection over Union (IoU), Recall, Precision, and Accuracy. These metrics provide a comprehensive assessment of the model's segmentation performance.
    - **Test Set Evaluation:** The final models were tested on the holdout test set to determine their generalization capability and overall effectiveness in segmenting THP1 cells.
7. **Post-training Optimization:**
    - **Model Pruning:** To reduce the model size and improve inference speed, model pruning techniques were applied. This involves removing

redundant parameters without significantly affecting the model's performance.

- **Quantization:** Further optimization was performed using quantization to reduce the model's memory footprint, enabling deployment on edge devices.

By following this rigorous training procedure, the UNet models were effectively trained to achieve accurate and efficient segmentation of THP1 cells, demonstrating robustness and high performance across various evaluation metrics.

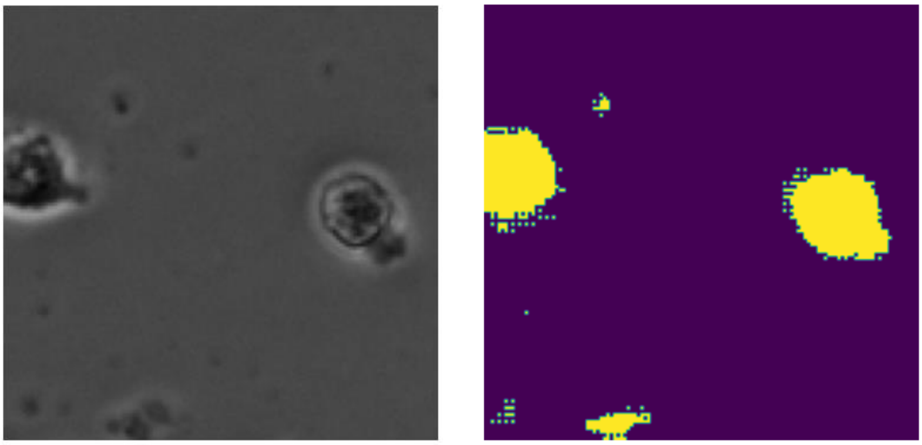


Figure 9 Some predictions from early testing

3.4 Model Evaluation

3.4.1 Performance Metrics

To thoroughly assess the performance of the UNet model in segmenting THP1 cells, several essential metrics were utilized. These metrics offered a comprehensive evaluation of the model's accuracy, precision, and overall effectiveness in segmentation tasks. The performance metrics included:

1. **Accuracy:**

- Accuracy measures how often the model makes the correct predictions. It is calculated by dividing the number of correct predictions (both true positives and true negatives) by the total number of predictions made.
- **Formula**:

$$\text{Accuracy} \ = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total PredictionsAccuracy}}$$

- **Significance**: Accuracy provides an overall idea of how well the model performs. However, it might not be enough on its own, especially when dealing with imbalanced classes where one class is much more frequent than the other.

2. **Precision**:
    - Precision, or Positive Predictive Value, measures how many of the model's positive predictions are actually correct. It shows the model's ability to avoid false positives.
    - **Formula**:

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

- **Significance:** High precision means the model has a low false positive rate, making it reliable for situations where false positives are especially costly.

3. **Recall**:
    - Recall, also called Sensitivity or True Positive Rate, measures how well the model identifies actual positive cases. It shows the model's ability to capture all relevant instances.
    - **Formula**:

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

- **Significance**: High recall means the model has a low false negative rate, making it reliable for applications where missing positive cases is critical..

4. **F1-Score**:
   - The F1-score is the harmonic mean of precision and recall, offering a single metric that balances both aspects. It is particularly useful when handling class imbalance.
   - **Formula**:

$$\mathrm{F1-Score} = \mathbf{2}\ \times \frac{\mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}$$

   - **Significance**: F1-score is the harmonic mean of precision and recall, giving a single metric that balances both aspects. It's particularly useful for situations with class imbalance.

5. **Intersection over Union (IoU)**:
   - IoU, or the Jaccard Index, measures how much the predicted segmentation mask overlaps with the ground truth mask. It is calculated by dividing the area of overlap (intersection) by the total area covered by both masks (union).
   - **Formula**:

$$\mathrm{IoU} = \frac{\mathrm{Intersection\ Area}}{\mathrm{Union\ Area}}$$

   - **Significance**: IoU is a robust metric for assessing the accuracy of segmentation models. A higher IoU signifies better segmentation performance, with perfect segmentation achieving an IoU of 1.

6. **Dice Coefficient**:
   - The Dice coefficient measures the similarity between the predicted and ground truth masks, similar to IoU but giving more weight to the overlapping area. It is calculated as twice the area of overlap divided by the total number of pixels in both masks.

- **Formula**:

$$\text{Dice Coefficient} = \frac{2 \times \text{Intersection Area}}{\text{Total Pixels in Both Masks}}$$

- **Significance**: The Dice coefficient is especially useful in medical image segmentation tasks, providing a reliable measure of model performance.

These performance metrics were used to evaluate the model at various stages of training and validation, ensuring a comprehensive understanding of its strengths and weaknesses. By analyzing these metrics, the effectiveness of the UNet model in segmenting THP1 cells was thoroughly assessed, providing confidence in its applicability and reliability for biomedical research.

3.2.2 Evaluation Procedure

The evaluation procedure was carefully designed to rigorously assess the performance of the UNet model on the test dataset. This ensured a thorough and unbiased evaluation, providing a reliable measure of the model's generalization capabilities. The following steps outline the evaluation process:

1. **Preparation of the Test Dataset:**
    - The test dataset consisted of 15% of the annotated images (approximately 9 images) set aside during the training phase. These images were not used for model training or validation, ensuring that the evaluation results accurately reflected the model's performance on unseen data.
2. **Loading and Preprocessing:**
    - The test images and their corresponding masks were loaded and preprocessed similarly to the training and validation data. This involved resizing the images to 128x128, 256x256, and 512x512 pixels,

normalizing the pixel values to a range of 0 to 1, and applying any necessary transformations to ensure compatibility with the model's input requirements.

3. **Model Inference:**
    - The trained UNet model was used to generate segmentation masks for the test images. Each test image was passed through the model to obtain the predicted segmentation masks.
4. **Performance Metrics Calculation:**
    - The model's performance was evaluated using several key metrics. For each test image, the predicted segmentation mask was compared to the ground truth mask, and the following metrics were calculated:
        - **Accuracy**
        - **Precision**
        - **Recall**
        - **F1-Score**
        - **Intersection over Union (IoU)**
        - **Dice Coefficient**
5. **Statistical Analysis:**
    - The performance metrics for all test images were aggregated and analyzed. Descriptive statistics, such as mean and standard deviation, were computed for each metric to provide a comprehensive summary of the model’s performance.
6. **Comparison with Validation Results:**
    - The performance metrics obtained from the test dataset were compared with those from the validation dataset. This comparison helped assess the model’s ability to generalize to new, unseen data and ensured that it was not overfitting to the training data.
7. **Visual Inspection:**

- In addition to quantitative metrics, a visual inspection of the predicted segmentation masks was conducted. Representative examples of segmented images were reviewed to qualitatively assess the accuracy and reliability of the model’s predictions.

By following these steps, the evaluation procedure ensured a comprehensive assessment of the UNet model's performance, demonstrating its effectiveness in accurately segmenting THP1 cells and providing insights into its generalization capabilities.

3.5 Model Optimization

3.5.1 Pruning

Pruning is a technique used to reduce the size and complexity of a neural network by removing less significant weights and neurons. In this study, pruning was applied to the UNet model to enhance its efficiency without significantly compromising its performance. The following sections describe the pruning techniques used, the criteria for pruning, and the fine-tuning process.

Magnitude-based pruning was employed, which involves removing weights based on their absolute values. This technique operates under the assumption that weights with smaller absolute values contribute less to the overall performance of the model. By eliminating these weights, the model's size is reduced while maintaining its accuracy.

Threshold-based pruning was implemented by setting a predefined threshold value to determine which weights to prune. Weights with absolute values below this threshold were pruned, ensuring that only the least significant weights were removed. Additionally, percentage-based pruning was used, where a specific percentage of the smallest magnitude weights was pruned. For instance, pruning 20% of the weights meant removing the 20% of weights with the smallest absolute values, allowing controlled reduction in model size.

