EVALUATING THE EFFECT OF BIOMATERIALS ONTO A549 CELL USING DEEP LEARNING

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

BY

GRISELDA ALUSHLLARI

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 **"Evaluating the Effect of Biomaterials onto A549 Cell Using Deep Learning"** 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. Dimitrios Karras (Computer Engineering) _________________

Assoc. Prof. Dr. Arban Uka (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: Griselda Alushllari

Signature: ______________

ABSTRACT

EVALUATING THE EFECT OF BIOMATERIALS ONTO A549 CELL USING DEEP LEARNING

Alushllari, Griselda
M.Sc., Department of Computer Engineering
Supervisor: Assoc. Prof. Dr. Arban Uka

This study investigates the performance of 28 different UNet models for segmenting and determining cell confluence in brightfield microscopy images, combining various hyperparameters such as loss functions, batch sizes, and epochs. Ground truths for the images were manually annotated which was another challenge of this study. Among the models, two of them were chosen since they achieved high accuracy results. The study also evaluates the effects of different biomaterial density on cell growth using these models. The results showed that low-density biomaterials (5 ug) were non-toxic, while medium (20 ug) and high concentrations (50 ug for PAR30 and 500 ug for PLL250) significantly suppress cell growth, with confluence ratios dropping below 70%. Additionally, various classification models were tested on datasets with different cell images and biomaterial densities. Principal Component Analysis (PCA) and hybrid models were found to significantly improve classification accuracy, particularly in binary classification tasks, which achieved accuracies nearing 98%. The study highlights the performance of different model architectures, manual annotation for ground truth, and dimensionality reduction techniques in enhancing the accuracy of cell confluence segmentation and biomaterial toxicity assessment.

Keywords: *A549 Cell, Image Analysis, Brightfield Images, Manual Annotation, Segmentation, Classification.*

ABSTRAKT

VLERËSIMI I EFEKTIT TË BIOMATERIALEVE NË QELIZËN A549 DUKE PËRDORUR DEEP LEARNING

Alushllari, Griselda

Master Shkencor, Departamenti i Inxhinierisë Kompjuterike

Udhëheqësi: Assoc. Prof. Dr. Arban Uka

Ne kete studim vleresohet performanca e 28 modeleve UNET per te segmentuar qelizat dhe per gjetur siperfaqen qe ze qeliza ne imazh (confluence), per imazhet te qelizava te pa ngjyrosura duke kombinuar parametra te ndryshme. Ground truths per imazhet jane bere manualisht (anotim manual), gje qe ishte nje sfide shume e madhe e ktij studimi. Nder modelet e krijuara, dy prej tyre u zgjodhen pasi arriten vlera te larta saktesie. Bazuar ne segmentim dhe gjetjen e konfluences, studimi gjithashtu vlereson efektet e densiteteve te ndryshme te biomaterialeve te injektuara ne qelize per te vezhguar cfare ndodh me qelizen (shumim ose demtim). Rezultatet treguan se biomaterialet me densitet te ulet (5 ug) ishin jo-toksike, ndersa biomaterialet me perqendrim mesatar (20 ug) dhe te larte (50 ug per PAR30 dhe 500 ug per PLL250) ndalonin rritjen e qelizes, duke e demtuar ate dhe duke cuar perqindjen e konfluences poshte 70% (duke theksuar qe 70% eshte pragu I toksicitetit). Përveç kësaj, modele të ndryshme klasifikimi u testuan ne te gjithe dataset-et. PCA dhe modelet hibride përmirësonin ndjeshëm saktësinë e klasifikimit, veçanërisht ne klasifikim binar, të cilat arritën saktësi prej 98%.

Fjalët kyçe: *Qeliza A549, Analiza e Imazheve, Imazhe pa ngjyrim, Segmentim, Klasifikim.*

ACKNOWLEDGEMENTS

I am deeply grateful for the opportunity to study at EPOKA University, an institution that has provided me with invaluable knowledge and experiences. I would like to extend my heartfelt thanks to my supervisor, Assoc. Prof. Dr. Arban UKA, for his guidance, expertise, and unwavering support in the completion of this work. Your insights and encouragement have been invaluable, Professor.

I would also like to thank my professors at the university for their support and collaboration, which have enriched my academic journey.

Lastly, I want to express my appreciation to my family. I was able to successfully complete this journey through your unwavering support, patience, and encouragement. Without your belief in me, this achievement would not have been possible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Problem Statement

The classification of cells in bright-field images is challenging due to variations in cell morphology and image quality. Effective preprocessing techniques, are essential to enhance image quality and improve the performance of deep learning models for cell segmentation and classification. This study aims to identify, implement, and optimize preprocessing techniques tailored to bright-field cell images. Additionally, the study focuses on developing a deep learning model, utilizing the UNet architecture, to accurately find the are coved by the cells differently known as cell confluence.

1.2 Thesis Objective

The objective of this thesis is to improve the accuracy and efficiency of cell classification in bright-field cell images by developing and optimizing preprocessing techniques, including quantization and pruning, tailored to these images. Based on the UNet architecture for segmentation, a cell classification model should be used.

1.3 Scope of works

The scope of work includes:

1. Identifying and analyzing preprocessing techniques suitable for enhancing image quality in bright-field cell images.
2. Implementing and optimizing preprocessing techniques to improve the performance of different models for cell classification.

3. Implement different UNet architectures for bright-field cell images segmentation.
4. Evaluating the performance of the system in terms of accuracy and efficiency compared to existing methods.
5. Finding the cell confluency for each dataset to predict the healthiness of biomaterials.

1.4 Organization of the thesis

This thesis is divided into 5 chapters. The organization is done as follows:

In Chapter 1, the problem statement, thesis objective and scope of works is presented.

Chapter 2, includes the Literature Review covering topics such as image analysis techniques, image analysis in various fields, challenges specific to bright-field images, cell types (A549, THP-1, and 3T3), preprocessing techniques, manual annotation, segmentation tasks (including UNet architecture and stacked UNet), and classification tasks.

Chapter 3 will consist of the methodology and some tests performed in this study, including data collection, implementation of preprocessing techniques, manual annotation process, UNet model development and trails.

Chapter 4 will consist of every experiment explained in the methodology.

Chapter 5 is called conclusions and will conclude every experiment discussed above.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

There are many reasons and benefits to computerize, digitalize and use Artificial Intelligence (AI) in various fields of human lives. Starting with a definition first introduced by Alan Turing in 1950; Artificial Intelligence is the simulation of intelligent behavior and critical thinking processed by computer systems. There are many subfields of AI, where Machine Learning (ML), Artificial Neural Network (ANN) and Deep Learning (DL) can be mentioned. As a subfield of AI, ML utilizes advanced statistical techniques and gives the computers the ability to learn without being specifically programmed (Arthur Samuel, 1959). In 1958, the psychologist Frank Rosenblatt invented the first ANN, also known as Perceptron. As a subfield of ML, ANN mimics the structure of the human brain by simulating the way it analyzes and processes information.

AI systems perform tasks much faster and without getting tired, leading to consistency and increased efficiency. Nowadays an enormous amount of data is generated daily, which needs to be processed, analyzed etc. It is impractical to analyze this kind of big data manually. AI systems excel at recognizing patterns and trends in data. These systems are capable of solving complex problems that may be beyond human capacity, such as optimization and simulations. One important beneficial reason we cannot forget to mention is cost reduction using AI. Automation and AI can minimize errors, optimize resources, predict equipment failures and reduce operational costs. All of these benefits serve as a tool for humans used in decision making but generally with the intention of making human lives easier.