After pruning, the model typically experiences a drop in performance. To mitigate this, the pruned model was retrained (fine-tuned) on the training dataset to recover any lost performance. The performance of the pruned and fine-tuned model was continuously monitored using the validation set. Metrics such as Dice coefficient, IoU, Recall, Precision, and Accuracy were tracked to ensure the model's performance was maintained or improved. The pruning and fine-tuning process was performed iteratively. After each round of pruning and fine-tuning, the model's performance was evaluated. If the performance was satisfactory, further pruning was carried out. This iterative process continued until the model reached an optimal balance between size reduction and performance.

3.2.2 Quantization

Quantization is a technique used to reduce the computational and memory demands of a neural network by representing weights and activations with lower precision. In this study, quantization was applied to the UNet model to further enhance its efficiency for deployment on resource-constrained devices. The following sections describe the quantization process, the types of quantization applied, and their effects on model performance.

Two types of quantization were applied: post-training quantization and quantization-aware training (QAT). Post-training quantization involves converting a trained model from high-precision (32-bit floating point) weights to lower-precision weights (such as 8-bit integers) after the training process is complete. This approach quantizes the weights and activations of the trained model without requiring retraining. While post-training quantization significantly reduces model size and inference time, there may be a slight degradation in model accuracy due to the reduced precision. However, this trade-off is often acceptable in scenarios where computational efficiency is critical.

Quantization-aware training involves simulating the effects of quantization during the training process. The model is trained with the knowledge that weights and activations will be quantized, allowing it to learn to compensate for the reduced precision. During training, fake quantization operations are inserted in the model, which mimic the effect of quantization. This allows the model to adjust and optimize its weights considering the lower precision. QAT typically results in better accuracy retention compared to post-training quantization because the model adapts to the quantization during training. The model learns robust features that are less sensitive to the reduced precision, leading to minimal performance loss.

The quantization process involved several steps. Initially, the trained UNet model was converted to a format suitable for quantization. This involved preparing the model's graph and ensuring all operations were compatible with the quantization process. Next, post-training quantization was applied, where the model's weights and activations were quantized from 32-bit floating point to 8-bit integers. This step drastically reduced the model size and improved inference speed. The quantized model was then evaluated on the test dataset to measure any changes in performance metrics such as accuracy, Dice coefficient, IoU, Recall, and Precision.

A separate version of the UNet model was trained using QAT. This involved modifying the training pipeline to include fake quantization operations. The model was trained on the same dataset with adjusted hyperparameters to ensure optimal learning under quantization constraints. The quantized model from QAT was evaluated on the test dataset, and its performance was compared to the original and post-training quantized models.

Quantization was an effective optimization technique that enhanced the efficiency of the UNet model. By reducing the precision of weights and activations, significant improvements in model size and inference speed were achieved while maintaining acceptable levels of accuracy and performance. Quantization-aware training further improved the model's robustness to reduced precision, making it an ideal choice for deployment in resource-constrained environments.

2.6 Cell Classification

The cell classification methodology employed in this study involved using the ResNet50 model to categorize THP1 cells into various categories based on their morphological features and treatment conditions. The process was divided into several key steps: data preparation, model training, validation, and performance evaluation.

2.6.1 Data Preparation

The data used for classification consisted of THP1 cell images annotated into distinct categories: 'D1_Cells', 'D2_Cells', 'D2_PAR30_5ug', 'D2_PAR30_20ug', 'D2_PAR30_50ug', and 'D2_PAR30_500ug'. Each category represented cells at different stages and treatment conditions. The images were preprocessed by resizing, normalization, and data augmentation techniques to ensure a robust dataset for training. Data augmentation included rotations, flips, and brightness adjustments to enhance the model's generalization capabilities.

2.6.2 Model Training

ResNet50, a deep convolutional neural network, was chosen for its proven effectiveness in image classification tasks. The model was initialized with weights pre-trained on the ImageNet dataset to leverage transfer learning. This approach allowed the model to utilize previously learned features, accelerating the training process and improving performance.

The training process involved fine-tuning the pre-trained ResNet50 model on the THP1 cell dataset. The final fully connected layer was replaced with a new layer tailored to the number of cell categories. The model was trained using a cross-entropy loss function and optimized with the Adam optimizer. The learning rate was carefully tuned to balance between convergence speed and stability.

2.6.3 Validation

To ensure the model's performance, a validation set was used during training. The dataset was split into training and validation subsets, with the validation set comprising

20% of the data. This allowed for continuous monitoring of the model's performance and early stopping to prevent overfitting. Key metrics such as accuracy, precision, recall, and F1-score were tracked throughout the training process.

2.6.4 Performance Evaluation

After training, the model's performance was thoroughly evaluated on a separate test set, which included images from each cell category. The evaluation metrics included overall accuracy, as well as per-category precision, recall, and F1-score. These metrics provided insights into the model's ability to correctly classify cells across different categories and treatment conditions.

The cell classification methodology utilizing ResNet50 involved meticulous data preparation, careful model training with transfer learning, continuous validation, and thorough performance evaluation. This comprehensive approach ensured accurate and reliable classification of THP1 cells, contributing to the understanding of cellular responses under various conditions.

2.7 Pipeline Development

2.7.1 Pipeline Overview

To facilitate the automated analysis of THP1 cell images, a comprehensive pipeline was developed. This pipeline handles the entire workflow from inputting the original images to calculating confluency, cell area, and classifying the cells. The following sections provide an overview of each component of the pipeline, detailing the processes involved in image preprocessing, segmentation, classification, and subsequent analysis.

Inputting Original Images: The pipeline begins with the acquisition of original THP1 cell images, captured at a resolution of 1080x1024 pixels. These images are sourced from the experimental dataset created at Epoka University. The acquired images are stored in a structured directory, ensuring efficient access and organization. Each image is named and categorized based on its experimental conditions (e.g., D1_Cells, D2_Cells, D2_PAR30_5ug, etc.).

Preprocessing: Each original image is resized into smaller patches of 128x128, 256x256, and 512x512 pixels. This resizing ensures that the images are compatible with the input dimensions of the UNet model and allows for segmentation at different resolutions. The pixel values of the images are normalized to a range of 0 to 1. Normalization stabilizes the training process and ensures that the model can effectively learn from the images without being affected by variations in pixel intensity. To enhance the diversity of the dataset and improve the model's generalization capabilities, data augmentation techniques are applied. These techniques include rotation, flipping, zooming, shifting, shearing, and brightness/contrast adjustments.

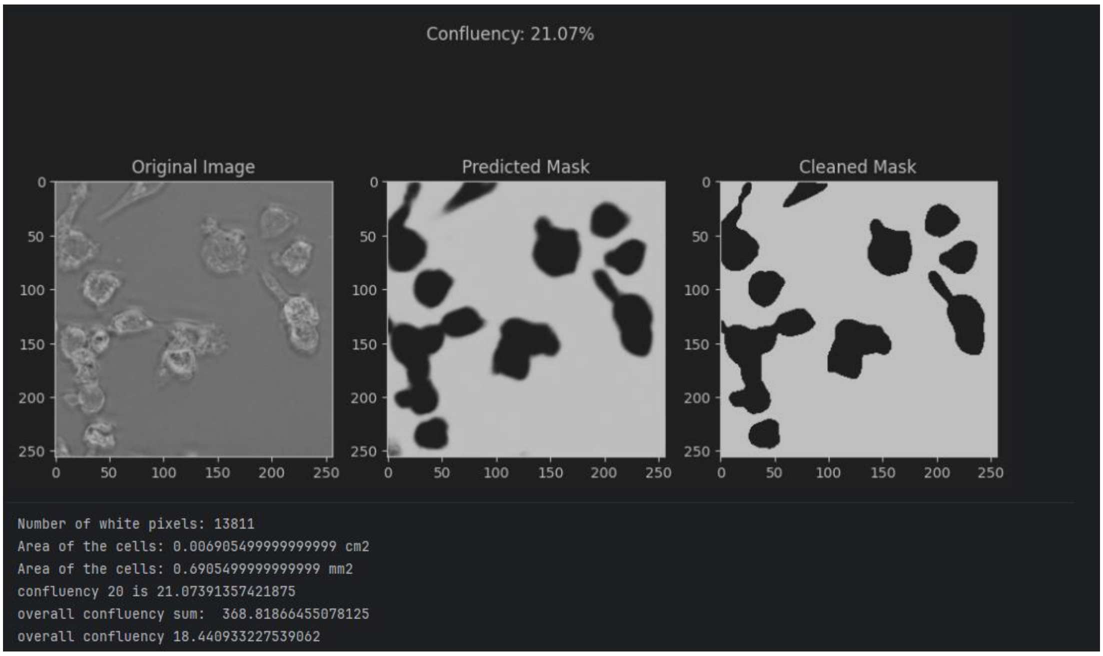
Segmentation: The preprocessed image patches are fed into the trained and optimized UNet model. The model generates segmentation masks for each patch, identifying the boundaries of the cells. The segmented patches are then stitched back together to form a complete segmentation mask for the original image. This reconstruction step ensures that the entire image is accurately segmented.

Confluency and Area Calculation: The segmented masks are analyzed to calculate the confluency and area of the cells. Confluency is determined by calculating the proportion of the image area covered by cells, while the cell area is measured by counting the number of pixels within the segmented cell boundaries. The calculated confluency and cell area values are aggregated across different experimental conditions. This aggregation allows for a comprehensive analysis of cell growth patterns, the effects of PAR30 treatments, and the overall health of the cells.

Cell Classification: To classify the cells, the pipeline utilizes the ResNet50 model. The model is fine-tuned to categorize the THP1 cells into various categories based on their morphological features and treatment conditions. The classification process involves feeding the segmented and preprocessed images into the ResNet50 model, which outputs the predicted categories. This classification step provides detailed insights into the health and condition of the cells under different experimental treatments.

Automation and Integration: The entire pipeline is automated using a series of scripts and tools. Automation ensures that the process is efficient and reproducible, minimizing manual intervention and reducing the potential for errors. The pipeline is integrated with various analysis tools to facilitate data visualization and statistical analysis. Graphical representations of confluency, cell area, and classification results are generated, providing clear insights into the experimental results.

Benefits of the Automated Pipeline: The automated pipeline significantly reduces the time and effort required for image preprocessing, segmentation, classification, and analysis. This efficiency is crucial for handling large datasets and ensuring timely results. By standardizing the preprocessing, segmentation, and classification processes, the pipeline ensures consistent and accurate results. This standardization minimizes variability and enhances the reliability of the analysis. The pipeline is designed to handle datasets of varying sizes, making it scalable for large-scale studies. Its modular structure allows for easy adaptation and extension to accommodate additional data and new experimental conditions.



```
Number of white pixels: 13811
Area of the cells: 0.006905499999999999 cm2
Area of the cells: 0.6905499999999999 mm2
confluency 20 is 21.07391357421875
overall confluency sum:  368.81866455078125
overall confluency 18.440933227539062
```

Figure 10 Example of the result of the pipeline

With this comprehensive pipeline, we can effectively analyze THP1 cell images to determine cell confluency, area, and health, as well as classify cells based on their treatment conditions, contributing valuable insights into cellular responses and experimental outcomes.

3.7.2 Integration of Model and Tools

The integration of the trained and optimized UNet model into the automated pipeline involved several steps, ensuring seamless processing of THP1 cell images from input to analysis. The following sections detail how the model was incorporated into the pipeline, including the software and tools used.

1. **Software and Tools:**
    - **TensorFlow and Keras:**
        - The UNet model was implemented and trained using TensorFlow and Keras, popular open-source deep learning libraries. TensorFlow provided the necessary framework for building, training, and optimizing the model, while Keras offered a high-level API for easy model development.
    - **OpenCV:**
        - OpenCV (Open Source Computer Vision Library) was used for image processing tasks such as resizing, normalization, and data augmentation. Its extensive set of functions and high performance made it suitable for handling the preprocessing steps efficiently.
    - **NumPy:**
        - NumPy was utilized for numerical operations, particularly for handling image data and performing calculations related to normalization and statistical analysis.
    - **TensorFlow Model Optimization Toolkit:**

    - This toolkit was used for implementing pruning and quantization, optimizing the model for deployment on resource-constrained devices. The toolkit provided the necessary functions and utilities to apply these optimization techniques effectively.
  - **Python Scripts:**
    - Custom Python scripts were developed to automate the various stages of the pipeline, from data preprocessing to model inference and analysis. These scripts ensured a streamlined and reproducible workflow.

2. **Integration Steps:**
   - **Model Loading:**
     - The trained and optimized UNet model was saved in the TensorFlow SavedModel format. This format facilitated easy loading of the model into the pipeline for inference. The model was loaded using TensorFlow functions, ensuring that it was ready for processing input images.
   - **Preprocessing Integration:**
     - The preprocessing steps, including resizing, normalization, and data augmentation, were integrated into the pipeline using OpenCV and NumPy. The Python scripts handled the preprocessing tasks, preparing the images for segmentation by the UNet model.
   - **Model Inference:**
     - The preprocessed image patches were passed to the loaded UNet model for segmentation. TensorFlow functions were used to feed the images into the model and obtain the predicted segmentation masks. This step was automated to process batches of images efficiently.
   - **Post-Processing:**
     - After obtaining the segmentation masks, the patches were stitched back together to form complete segmentation masks for the original images. OpenCV was used to handle the stitching

    and any necessary post-processing to ensure the accuracy and integrity of the segmented images.

- **Confluency and Area Calculation:**
    - The segmented masks were analyzed to calculate cell confluency and area. Custom Python functions performed pixel counting and statistical calculations, leveraging NumPy for efficient numerical operations.

3. **Automation and Workflow:**
    - **Pipeline Automation:**
        - The entire workflow, from loading images to calculating confluency and area, was automated using Python scripts. This automation ensured a consistent and efficient process, reducing manual intervention and potential errors.
    - **Integration with Analysis Tools:**
        - The results from the segmentation and analysis steps were integrated with data visualization and statistical analysis tools. Graphs and charts were generated using libraries such as Matplotlib and Seaborn, providing clear visual insights into the data.
    - **Scalability and Flexibility:**
        - The pipeline was designed to be scalable, capable of handling large datasets and multiple experimental conditions. Its modular structure allowed for easy updates and extensions, accommodating new models or additional data processing steps as needed.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Architectures Comparisons

First, we compared the models trained on different sizes of the THP1 dataset:

- **Model 1**: thp1_dataset_128
- **Model 2**: thp1_dataset_256
- **Model 3**: thp1_dataset_512
- **Model 4**: thp1_dataset_512_augmented

The training accuracy over epochs for these models is illustrated in the graph above. The results show that:

1. **Model 1 (128):** Despite being trained on the smallest dataset, this model showed decent performance with an accuracy of 0.948 and a Dice coefficient of 0.722. However, its IOU and validation metrics were relatively lower compared to the larger models.
2. **Model 2 (256):** This model achieved the highest accuracy (0.978) and showed substantial improvements in Dice coefficient (0.845) and IOU (0.733). It also maintained low loss values during training and validation phases.
3. **Model 3 (512):** Although it was trained on a larger dataset, its performance was slightly lower than Model 2 in terms of validation metrics. It had a high training accuracy (0.976) and good precision (0.937), but the validation accuracy dropped to 0.833, indicating potential overfitting.
4. **Model 4 (512 Augmented):** This model benefited from data augmentation, achieving a high Dice coefficient (0.891) and IOU (0.804). It showed balanced

performance across training and validation metrics, with the lowest loss values, indicating robust generalization capabilities.

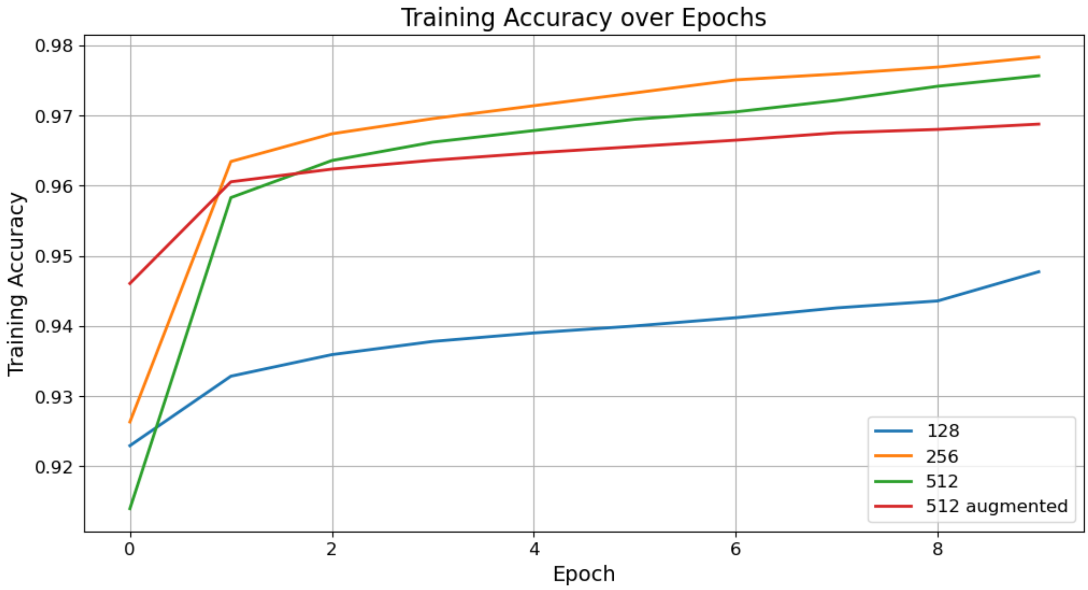


Figure 11 Training accuracy of the base Unet model on different dataset sizes.