2.2 Image Analysis

Image analysis (also called image understanding) is treated as a research area between image processing and computer vision (Gonzalez and Woods (2018). Since it falls between image processing and computer vision it can be said that a continuum without distinct boundaries is formed and it is categorized into three computerized processes: low-level, mid-level and high-level processes. Low-level processes consist of basic tasks such as noise reduction, image sharpening and contrast enhancement. The above-mentioned tasks are part of image processing, where both input and output are an image. Segmentation and classification can be considered mid-level processes. In this level, the input is an image, but output are attributes extracted (edges, contours, objects) from the imputed image.

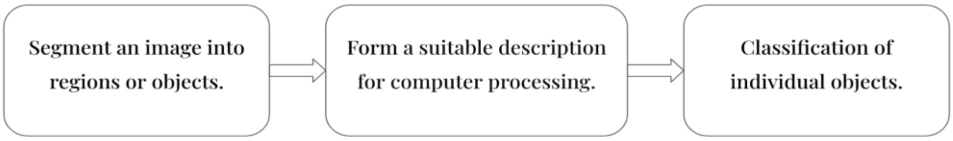

Segment an image into regions or objects.
Form a suitable description for computer processing.
Classification of individual objects.

Figure 1: Mid-Level Process Task Example

Similar to image analysis, high-level processing means to understand a group of identified objects. Compared to how humans use their vision to think and understand, complex tasks are performed at the further end of this continuum [1].

The purpose of image analysis is to use different techniques and methods to better understand and comprehend images by understanding characteristics, hidden structures or patterns of the visual data. This purpose is to obtain quantitative measurements and/or relevant feature extraction by performing different computational algorithms. Image analysis is becoming an essential part of modern research and technology due to the ongoing development of complex algorithms with the incorporation of AI.

Another point of view is to consider image analysis as the process of extracting quantitative information from acquired images (Uka, 2021) [10]. Image analysis plays an essential role evaluating medical images. While looking at cells under a microscope provides qualitative information, image analysis enables quantitative measurements of key parameters like cell number, size, shape, and spreading area. Subtle changes in cell morphology, texture, and adhesion patterns that might be missed by the naked eye can be detected and quantified through image analysis, providing deeper insights into biocompatibility and cell response. Another important characteristic is standardized image analysis protocols which ensures consistency and also helps in comparison.

2.2.1. Image Analysis Techniques

Image analysis plays an essential role evaluating cell images. Processing an image into fundamental components to extract meaningful information is its main aim. Cell segmentation and counting can be mentioned as one of the tasks of image analysis. Isolating individual cells within the image and calculating their total number it's essential for quantifying cell growth. Texture analysis techniques like Grey Level Co-occurrence Matrix (GLCM) analyzes the spatial distribution of pixel intensities, providing insights into cell density, granularity, and surface texture changes caused by biomaterial interaction (Uka, 2021) [10].

2.2.2 Image Analysis in Various Fields

The need to use artificial intelligence can be distinguished in various fields. In biology, cell image analysis helps researchers to better understand cell structure and function by making it easier to comprehend processes such as: mitosis, apoptosis, interactions etc [2]. Assessing the impact of chemical compounds on cell structures through image analysis aids in drug discovery, toxicology studies, and environmental science [2]. Based on a review article regarding AI in medical field, rather than algorithm-only-based medicine, patients have the opportunity for personalized medicine. Predictive models can be used for diagnosis of diseases, prediction of therapeutic response, and potentially preventative medicine in the future based on cell characteristics, enhancing patient outcomes and medical interventions' efficacy. Kaul, Enslin and Gross conclude that AI can improve patient diagnostics, risk classification and the efficiency of workflow. Cost-effective AI models and products are needed to integrate AI into daily clinical use, promoting a partnership between physicians and machines. (Kaul, Enslin, & Gross, 2020).

Microscopy image processing is essential in enhancing disease characterizations (Xing et al. 2018). Computers are quite fast in evaluating large datasets in a quantitative approach. This quantitative element can be distinguished in different types of tumors (breast, lung, and brain) This way, a crucial role in automated diagnosis and prediction can be mentioned. With the exponential growth of data and image data, manual processing becomes inefficient and impractical. Xing. et al, 2018 conclude that this range of new data prompts the development of computational approaches that improve efficiency and objectivity [3].

2.3 Bright-field Challenges

Another article introduces a deep-learning-powered method which employs a conditional generative adversarial neural network (cGAN) to assess bright-field images of human stem-cell-derived fat cells (adipocytes), which are important towards nanomedicine and vaccine development (Helgadottir et al., 2021). The cGAN is trained to virtually stain cellular structures (lipid droplets, cytoplasm, and nuclei) which help for a better quantitative assessment of cell structures. The method of generating virtually stained fluorescent images is less invasive, less expensive, more consistent than traditional chemical staining, and more information can be extracted from each cell. According to Helgadottir et al., 2021 fluorescence cell images have various disadvantages; including a requirement for a fluorescence microscope with filters that match the spectrum profiles of dyes, the complexity of the optical setup, and the limited number of dyes that can be seen during a single experiment. By adding chemical fluorescence dyes the cell sample can become toxic, therefore a swap may occur during image acquisition (phototoxicity and photobleaching) which negatively impacts data quality, time scales and cell health. These drawbacks make it difficult to acquire trustworthy and periodic data on the same population (for example when studying cell behavior). [4]

On the other hand, bright field images are less challenging to obtain and require no complicated or additional preparation. By replacing fluorescence images with bright-field images, the same information can be extracted which is less expensive and harmless. However, a challenge of bright-field images is the lack of clear contrast when compared to fluorescence images. Unstained brightfield images often lack high contrast between cells and background. This can make segmentation and accurate feature extraction difficult, leading to potential errors in measurements. This disadvantage limits their utility in quantitative analysis. [4] Unlike fluorescence microscopy where specific markers can highlight different cell types or structures, brightfield images provide limited information about cell identity or internal features. This limits the ability to differentiate between various cell types or assess specific cellular responses. Brightfield images can be vulnerable to dust, glare, or uneven illumination, introducing noise that can interfere with analysis and mask relevant information. The complexity of image analysis techniques, especially with large datasets, can require significant computational resources and expertise. Except the above-mentioned challenge, others drawbacks that can be mentioned are: (a) the complexity of segmenting individual cells since they vary in size, shapes and may overlap, (b) resolution issues, and (c) data standardization, otherwise it will be difficult to compare them.

Challenges cannot be mentioned without proposed solutions. To improve image quality and make the analysis easier, advanced preprocessing techniques such as: noise reduction filters, illumination correction algorithms, and background subtraction methods are used. Feature extraction using trained algorithms can overcome limitations due to low contrast (Uka,2021). Efforts to develop standardized protocols can improve comparison and image analysis.

2.4 Workflow

To have a harmonious workflow for image analysis, binary annotation, segmentation and classification are important. This combination improves the analysis of cells by allowing researchers to gather numerical data and draw conclusions.

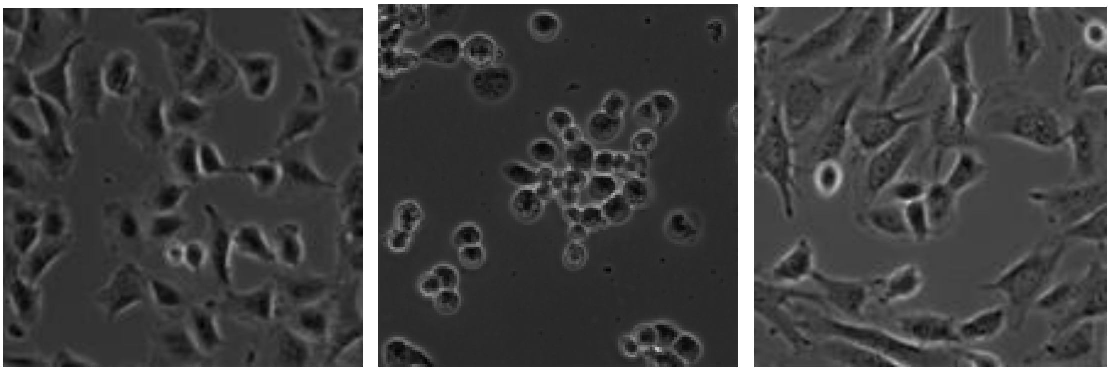
Binary annotation is an essential step in cell image analysis studies. First a binary mask should be created which distinguishes regions of interest (cells in this case) from the background. It simplifies the following analysis and aids in unique feature extraction of cellular structures. The annotation will be manually done for the images using 'apeer.com'.