The following table provides a summary of the performance metrics for the models on both training and validation datasets:

Table 1 Segmentation Model Performance Comparison Table for Different Image Sizes

Metric	Model 1 (128)	Model 2 (256)	Model 3 (512)	Model 4 (512 Augmented)
Accuracy	0.948	0.978	0.976	0.969
Dice Coefficent	0.722	0.845	0.835	0.891
IOU	0.573	0.733	0.717	0.804
Loss	0.155	0.070	0.080	0.055

Precision	0.874	0.934	0.937	0.953
Recall	0.793	0.930	0.913	0.900
Val Accuracy	0.939	0.971	0.833	0.965
Val Dice Coefficent	0.739	0.830	0.087	0.881
Val IOU	0.600	0.710	0.045	0.788
Val Loss	0.193	0.088	0.520	0.067
Val Precision	0.802	0.933	0.000	0.947
Val Recall	0.809	0.887	0.000	0.885

These findings highlight the effectiveness of larger input sizes and data augmentation in improving model performance.

Here are some visual results:

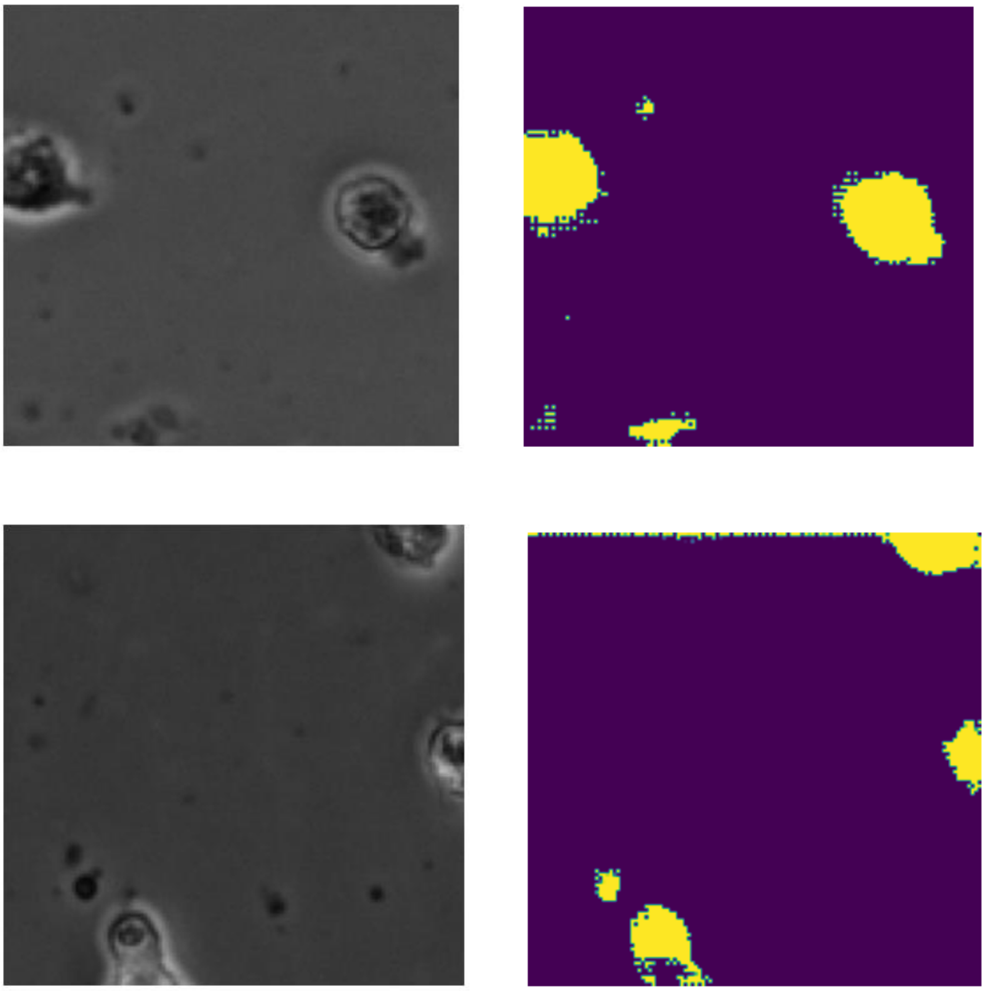


Figure 12 Examples of the 128x128 model with 10 epochs

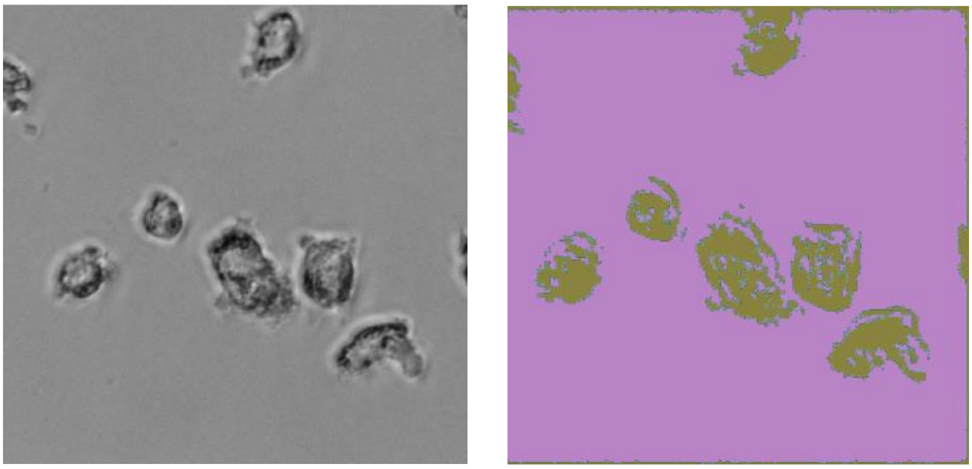


Figure 13 Example of 256x256 model with 15 epochs

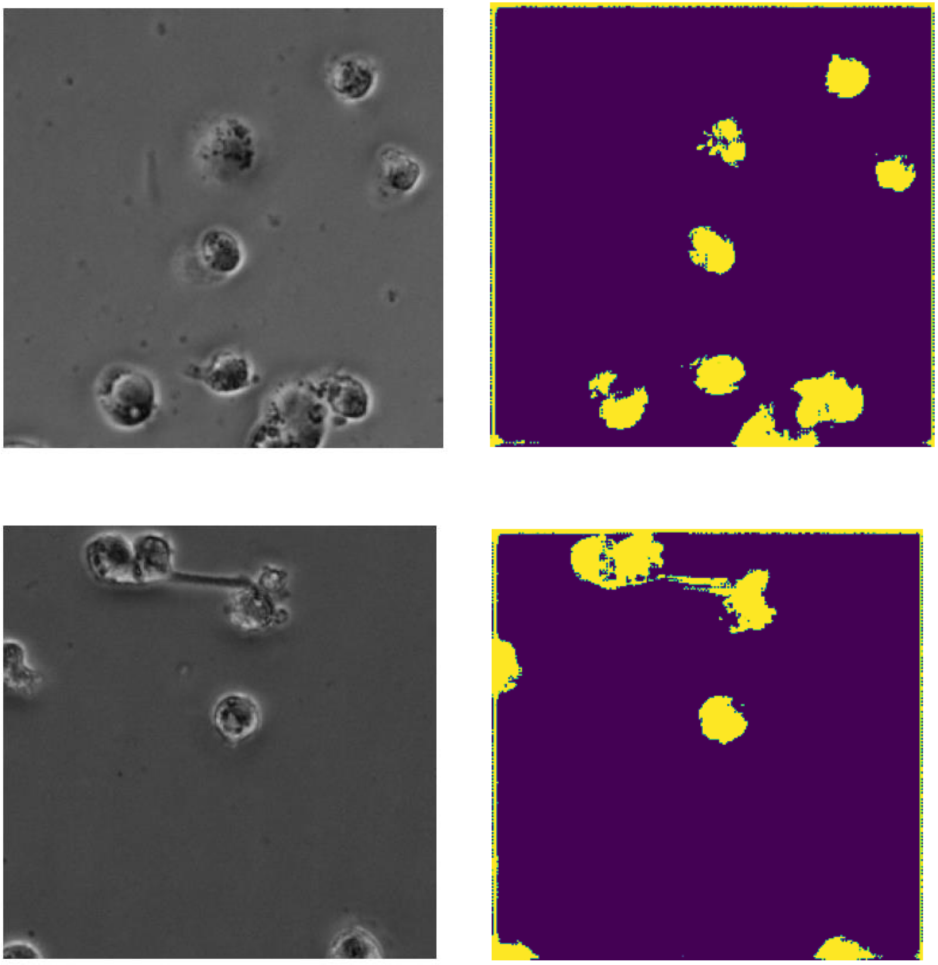


Figure 14 Examples of 256x256 model with 10 epochs

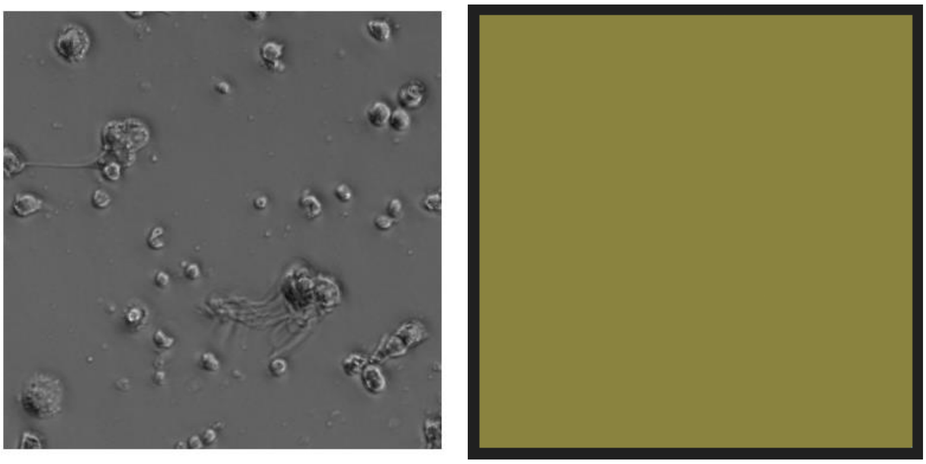


Figure 15 Example of 512x512 model with 10 epochs

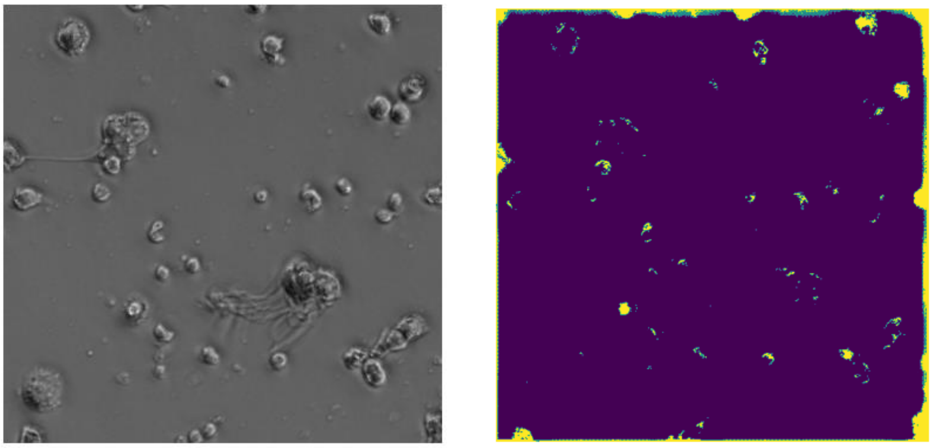


Figure 16 Example of 512x512 augmented with 10 epochs

Based on these results, we continued our work with the 256x256 dataset, which showed the highest accuracy and robust performance.

In this study, we compared the performance of four different U-Net models trained on various sizes of the THP1 dataset. The models and their configurations are as follows:

- **Model 1**: Basic U-Net
- **Model 2**: Basic U-Net with dropout (0.3)
- **Model 3**: U-Net with attention mechanism
- **Model 4**: Basic U-Net with dropout (0.3) and VGG16 encoder

The training accuracy over epochs for these models is shown in the graph. The training results indicate the following:

- **Model 1 (**Basic U-Net**)**: Showed steady improvement over the epochs but had the lowest accuracy among the four models, reaching approximately 0.93 at the end of training.
- **Model 2 (**Basic U-Net with dropout (0.3)**)**: Achieved the highest training accuracy of around 0.98, outperforming the other models. This model benefited

from the dropout mechanism, which likely helped in regularization and preventing overfitting.

- **Model 3 (**U-Net with attention mechanism**)**: Also performed well with an accuracy of around 0.97. The attention mechanism incorporated in this model likely contributed to its high performance by allowing the model to focus on relevant features during training.

- **Model 4 (**Basic U-Net with dropout (0.3) and VGG16 encoder**)**: Reached a similar accuracy to Model 3, around 0.97. The combination of dropout and VGG16 encoder provided robust feature extraction capabilities, enhanced by the augmentation process.

These results underscore the value of advanced architectures in achieving high performance in U-Net models for cell segmentation tasks.

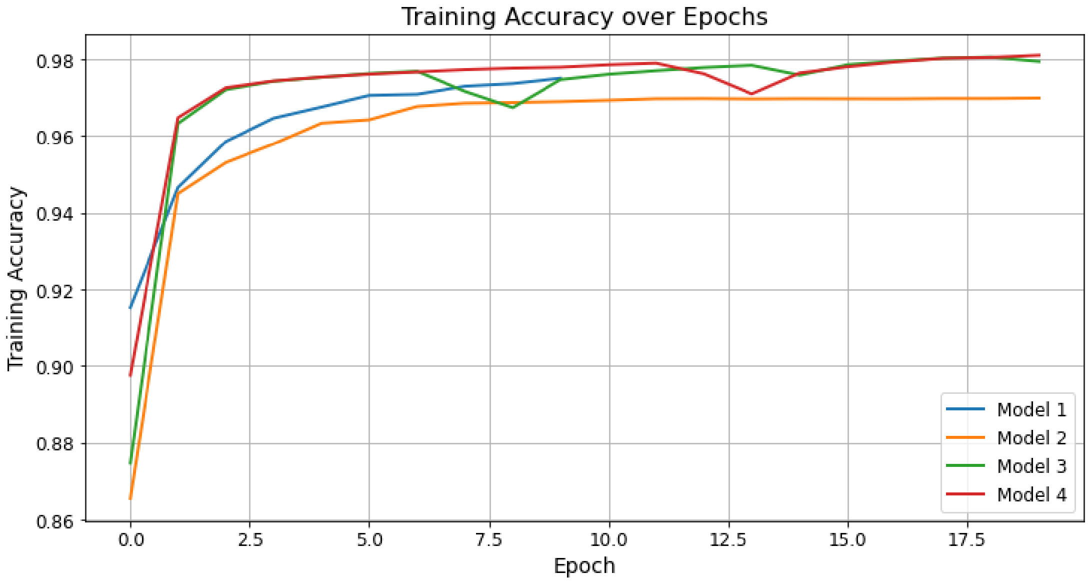


Figure 17 The best model metrics performance overview

The following table provides a summary of the performance metrics for the models on both training and validation datasets:

Table 2 Segmentation Model Performance Comparison Table for Different Architectures

Metric	Model 1	Model 2	Model 3	Model 4
Accuracy	0.97	0.97	0.97	**0.98**
Dice Coefficient	0.77	0.74	**0.79**	**0.79**
IoU	0.63	0.59	**0.66**	**0.66**
Loss	0.1	0.12	**0.09**	**0.09**
Precision	0.9	0.9	**0.93**	**0.93**
Recall	0.89	0.86	**0.91**	**0.91**
Val Accuracy	0.84	0.84	**0.97**	**0.97**
Val Dice Coefficient	0.14	0.04	**0.79**	0.72
Val IoU	0.08	0.02	**0.66**	0.57
Val Loss	0.47	0.62	**0.09**	**0.09**
Val Precision	**0.99**	0	0.94	0.93
Val Recall	0.03	0	**0.91**	0.86

4.2 Optimization Techniques: Pruning and Quantization

To further optimize the models, we applied pruning and quantization techniques.

Pruning:

- **Objective**: Reduce model size and improve inference speed.
- **Results**:
  - Model size was reduced from 355 MB to 100 MB.
  - Inference speed showed a noticeable increase, making the model more efficient for real-time applications.

Quantization:

- **Objective**: Further compress the model for edge deployment.
- **Results**:

- Model size was further reduced from 355 MB to 35 MB.
- The quantized model retained the best accuracy based on visual assessment, ensuring that compression did not significantly compromise model performance.

Here are some visual results:

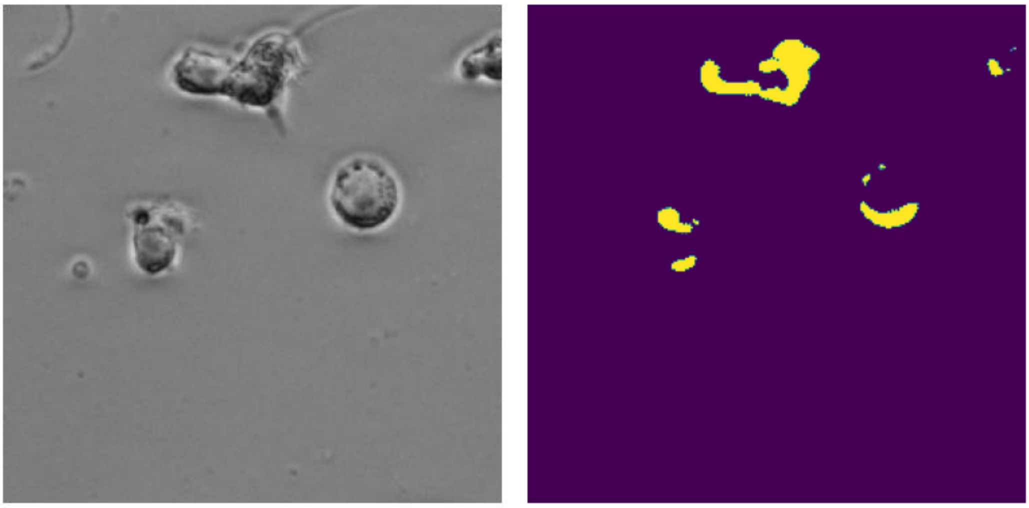


Figure 18 Example of pruning model

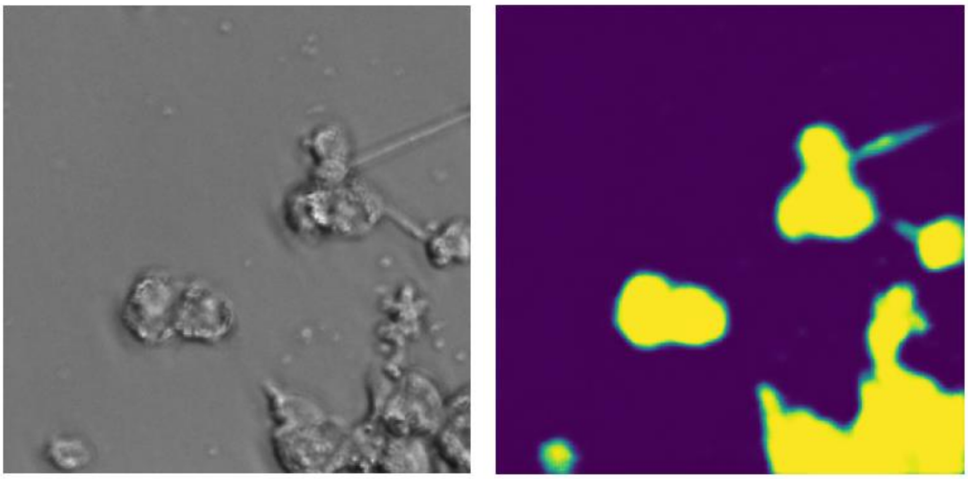


Figure 19 Example of Quantization model

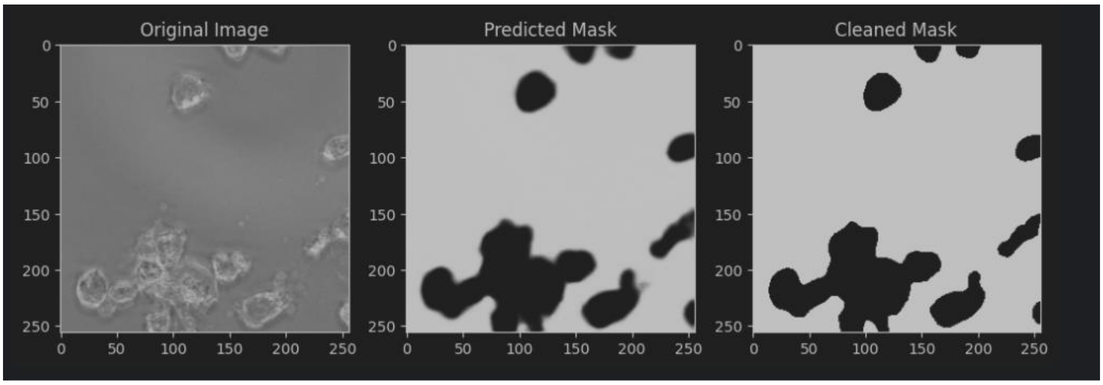


Figure 20 Example of Quantization model + Cleaning

4.3 Confluency and Area Analysis

We evaluated the confluency of cells from different image types by analyzing 10 fully annotated images for each type. The average confluency values and their standard deviations (SD) were calculated, and outliers were removed to generate a final confluency report.

Confluency Results:

Table 3 Confluency Analysis Table

Type	AVG Confluency	SD	RM Outliers	SD	Report	Status
D1	15.07	0.029				
D2	25.26	13.545	19.658	7.023		
D2_PAR30_5ug	16.93	4.815	15.75	3.24	0.801	Healthy
D2_PAR30_20ug	22.27	11.387	19.75	6.794	1.004	Healthy
D2_PAR30_50ug	14.17	4.441			0.721	Healthy
D2_PAR30_500ug	0.03	0.008			0.005	Unhealthy

The confluency results indicate that cells treated with lower doses of PAR30 (5ug, 20ug and 50ug) were generally healthy, while cells treated with higher doses (500ug) showed significantly lower confluency, indicating poor health.

4.5 Binary Classification Results

We conducted binary classification tests using different models and evaluated their performance based on true predictions and accuracy percentages.

Binary Classification Accuracy:

Table 4 Classification Accuracy Results

Test	True Pred	Accuracy (%)
D2_PAR30_5ug with model 5ug_vs_50ug	651/860	75.7
D2_PAR30_50ug with model 5ug_vs_50ug	650/880	73.9
D2_PAR30_5ug with model 5ug_vs_20ug	759/860	88.3
D2_PAR30_20ug with model 5ug_vs_20ug	673/860	78.3
D2_PAR30_5ug with model 5ug_vs_500ug	860/860	100
D2_PAR30_500ug with model 5ug_vs_500ug	783/860	91.1
D2_PAR30_20ug with model 20ug_vs_50ug	807/860	93.8
D2_PAR30_50ug with model 20ug_vs_50ug	132/860	15.3
D2_PAR30_20ug with model 20ug_vs_500ug	858/860	99.8
D2_PAR30_500ug with model 20ug_vs_500ug	795/860	92.4
D2_PAR30_50ug with model 50ug_vs_500ug	880/880	100

D2_PAR30_500ug with model 50ug_vs_500ug	799/800	99.9

The binary classification results show high accuracy for most tests, particularly for models distinguishing between 5ug and 500ug doses. The high accuracy in these tests demonstrates the models' ability to effectively classify cells based on different treatments.

The results indicate that the U-Net models, particularly the ones with larger input sizes and augmented data, perform well in terms of accuracy, precision, and recall. Optimization techniques such as pruning and quantization significantly reduced the model size and improved inference speed, making them suitable for deployment on edge devices. The confluency calculations and binary classification tests further validated the effectiveness of these models in practical applications, showing their potential for accurately assessing cell health and treatment effects.