Segmentation is the process of dividing an image into uniform/homogeneous regions, separating cells, and precisely identifying their boundaries. It is essential for getting precise measurements, making it possible for comprehensive analysis, cell counting and understanding how cells are organized.

Classification is the process of grouping segmented regions according to their characteristics/features such as various cell types or phases. It provides an understanding of the nature of cells, helping recognize the anomalies and measuring cell states. All of this is done for the sole purpose of better understanding the biological content of an image.

2.5 Cell Types

There will be three types of cells taken into consideration: (1) A549 is a human lung carcinoma cell line, (2) THP-1 a human monocytic cell line derived from an acute monocytic leukemia patient, and (3) 3T3 a mouse fibroblast cell line. Regarding immortalization, all three cells can divide indefinitely under laboratory conditions.



(a) A549 (b) THP-1 (c) 3T3

Figure 1: Representative images for different cells.

2.6 Preprocessing Techniques

Preprocessing is an important step when dealing with image analysis. Before applying analysis techniques, several preprocessing steps are crucial. Some common preprocessing techniques for bright-field images are: (1) Contrast Enhancement which improves detail visibility, (2) Noise Reduction Techniques like filtering which minimize signal noise that can interfere with measurements., (3) Background Subtraction which removes uneven illumination, (4) Color Correction which corrects color inconsistencies, (5) Rescaling which adjusts image size, (6) Smoothing and Sharpening which softens image features, (7) Thresholding which converts grayscale images to binary images simplifies object identification and separation, (8) Cropping which removes unnecessary borders or regions, and (9) Inversion which reverses intensity values to enhance specific features. [Uka et al, 2020]

Taking into consideration a paper titled “Brightfield vs Fluorescent Staining Dataset–A Test Bed Image Set for Machine Learning based Virtual Staining. Scientific Data” a numerous of the above-mentioned techniques are used. [7] The images were resized to 512x512 to accommodate the limitations of the neural network. Standardization and normalization were done to enhance the convergence rate and improve segmentation accuracy. Images were expanded with Gaussian noise and random smoothing. Multi-threshold segmentation was used to fit the same quantiles of the physical and virtual channel intensity distributions (Trizna et.al, 2023).

2.7 Annotation Type – Manual

According to Adnan et al., 2020, annotating digital images is important for organizing and accessing them effectively. Adding keywords tags to every image simplifies searching and improves the use and accessibility of stored data. To conclude, image annotation enables efficient management of large collections of digital images.

The type of annotation used is determined by the image annotation task's requirements and constraints. There are three types: (a) manual annotation which offers accurate descriptions, but it is time-consuming, expensive and also has subjectivity issues, (b) semi-automatic annotation which achieves a balance between manual and automatic approaches, and (c) automatic annotation which is efficient but might require massive sample sizes for learning. In this paper (Adnan et al., 2020) authors advise to combine user feedback and a semantic hierarchy in automatic annotation models, proposing the use of probabilistic graphical models. [5]

The chosen type of annotation for this thesis will be manual annotation. As every other instance, it has both benefits and drawbacks. Accurate and meaningful annotations are counted as benefits. On the other hand, some of its challenges are that it is time consuming, expensive and has subjectivity issues based on the annotator’s perspective. This can result in annotation inconsistencies, which reduces accuracy and reliability. Furthermore, manual annotation may be impractical for large datasets because of the high human work and resource requirements. [5]

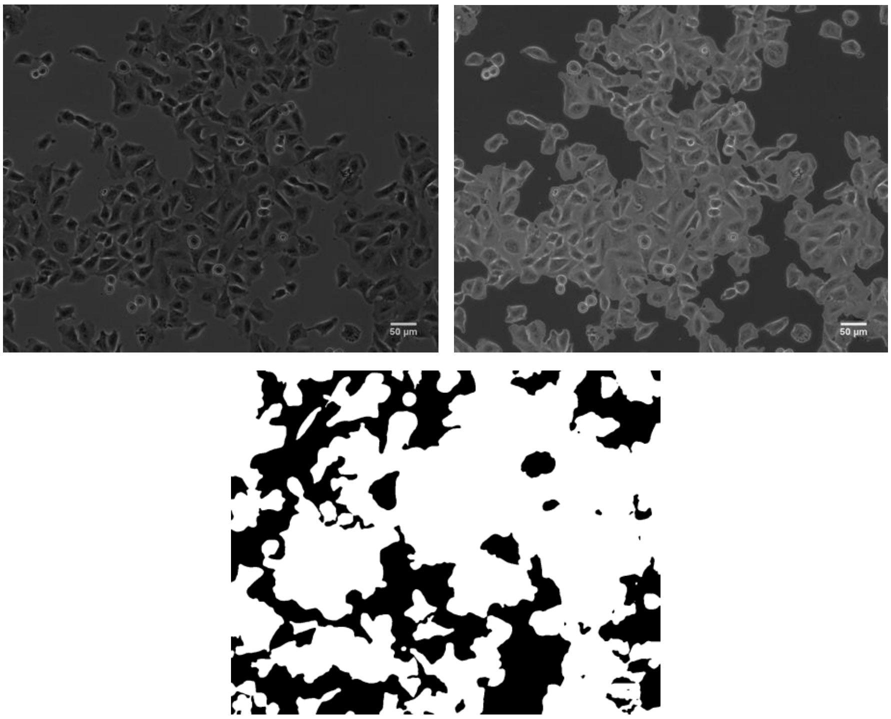


Figure 2: A549 Annotated and Raw

2.8 Segmentation Task

U-Net is a popular CNN architecture used for accurate cell segmentation in brightfield microscopy images. Inspired by the human visual cortex, it combines a contracting pathway for capturing contextual information with an expansive pathway for precise localization. The "U-shaped" structure allows the network to learn detailed features at lower resolutions and progressively recover spatial information, ultimately producing high-resolution segmentation masks.

2.8.1 UNet Architecture

The Unet architecture is a popular CNN model designed for semantic segmentation tasks in various fields such as medical imaging, satellite imagery analysis and object detection. Introduced in 2015 [6], the architecture is based on an encoder-decoder structure, which enables the acquisition of both global and local contextual information for accurate segmentation. The major innovation U-net introduces to previous models is the U-shaped design, where the encoder acquires the image’s features, the decoder generates the segmentation map, and the skip connections facilitate communication between the contracting and expansive paths. These connections enable the encoder and decoder to communicate, by enabling the decoder to access the high-resolution feature map from the encoder stages and use it to perform precise segmentation.