CHAPTER 5

CONCLUSIONS

5.1 Conclusions

In this thesis, we explored the effectiveness of different U-Net architectures and optimization techniques for the task of cell segmentation using the THP1 dataset. Our study involved training and comparing four models with various configurations and input sizes, followed by the application of pruning and quantization techniques to optimize the best-performing models.

First, we compared the models trained on different sizes of the THP1 dataset: Model 1 (thp1_dataset_128), Model 2 (thp1_dataset_256), Model 3 (thp1_dataset_512), and Model 4 (thp1_dataset_512_augmented). The training accuracy over epochs for these models revealed that Model 2 (256x256) achieved the highest accuracy of around 0.98, outperforming the other models. This model's performance underscores the importance of dropout in regularizing the model and preventing overfitting. Models trained on 512x512 input sizes, both with and without data augmentation (Models 3 and 4), also demonstrated high accuracy, around 0.97 but still not good enough. Then we continued to do architecture changes and tested 4 different architectures, Model 1 Basic U-Net, Model 2 Basic U-Net with dropout (0.3)**,** Model 3 U-Net with attention mechanism and Model 4 Basic U-Net with dropout (0.3) and VGG16 encoder. The attention mechanism in Model 3 and the combination of dropout with the VGG16 encoder in Model 4 significantly contributed to their robust performance. In contrast, the Model 1 showed the lowest accuracy, highlighting the limitations of smaller input sizes for this task. Based on these results, we continued our work with the 256x256 dataset and with Model 4 Basic U-Net with dropout (0.3) and VGG16 encoder, which showed the highest accuracy and robust performance.

The optimization efforts using pruning and quantization were successful. Pruning reduced the model size from 355 MB to 100 MB and noticeably improved inference speed, enhancing the model's efficiency for real-time applications. Quantization further compressed the model size to 35 MB while maintaining high accuracy based on visual assessment, making it suitable for deployment on edge devices.

We also evaluated the confluency of cells treated with different concentrations of PAR30. The results indicated that cells treated with lower doses (5µg and 20µg) were generally healthy, while higher doses (500µg) significantly reduced cell confluency, indicating poor health. The binary classification tests showed high accuracy for distinguishing between different doses of PAR30, particularly for the models comparing 5µg and 500µg doses, demonstrating the models' capability in effectively classifying cells based on treatment.

The findings from this thesis have several important implications. They demonstrate that larger input sizes and data augmentation significantly improve model performance. The incorporation of dropout and advanced encoders like VGG16 further enhances the model's ability to accurately segment and classify cells. Pruning and quantization are effective techniques for reducing model size and improving inference speed without substantially compromising accuracy, making the models viable for deployment in resource-constrained environments. The ability to asssccurately calculate cell confluency and classify cell health based on treatment concentrations has practical applications in biomedical research and drug testing, providing valuable insights into cell behavior under different conditions.

In conclusion, this thesis has demonstrated the potential of advanced U-Net models and optimization techniques for effective cell segmentation and classification. The insights gained from this study provide a strong foundation for future advancements in this field, contributing to the broader application of AI in biomedical research.

5.2 Recommendations for future research

Future research could explore the analysis of eccentricity, a valuable metric for understanding the morphology of THP1 cells, which can provide insights into their differentiation, activation, and response to treatments. In the context of drug testing, eccentricity can be used to evaluate the effect of drugs on THP1 cells. Changes in cell shape after drug treatment can indicate efficacy or cytotoxicity. To achieve this, future work should focus on instance cell segmentation to separate merged cells, which will facilitate the calculation of cell eccentricity.

REFERENCES

[1] E. Meijering, "Cell Segmentation: 50 Years Down the Road [Life Sciences]," in IEEE Signal Processing Magazine, vol. 29, no. 5, pp. 140-145, Sept. 2012, doi: 10.1109/MSP.2012.2204190.

[2] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," in IEEE Transactions on Systems, Man, and Cybernetics, vol. 9, no. 1, pp. 62-66, Jan. 1979, doi: 10.1109/TSMC.1979.4310076.

[3] J. Canny, "A Computational Approach to Edge Detection," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-8, no. 6, pp. 679-698, Nov. 1986, doi: 10.1109/TPAMI.1986.4767851.

[4] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 13, no. 6, pp. 583-598, June 1991, doi: 10.1109/34.87344.

[5] Y. Bengio, A. Courville and P. Vincent, "Representation Learning: A Review and New Perspectives," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 8, pp. 1798-1828, Aug. 2013, doi: 10.1109/TPAMI.2013.50. [6] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. Medical Image Computing and Computer-Assisted Intervention (MICCAI), 234-241.

[7] Cooper, G. M., & Hausman, R. E. (2013). The Cell: A Molecular Approach. Sinauer Associates.

[8] Bray, M. A., Fraser, A. N., Hasaka, T. P., Carpenter, A. E. (2012). Workflow and metrics for image quality control in large-scale high-content screens. *Journal of Biomolecular Screening*, 17(2), 266-274.

[9] Lutolf, M. P., & Hubbell, J. A. (2005). Synthetic biomaterials as instructive extracellular microenvironments for morphogenesis in tissue engineering. *Nature Biotechnology*, 23(1), 47-55.

[10] Fuchs, T. J., & Buhmann, J. M. (2011). Computational pathology: Challenges and promises for tissue analysis. *Computerized Medical Imaging and Graphics*, 35(7-8), 515-530.

[11] Prewitt, J. M., & Mendelsohn, M. L. (1966). The analysis of cell images. Annals of the New York Academy of Sciences, 128(3), 1035-1053.

[12] Xing, F., & Yang, L. (2016). Robust nucleus/cell detection and segmentation in digital pathology and microscopy images: A comprehensive review. IEEE Reviews in Biomedical Engineering, 9, 234-263.

[13] Coelho, L. P. (2012). Mahotas: Open source software for scriptable computer vision. Journal of Open Research Software, 1(1), e3.

[14] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.

[15] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

[16] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, 25, 1097-1105.

[17] Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., ... & van der Laak, J. A. W. M. (2017). A survey on deep learning in medical image analysis. Medical Image Analysis, 42, 60-88.

[18] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... & Zhang, X. (2016). TensorFlow: A system for large-scale machine learning. 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), 265-283.

[19] Gonzalez, R. C., & Woods, R. E. (2002). Digital Image Processing. Prentice Hall.

[20] Prewitt, J. M. (1970). Object enhancement and extraction. Picture Processing and Psychopictorics, 75-149.

[21] Haralick, R. M., & Shapiro, L. G. (1992). Computer and Robot Vision. Addison-Wesley.

[22] Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? Advances in Neural Information Processing Systems, 27, 3320-3328.

[23] Wang, S., Su, Z., Ying, L., Peng, R., Zhu, S., Liang, F., ... & Liang, D. (2016). Accelerating magnetic resonance imaging via deep learning. IEEE International Symposium on Biomedical Imaging (ISBI), 514-517.

[24] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770-778.

[25] Drozdzal, M., Vorontsov, E., Chartrand, G., Kadoury, S., & Pal, C. (2016). The importance of skip connections in biomedical image segmentation. Deep Learning and Data Labeling for Medical Applications (Lecture Notes in Computer Science), 179-187.

[26] Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., & Ronneberger, O. (2016). 3D U-Net: Learning dense volumetric segmentation from sparse annotation. Medical Image Computing and Computer-Assisted Intervention (MICCAI), 424-432.

[27] Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., & Liang, J. (2018). UNet++: A nested U-Net architecture for medical image segmentation. Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support, 3-11.

[28] Milletari, F., Navab, N., & Ahmadi, S. A. (2016). V-Net: Fully convolutional neural networks for volumetric medical image segmentation. 2016 Fourth International Conference on 3D Vision (3DV), 565-571.

[29] Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. Journal of Big Data, 6(1), 60.

[30] Setio, A. A. A., Traverso, A., de Bel, T., Berens, M. S., van den Bogaard, C., Cerello, P., ... & Jacobs, C. (2017). Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: The LUNA16 challenge. Medical Image Analysis, 42, 1-13.

[31] Kamnitsas, K., Ledig, C., Newcombe, V. F., Simpson, J. P., Kane, A. D., Menon, D. K., ... & Rueckert, D. (2017). Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. Medical Image Analysis, 36, 61-78.

[32] Fu, H., Cheng, J., Xu, Y., Zhang, C., Wong, D. W. K., Liu, J., & Cao, X. (2018). Disc-aware ensemble network for glaucoma screening from fundus image. IEEE Transactions on Medical Imaging, 37(11), 2493-2501.

[33] Christ, P. F., Elshaer, M. E. A., Ettlinger, F., Tatavarty, S., Bickel, M., Bilic, P., ... & Menze, B. H. (2016). Automatic liver and lesion segmentation in CT using

cascaded fully convolutional neural networks and 3D conditional random fields. International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), 415-423.

[34] Janowczyk, A., & Madabhushi, A. (2016). Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases. Journal of Pathology Informatics, 7, 29.

[35] Bai, W., & Rueckert, D. (2015). A unified framework for joint segmentation and motion estimation from cardiac MR sequences. IEEE Transactions on Medical Imaging, 34(3), 513-523.

[36] Valindria, V. V., Pawlowski, N., Rajchl, M., Lavdas, I., Aboagye, E. O., Rockall, A. G., ... & Glocker, B. (2018). Multi-modal learning from unpaired images: Application to multi-organ segmentation in CT and MRI. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 7285-7294.

[37] Tsuchiya, S., Yamabe, M., Yamaguchi, Y., Kobayashi, Y., Konno, T., & Tada, K. (1980). Establishment and characterization of a human acute monocytic leukemia cell line (THP-1). International Journal of Cancer, 26(2), 171-176.

[38] Chanput, W., Mes, J. J., & Wichers, H. J. (2014). THP-1 cell line: An in vitro cell model for immune modulation approach. International Immunopharmacology, 23(1), 37-45.

[39] Park, E. K., Jung, H. S., Yang, H. I., Yoo, M. C., Kim, C., & Kim, K. S. (2007). Optimized THP-1 differentiation is required for the detection of responses to weak stimuli. Inflammation Research, 56(1), 45-50.

[40] Auwerx, J. (1991). The human leukemia cell line, THP-1: A multifacetted model for the study of monocyte-macrophage differentiation. *Experientia*, 47(1), 22-31.

[41] Chanput, W., Peters, V., Wichers, H. J., & Schols, H. A. (2012). THP-1 and U937 cells. In Encyclopedia of Immune Responses (pp. 2662-2667). Springer.

[42] Daigneault, M., Preston, J. A., Marriott, H. M., Whyte, M. K. B., & Dockrell, D. H. (2010). The identification of markers of macrophage differentiation in PMA-stimulated THP-1 cells and monocyte-derived macrophages. PLoS ONE, 5(1), e8668.

[43] Stokes, K. L., Chi, M. H., Sakamoto, K., Newcomb, D. C., Currier, M. G., Huckabee, M. M., ... & Peebles, R. S. (2011). Differential pathogenesis of

respiratory syncytial virus clinical isolates in BALB/c mice. Journal of Virology, 85(12), 5782-5793.

[44] Arrouchi, N. A., Zaid, Y., Dubé, M., & Flamand, L. (2014). THP-1 cells as a model to study human cytomegalovirus virulence and therapeutic options. Journal of Virological Methods, 207, 134-140.

[45] Savary, K., Caglayan, D., Caja, L., Tzavlaki, K., Binnewies, M., Nervi, B., ... & Heldin, C. H. (2018). THP-1 and HL-60 promyelocytic leukemia cells provide targets for the treatment of acute myeloid leukemia. Cancer Research, 78(15 Supplement), 1161-1161.

[46] Franchi, S., Sacchetti, G., Moretti, S., Gerli, F., Borgonovo, G., Petrini, P., & Colonna, M. (2017). Toxicity of different nanoparticles in THP-1 cells: An in vitro study. Journal of Nanoparticle Research, 19(7), 1-10.

[47] Beumer, J. H., Yang, Z., & Capodagli, G. (2014). THP-1 as a model for high-throughput screening in drug discovery. Current Protocols in Chemical Biology, 6(1), 30-36.

[48] Nishizawa, M., Iba, H., Miyazawa, T., & Yoshimura, A. (2016). Epigenetic modification during macrophage differentiation. Journal of Leukocyte Biology, 99(4), 717-724.

[49] Han, S., Pool, J., Tran, J., & Dally, W. J. (2015). Learning both weights and connections for efficient neural network. Advances in Neural Information Processing Systems, 28, 1135-1143.

[50] LeCun, Y., Denker, J. S., & Solla, S. A. (1990). Optimal brain damage. Advances in Neural Information Processing Systems, 2, 598-605.

[51] Li, H., Kadav, A., Durdanovic, I., Samet, H., & Graf, H. P. (2017). Pruning filters for efficient convnets. International Conference on Learning Representations (ICLR).

[52] Zhu, M., & Gupta, S. (2018). To prune, or not to prune: Exploring the efficacy of pruning for model compression. International Conference on Learning Representations (ICLR).

[53] Molchanov, P., Tyree, S., Karras, T., Aila, T., & Kautz, J. (2017). Pruning convolutional neural networks for resource efficient transfer learning. International Conference on Learning Representations (ICLR).

[54] Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., ... & Kalenichenko, D. (2018). Quantization and training of neural networks for efficient integer-arithmetic-only inference. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2704-2713.

[55] Gupta, S., Agrawal, A., Gopalakrishnan, K., & Narayanan, P. (2015). Deep learning with limited numerical precision. International Conference on Machine Learning (ICML), 1737-1746.

[56] Courbariaux, M., Hubara, I., Soudry, D., El-Yaniv, R., & Bengio, Y. (2016). Binarized neural networks: Training deep neural networks with weights and activations constrained to +1 or -1. Advances in Neural Information Processing Systems, 29, 4107-4115.

[57] Wu, J., Leng, C., Wang, Y., Hu, Q., & Cheng, J. (2016). Quantized convolutional neural networks for mobile devices. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 4820-4828.

[58] Krishnamoorthi, R. (2018). Quantizing deep convolutional networks for efficient inference: A whitepaper. arXiv preprint arXiv:1806.08342.

[59] Falk, T., Mai, D., Bensch, R., Çiçek, Ö., Abdulkadir, A., Marrakchi, Y., ... & Jug, F. (2019). U-Net: deep learning for cell counting, detection, and morphometry. Nature Methods, 16(1), 67-70.

[60] He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[61] Xie S, Girshick R, Dollár P, Tu Z, He K. Aggregated residual transformations for deep neural networks. InProceedings of the IEEE conference on computer vision and pattern recognition 2017 (pp. 1492-1500).

[62] He K, Zhang X, Ren S, Sun J. Identity mappings in deep residual networks. InComputer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14 2016 (pp. 630-645). Springer International Publishing.

[63] Xie L, Chen X, Bi K, Wei L, Xu Y, Wang L, Chen Z, Xiao A, Chang J, Zhang X, Tian Q. Weight-sharing neural architecture search: A battle to shrink the optimization gap. ACM Computing Surveys (CSUR). 2021 Oct 7;54(9):1-37.

[64] Wang J, Zhu H, Wang SH, Zhang YD. A review of deep learning on medical image analysis. Mobile Networks and Applications. 2021 Feb;26(1):351-80.

[65] X. Polisi, A. N. Halili, A. Uka and C. Ciulla, "Two-Stage Unsupervised Classification of Cell Health," *2023 International Conference on Computing, Electronics & Communications Engineering (iCCECE)*, Swansea, United Kingdom, 2023, pp. 145-149, doi: 10.1109/iCCECE59400.2023.10238637.

[66] X. Polisi, D. Avdiu, A. Uka, A. N. Halili, K. Kollcaku and C. Ciulla, "Evaluation of Cell Segmentation Using Pruning and Quantization," *2023 International Conference on Computing, Electronics & Communications Engineering (iCCECE)*, Swansea, United Kingdom, 2023, pp. 139-144, doi: 10.1109/iCCECE59400.2023.10238641.

[67] Uka, A., Ndreu Halili, A., Polisi, X., Topal, A. O., Imeraj, G., & Vrana, N. E. (2021). Basis of image analysis for evaluating cell biomaterial interaction using brightfield microscopy. *Cells Tissues Organs*, *210*(2), 77-104.

[68] Uka, A., Tare, A., Polisi, X., & Panci, I. (2020, December). FASTER R-CNN for cell counting in low contrast microscopic images. In *2020 International Conference on Computing, Networking, Telecommunications & Engineering Sciences Applications (CoNTESA)* (pp. 64-69). IEEE.

[69] Uka, A., Polisi, X., Barthes, J., Halili, A. N., Skuka, F., & Vrana, N. E. (2020, August). Effect of Preprocessing on Performance of Neural Networks for Microscopy Image Classification. In *2020 International Conference on Computing, Electronics & Communications Engineering (iCCECE)* (pp. 162-165). IEEE.