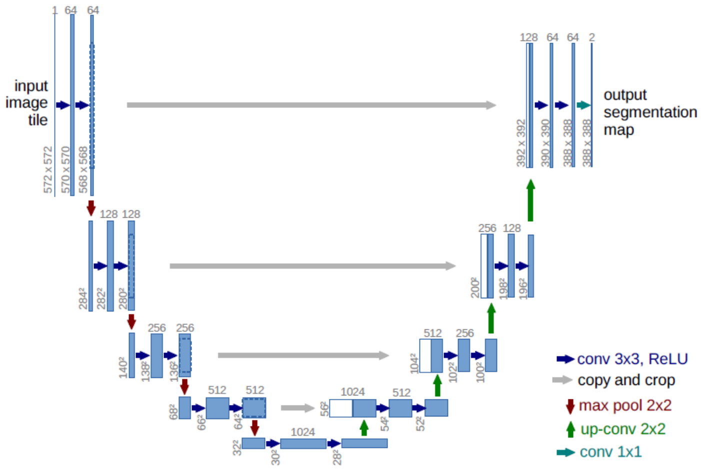


Figure 3: UNET Architecture

The architecture of the network comprises a contracting path on the left side and an expensive path on the right side:

The contracting path follows a typical convolutional network structure, consisting of repeated operations: two 3x3 convolutions with ReLU activation, and a 2x2 max pooling with a stride of 2 for down-sampling. With each down-sampling step, the number of feature channels is doubled.

The expansive path includes an up-sampling of the feature map, followed by a 2x2 convolution (referred to as "up-convolution") that reduces the number of feature channels by half. It also involves concatenating the corresponding cropped feature map from the contracting path and applying two 3x3 convolutions with ReLU activation.

The cropping is necessary to account for the loss of border pixels in each convolution operation.

The final layer of the network employs a 1x1 convolution to map each 64-component feature vector to the desired number of classes. ***Overall, the network consists of 23 convolutional layers.***

The paper demonstrates that U-Net can outperform the best method on the ISBI challenge for segmenting neuronal structures in electron microscopic stacks, using data augmentation to generate new training samples. It enhances training efficiency by using data augmentation, which generates new training samples by applying random transformations to existing ones. This reduces the need for large amounts of annotated data, improving performance on image segmentation tasks. U-Net's architecture allows precise localization through its symmetric expanding path, which captures context and reduces spatial resolution of the input image. This process consists of upsampling layers and convolutional layers, capturing local and global information for accurate segmentations. [6]

2.8.2 Stacked UNet

In the realm of cell segmentation, advancements in deep learning architectures have led to the exploration of Stacked U-Nets. Based on the UNet architecture this approach employs a hierarchical stacking mechanism. The key idea behind a Stacked UNet is to stack multiple UNets together, where the output of one UNet becomes additional input for the next. This allows the model to learn more complex features and make more accurate predictions, as each UNet can improve the predictions made by the previous ones.

Stacked U-Net is an extension of the original U-Net model designed for biomedical image segmentation. The original U-Net, introduced by Ronneberger et al. (2015), utilizes a symmetric encoder-decoder structure with skip connections to capture both high-level and low-level features. The Stacked U-Net enhances this by incorporating multiple U-Net structures in a stacked formation, allowing for more refined and accurate segmentation outputs, especially in complex medical imaging tasks.

The Stacked U-Net typically employs a series of U-Net modules where the output of one U-Net serves as the input to the next. This stacking mechanism allows for deeper feature extraction and better contextual understanding of the input images. Each U-Net in the stack maintains the traditional architecture with contracting and expanding paths, skip connections, and convolutional layers. The models often use data augmentation and patch-wise training to handle large biomedical images effectively (Ronneberger et al., 2015).

Stacked U-Nets have demonstrated significant improvements in segmentation accuracy across various biomedical imaging datasets. Studies have shown that this architecture excels in tasks such as cell segmentation, organ delineation, and tumor detection. The repeated application of the U-Net model within the stack helps in refining segmentation boundaries and reducing false positives. For instance, in liver and tumor segmentation challenges, Stacked U-Net variants have outperformed traditional U-Net models by a notable margin in metrics like Dice coefficient and Intersection over Union (IoU) (Çiçek et al., 2016).

Despite its success, the Stacked U-Net architecture can be computationally intensive due to its multiple U-Net models. This complexity might limit its application in real-time or resource-constrained environments. Future research could focus on optimizing the stack formation or introducing lightweight variants that maintain segmentation performance while reducing computational load (Ronneberger et al., 2015; Çiçek et al., 2016).

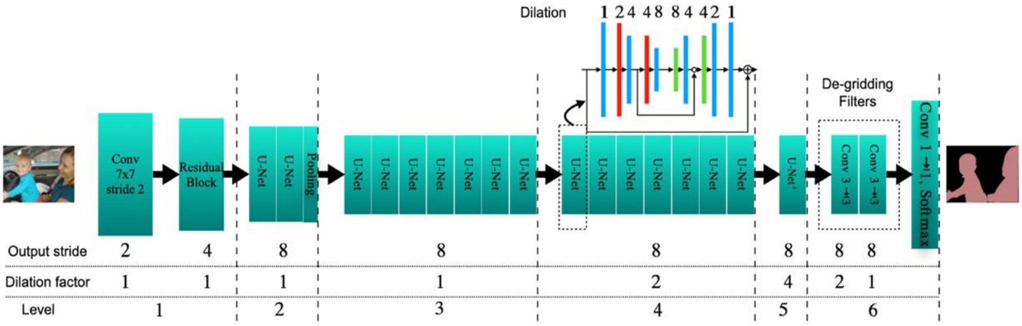


Figure 4: Stacked UNET Visualization

2.8.3 U2net

U^2-Net (pronounced "U squared net") is a novel architecture designed specifically for salient object detection. It builds on the traditional U-Net structure by introducing a nested U-structure within each stage of the encoder and decoder, enabling more detailed and hierarchical feature extraction. This model aims to accurately detect and segment salient objects in images, even in challenging backgrounds.

The U^2-Net architecture integrates the ReSidual U-block (RSU) within the standard U-Net framework. Each RSU block contains its own encoder-decoder structure, effectively creating multiple levels of nested U-Nets. This design enhances the network's capacity to capture both fine details and global context. The model utilizes standard convolutional layers, ReLU activations, and max-pooling operations within these nested structures. Training involves optimizing a combination of binary cross-entropy and intersection-over-union losses (Qin et al., 2020).

U^2-Net has achieved state-of-the-art performance in salient object detection benchmarks, significantly outperforming existing models. The architecture's ability to maintain high-resolution feature maps throughout the network contributes to its superior performance. Evaluation metrics such as F-measure, mean absolute error (MAE), and precision-recall curves indicate that U^2-Net provides precise and reliable segmentation of salient objects across various challenging datasets (Qin et al., 2020).

While U^2-Net has shown remarkable results in salient object detection, its application to other segmentation tasks such as medical imaging or autonomous driving remains to be fully explored. Additionally, the nested structure increases the model's complexity, which may hinder deployment in real-time applications. Future research could explore ways to generalize the architecture to different domains or optimize it for faster inference (Qin et al., 2020).

2.8.4 CellSegNet

CellSegNet is a modified U-Net architecture tailored for cell segmentation tasks in biomedical images. It builds on the strengths of the traditional U-Net by incorporating additional layers and mechanisms to enhance its performance in segmenting individual cells from complex and cluttered backgrounds. This architecture aims to address the unique challenges posed by cell segmentation, such as varying cell sizes, shapes, and overlapping cells.

CellSegNet typically includes modifications like additional convolutional layers, attention mechanisms, and custom loss functions designed to improve segmentation accuracy. The encoder-decoder structure of U-Net is preserved, but enhanced with additional layers for more robust feature extraction. Some variants of CellSegNet also incorporate mechanisms like squeeze-and-excitation blocks or spatial attention modules to better capture the spatial dependencies and hierarchical features within the images (Ronneberger et al., 2015; Çiçek et al., 2016).

Studies have shown that CellSegNet variants achieve superior performance in cell segmentation tasks compared to the traditional U-Net. Metrics such as the Dice coefficient, pixel accuracy, and Jaccard index have shown improvements, indicating more precise and reliable segmentation results. The incorporation of attention mechanisms, in particular, helps the network focus on relevant features and ignore background noise, which is crucial in biomedical images with high variability (Çiçek et al., 2016).

One of the primary challenges with CellSegNet is its scalability to large datasets or real-time applications due to increased model complexity. Moreover, the architecture might require extensive hyperparameter tuning and training data to generalize well across different types of cells and imaging conditions. Future research could focus on developing more lightweight variants or exploring transfer learning approaches to enhance generalizability (Ronneberger et al., 2015; Çiçek et al., 2016).

2.9 Classification Task

Deep learning models have revolutionized the field of cell classification and segmentation, particularly in the context of analyzing unstained brightfield images. Uka (2021) demonstrated that deep learning models trained on labeled fluorescence images could effectively classify cell types in unstained brightfield images based on their morphological features. This approach leverages the powerful feature extraction capabilities of deep neural networks (DNNs) to distinguish between different cell types without the need for traditional staining techniques, which can be time-consuming and costly.

The use of DNNs in image segmentation has consistently outperformed traditional segmentation techniques. These models have shown substantial improvements in various evaluation metrics, including accuracy, precision, and recall (Uka, 2023). Traditional image segmentation methods often struggle with the variability and complexity of biological images, whereas DNNs can learn and adapt to these complexities through extensive training on large datasets. The deep learning models' ability to capture intricate patterns and features within the data contributes significantly to their superior performance.

One of the significant challenges in classifying cells using deep learning is the lack of labeled datasets. Annotating large datasets for training can be labor-intensive and requires expert knowledge. To address this issue, Uka (2023) proposed a two-stage unsupervised classification approach for cell health. The first phase of this method involves dividing the dataset into two primary groups: healthy and unhealthy cells. In the second phase, the healthy group is further subdivided into two smaller clusters, while the unhealthy group is divided into three distinct clusters. These clusters are defined according to the ISO standard for in vitro cytotoxicity evaluation, ensuring that the classification aligns with established health criteria. The K-means algorithm was employed for clustering, and the approach was validated using two different datasets, demonstrating its effectiveness in organizing cells into meaningful categories without the need for labeled data.

Convolutional Neural Networks (CNNs) have also been prominently featured in image classification tasks due to their high accuracy and robustness. According to the study "Effect of Preprocessing on Performance of Neural Networks for Microscopy Image Classification," CNNs achieved an impressive accuracy of 99% when optimized with various preprocessing techniques. This underscores the importance of preprocessing in enhancing the performance of neural networks. Preprocessing steps such as normalization, augmentation, and noise reduction can significantly impact the model's ability to generalize from training data to unseen data, leading to better classification results.

2.9.1 Supervised Learning – LeNet

LeNet, one of the pioneering convolutional neural networks (CNNs) introduced by Yann LeCun and colleagues in the late 1980s, remains relevant in modern image classification tasks, particularly in the field of biomedical image analysis (LeCun et al., 1998). Its architecture, characterized by a series of convolutional and subsampling layers followed by fully connected layers, is well-suited for supervised learning applications, including the classification of cell images. In recent years, LeNet has been adapted and optimized for various cell image classification tasks, demonstrating its enduring effectiveness and flexibility.

In cell image classification, LeNet has been extensively utilized due to its ability to learn and generalize features from labeled datasets. A study by Shao et al. (2018) employed an enhanced version of LeNet to classify different types of cells in microscopy images. The model was trained on a dataset of stained cell images and successfully distinguished between various cell types based on their morphological features. This application highlighted LeNet's robustness in handling diverse cellular structures and its capability to perform accurate classification even with relatively simple architectures.

Another significant application of LeNet in cell classification was presented by Wei et al. (2019), who used the architecture for the classification of live and dead cells in fluorescence microscopy images. The study demonstrated that LeNet, when trained on a well-labeled dataset, could achieve high classification accuracy, effectively differentiating live cells from dead ones. The simplicity of LeNet's architecture allowed for efficient training and deployment, making it a practical choice for real-time cell analysis in laboratory settings.

Recent advancements have further refined LeNet's application in cell image classification. For instance, Lu et al. (2021) proposed a modified LeNet architecture for classifying cancerous and non-cancerous cells. The model incorporated additional layers and dropout mechanisms to enhance feature extraction and prevent overfitting. This approach resulted in improved classification accuracy and robustness, showcasing how LeNet can be adapted to meet the specific challenges of biomedical image classification.

In the context of unsupervised learning, LeNet has also been applied as a feature extractor before clustering methods like K-Means. For example, Wang et al. (2022) used LeNet to extract features from cell images, which were then clustered using K-Means to group similar cell types. This combination of supervised feature extraction and unsupervised clustering allowed for effective classification and grouping of cells, demonstrating the versatility of LeNet in different learning paradigms.

A comprehensive review by Ching et al. (2018) also highlighted the use of CNN architectures, including LeNet, in various biomedical imaging tasks. The review emphasized that LeNet’s architecture, despite its simplicity, continues to be effective for a wide range of classification tasks due to its ability to learn hierarchical feature representations from image data. This capability makes it particularly well-suited for tasks involving the classification of cell images, where subtle differences in morphology can be critical for accurate diagnosis and analysis.

While LeNet is effective, it does have limitations, particularly when dealing with complex or high-dimensional datasets. Its relatively shallow architecture may not capture intricate features as effectively as deeper networks like ResNet or VGG. However, the simplicity and efficiency of LeNet make it a valuable tool for supervised learning in cell image classification, especially for applications where interpretability and computational efficiency are crucial.

2.9.2 Unsupervised Learning - KMeans

K-Means clustering is a foundational unsupervised learning algorithm that has been extensively used in various image classification tasks, including the classification of cell images. It works by partitioning data into K clusters based on feature similarities, which helps in identifying intrinsic patterns without the need for labeled datasets.

In the field of biomedical imaging, K-Means has proven effective for segmenting and classifying cell images. A notable study by Yao et al. (2019) demonstrated the application of K-Means clustering for the classification of cell types in histopathological images. The researchers used K-Means to group similar cell nuclei based on their morphological features, which allowed for the differentiation between normal and abnormal cells. This approach facilitated accurate diagnosis and assessment of pathological conditions, showcasing K-Means' potential in medical diagnostics.

K-Means has also been used in conjunction with feature extraction techniques to enhance the classification accuracy of cell images. For instance, Zhang et al. (2020) employed K-Means clustering to classify live and dead cells in fluorescence microscopy images. The study involved extracting texture and intensity features from the cell images, which were then clustered using K-Means. This method achieved a high classification accuracy, effectively distinguishing live cells from dead ones, and demonstrating the algorithm's utility in biomedical applications where cell viability is a critical parameter.

Further advancements in the application of K-Means for cell image classification include its integration with deep learning frameworks. Xie et al. (2021) proposed a hybrid approach that combined K-Means with a convolutional neural network (CNN) to classify cancerous and non-cancerous cells. In this method, features extracted by the CNN were clustered using K-Means to identify distinct cell types. This integration leveraged the strengths of both unsupervised clustering and deep learning, leading to significant improvements in classification performance.

In a recent study, Li et al. (2022) utilized K-Means clustering for the classification of different cell stages in time-lapse microscopy images. The researchers used K-Means to cluster cell images based on their morphological changes over time, which enabled the identification of various stages of cell development. This approach provided valuable insights into cell dynamics and behavior, demonstrating the algorithm's versatility in handling dynamic image data.

Additionally, the study by Uka et al. (2020) investigated the impact of preprocessing on the performance of neural networks for microscopy image classification, incorporating K-Means as a key component. The researchers demonstrated that effective preprocessing techniques, such as normalization and contrast enhancement, significantly improved the clustering results obtained by K-Means, leading to better classification outcomes. This work underscores the importance of preprocessing in optimizing the performance of K-Means for unsupervised cell image classification tasks.

Another significant contribution to the field comes from Yang et al. (2021), who applied K-Means clustering for the segmentation and classification of single-cell images. The study focused on identifying various subtypes of immune cells in complex tissue samples. By clustering cell images based on their morphological features, K-

Means effectively classified different immune cell types, facilitating a deeper understanding of immune responses and disease mechanisms.

Despite its advantages, K-Means has certain limitations, such as sensitivity to the initial selection of cluster centers and the requirement to specify the number of clusters (K) beforehand. It can also struggle with high-dimensional data or datasets with clusters of varying sizes and densities. However, its simplicity and effectiveness in uncovering underlying data structures make K-Means a valuable tool for unsupervised cell image classification across a range of biomedical application.

CHAPTER 3

METHODOLOGY AND TESTS

1.1 3.1 Dataset

A549 is a commonly used human lung cancer cell that was originally obtained in 1972 from a male Caucasian patient who had lung adenocarcinoma. These cells share characteristics with the epithelial cells lining the lungs' alveoli, which are small air sacs where gas exchange takes place. Regarding the cell morphology, it consists of both (a) nucleus and (b) cytoplasm. One of the most important organelles in a cell is the nucleus. It serves a crucial role in controlling cellular functions and contains the genetic material (DNA) of the cell. The genetic instructions required for cell development, division, and function are found in the nucleus of A549 cells. The cytoplasm surrounds the nucleus and fills the cell’s interior. Numerous organelles, such as the endoplasmic reticulum, Golgi apparatus, mitochondria, and ribosomes, are found in the cytoplasm. The cytoplasm is where protein synthesis, cellular metabolism, and other vital processes occur. Crucial functions like protein synthesis, cell division, and response to external signals are carried out by A549 cell. Proper cell activity is ensured by the interaction between the cytoplasm - which houses organelles and performs biochemical reactions - and the nucleus - which houses genetic information. Considering their origin and characteristics, A549 cells play a key role in drug development, lung cancer, and respiratory research. These cells are used by researchers to examine the biology of cancer, including how the disease develops, spreads, and reacts to different therapies. A549 cells are also used in toxicology research to assess the impact of environmental and inhaled chemicals on lung tissue.

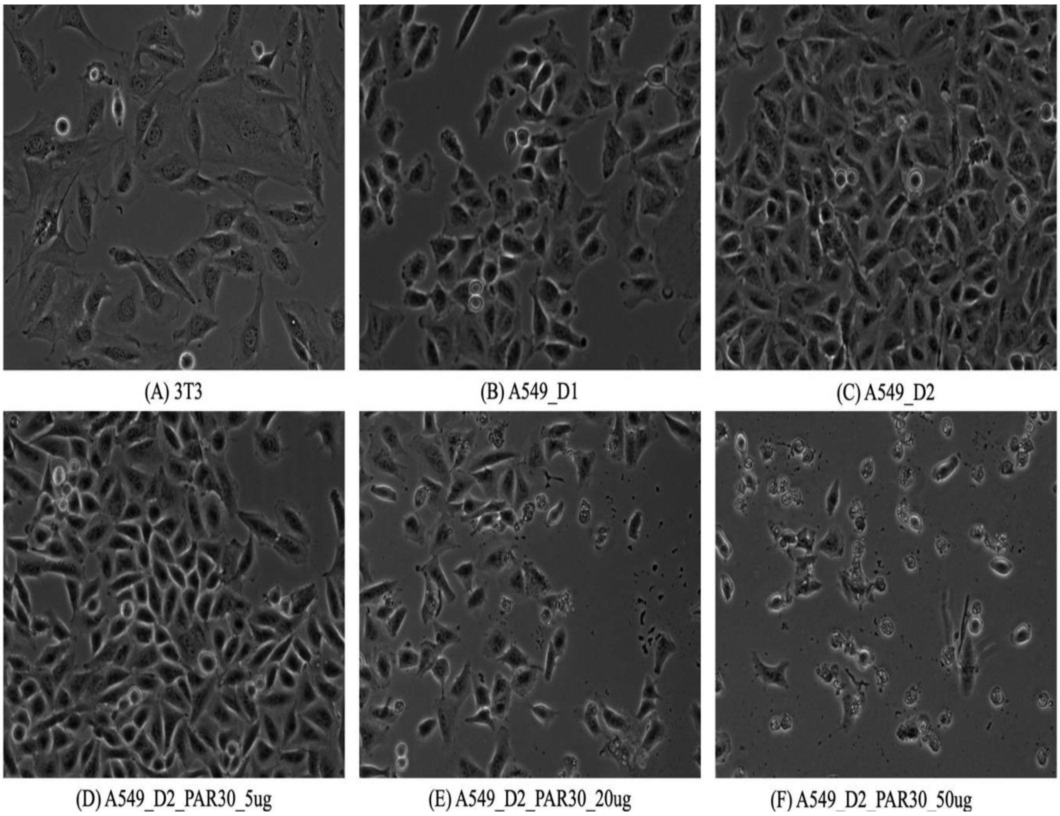
The images are acquired by a brightfield microscope, which means they are in grayscale. The size of an image is 1280x1024. The main dataset in this study consists of the A549 cell line, along with the implementation of two different biomaterials, resulting in two separate datasets: one incorporating biomaterial one and the other incorporating biomaterial two. Each dataset is organized into several folders:

- D1: Images acquired on Day 1,
- D2: Images acquired on Day 2,
- D2_BIOMATERIAL1_density: biomaterial one applied on Day 2 at a density of x µg,

A similar structure is followed for biomaterial two. Since the images are acquired by a brightfield microscopy, they are quite challenging to analyze. One reason being the low contrast between the cell cytoplasm and background, which makes it difficult to distinguish.

Dataset	***Dataset_1***		***Dataset_2***	***Dataset_3***	
Biomaterial	**PAR30**			**PLL250**	
Days	***D1***	30 images	56 images	***D1***	56 images
	D2	12 images	11 images	***D2***	11 images
	D2_5ug	12 images	10 images	***D2_5ug***	13 images
	D2_20ug	11 images	13 images	***D2_500ug***	11 images
	D2_50ug	11 images	12 images		

Table 1: Representative Table of Datasets



(A) 3T3

(B) A549_D1

(C) A549_D2

(D) A549_D2_PAR30_5ug

(E) A549_D2_PAR30_20ug

(F) A549_D2_PAR30_50ug

Figure 5: Images from the datasets

3.2 Annotation

In the absence of ground truths within the dataset, two approaches were employed to facilitate the analysis of images, focusing on semantic segmentation to divide the cells from their background. The first approach involved utilizing threshold labeling/annotation techniques. However, this method gave suboptimal results when trying to distinguish between the cytoplasm and the background particularly when they exhibited similar intensity levels.

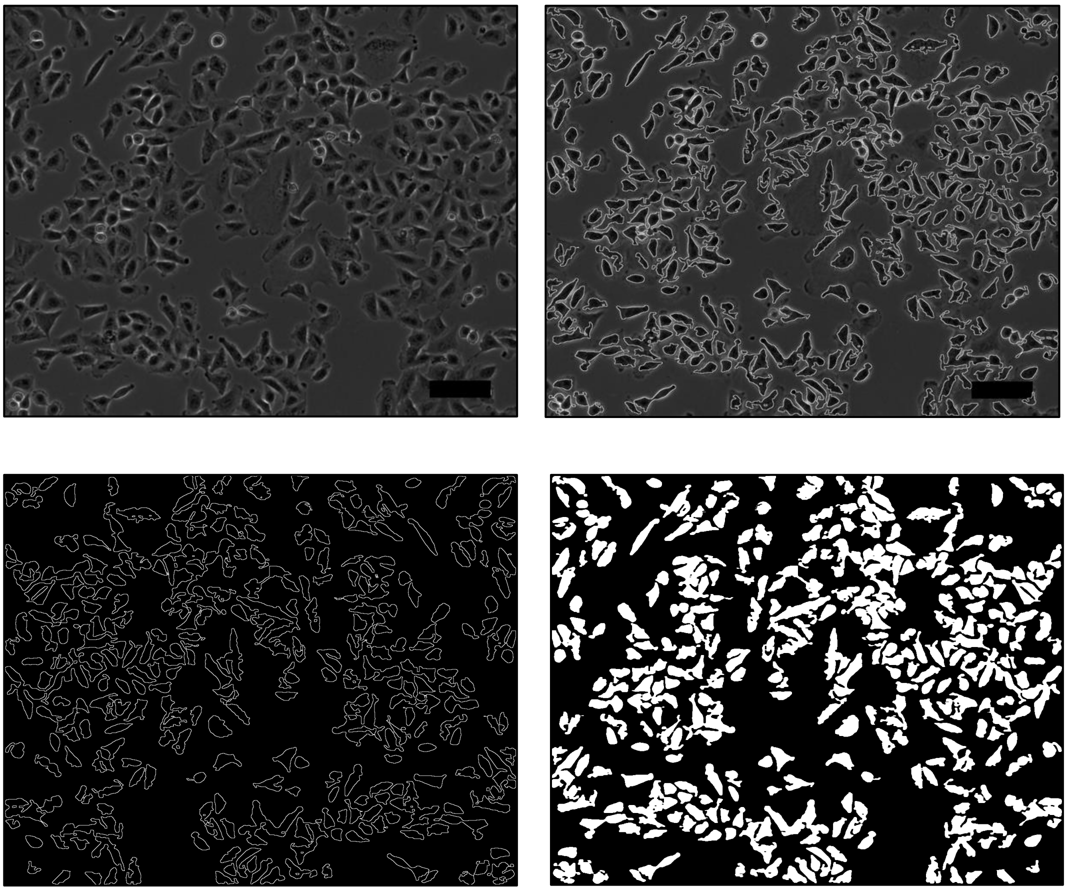


Figure 6: Threshold Annotation Technique

In response to the limitations faced with threshold labeling, a second approach was adopted, which involved manual annotation. This approach required meticulous labeling of the cellular structures, and which was executed using the platform apeer.com. Despite being a manual process, the manual annotation approach gave more favorable outcomes compared to threshold labeling. However, it's worth noting that these annotations were conducted by a master student rather than a specialist in image annotation or cellular biology. Since this process was manual, it was quite time consuming. For this reason, different preprocessing steps and augmentation were implemented.

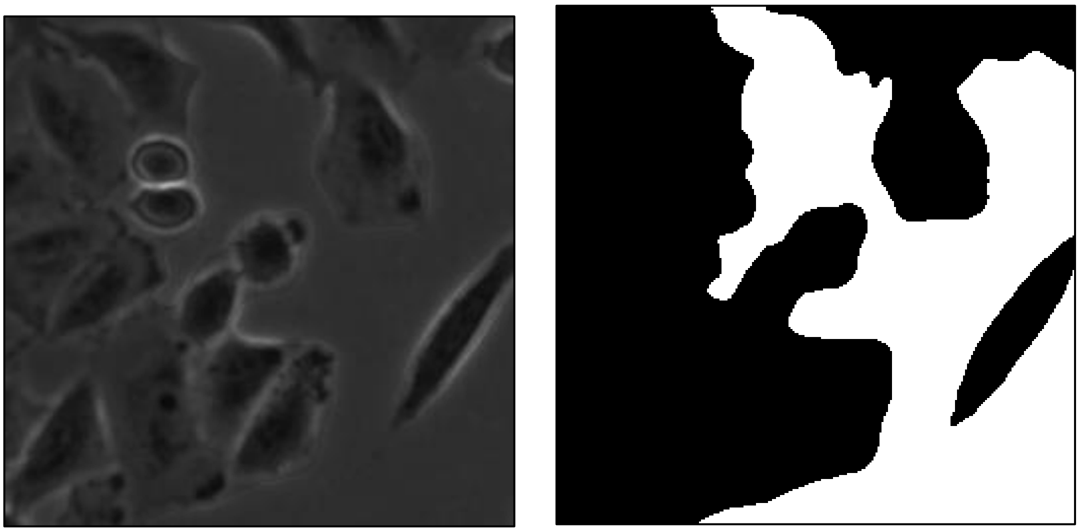


Figure 7: Manual Annotation Technique

3.3 Preprocessing

Several preprocessing steps were implemented individually to address the low contrast of the images and enhance their overall quality. Each preprocessing technique was applied separately to the raw dataset. First, histogram normalization was performed to adjust the intensity distribution, thereby improving the contrast and making the features more distinguishable.

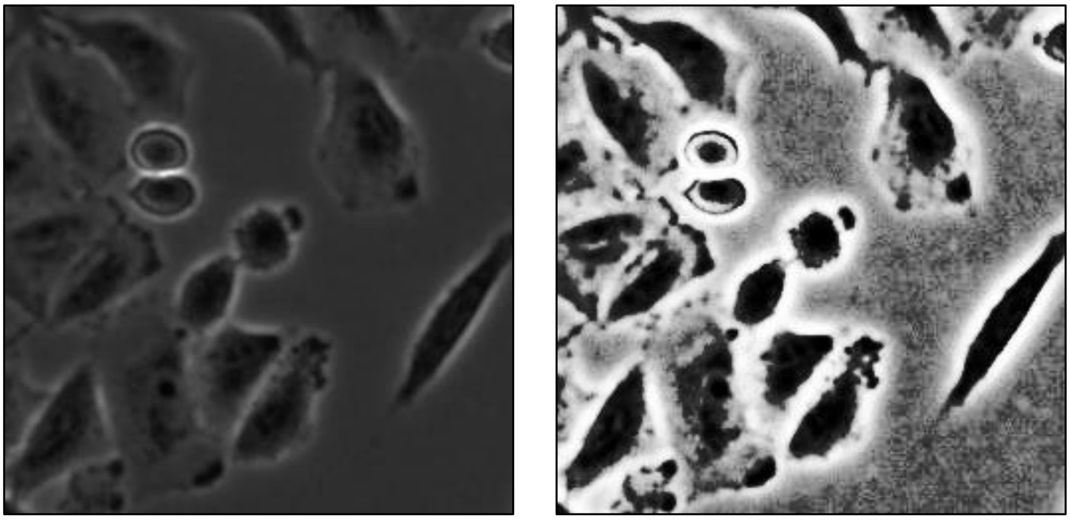
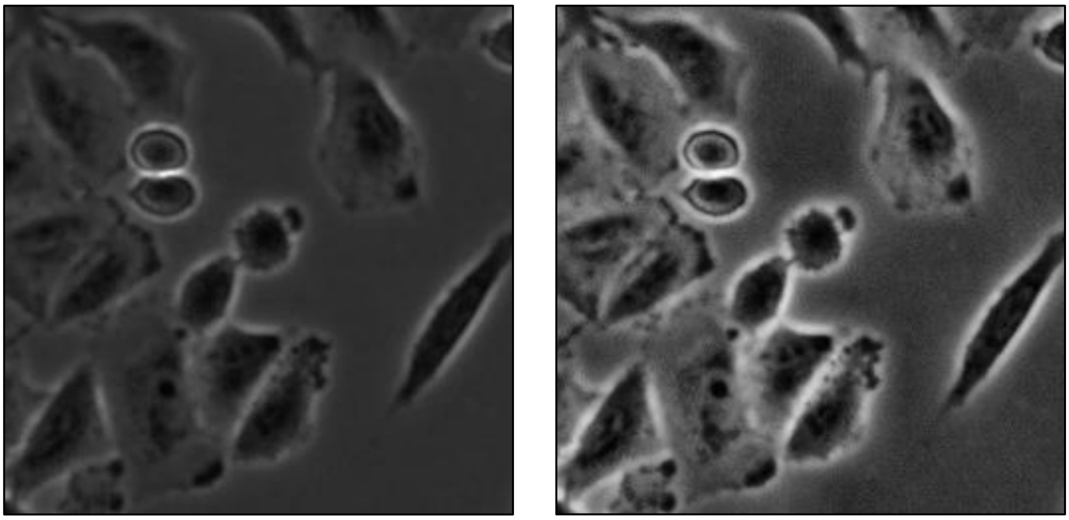


Figure 8: Original Image and Histogram Normalization

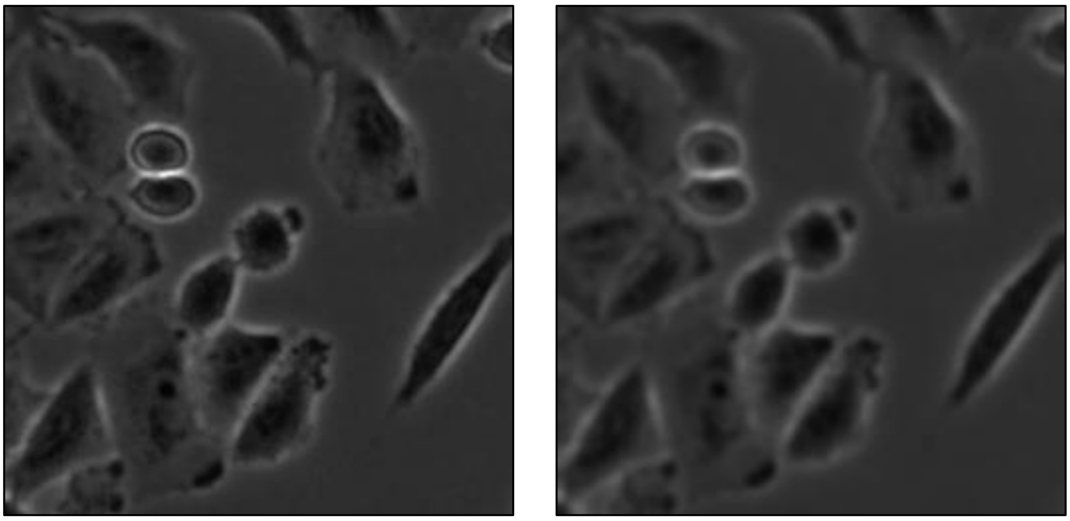
Second, Contrast Limited Adaptive Histogram Equalization (CLAHE) was applied to the raw dataset, which further enhanced the local contrast and provided better differentiation of cellular structures.



(a) Original Image ***(b) CLAHE***

Figure 9: Original Image and CLAHE

Third, a Gaussian filter was utilized on the raw dataset to reduce image noise and smooth the image, aiding in the preservation of important details.



(a) Original Image ***(b) Gaussian Filter***

Figure 10 : Original Image and Gaussian Filter

Finally, a median filter was applied to the raw dataset to effectively remove any remaining noise and enhance the edges of the cellular features.

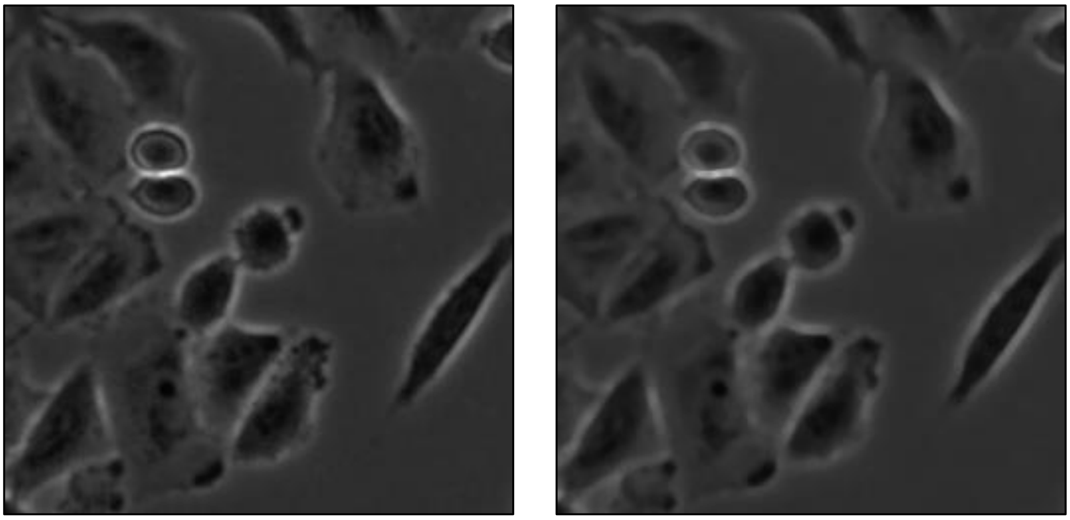


Figure 11: Original Image and Median Filter

3.3.2 Augmentation

For data augmentation, three approaches were chosen in order to increase the dataset and the segmentation model's performance. The first approach involved cropping the original images from their size of 1024x1280 pixels to smaller sections of 256x256 pixels. In order to train segmentation models more quickly and allow the model to focus on more specific features within the images, cropping is utilized to lower the computational load on the model.

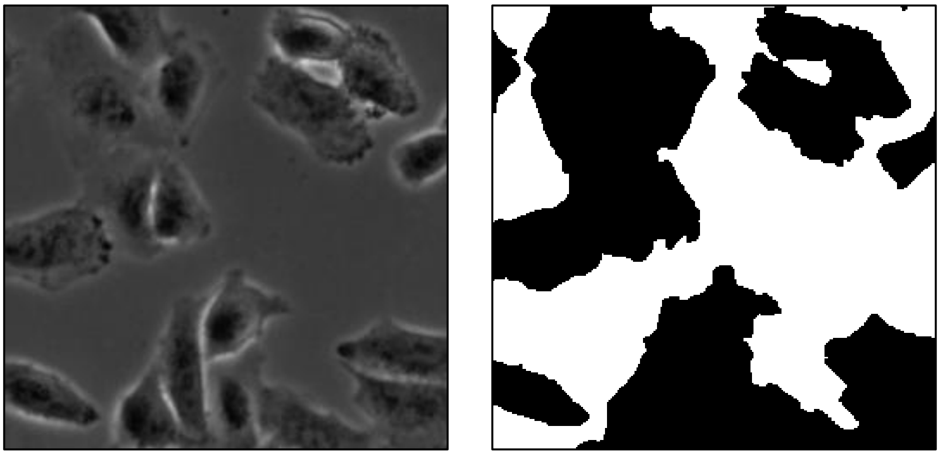


Figure 12: Cropped and Manual Annotation

The second approach used Albumemtations stretch operator to generate additional images. Applying this operator to the original images and the corresponding ground truth annotations was done consistently. Variability is added by extending the images, which is supposed to improve the model's capacity for generalization of various cellular structure sizes and forms.

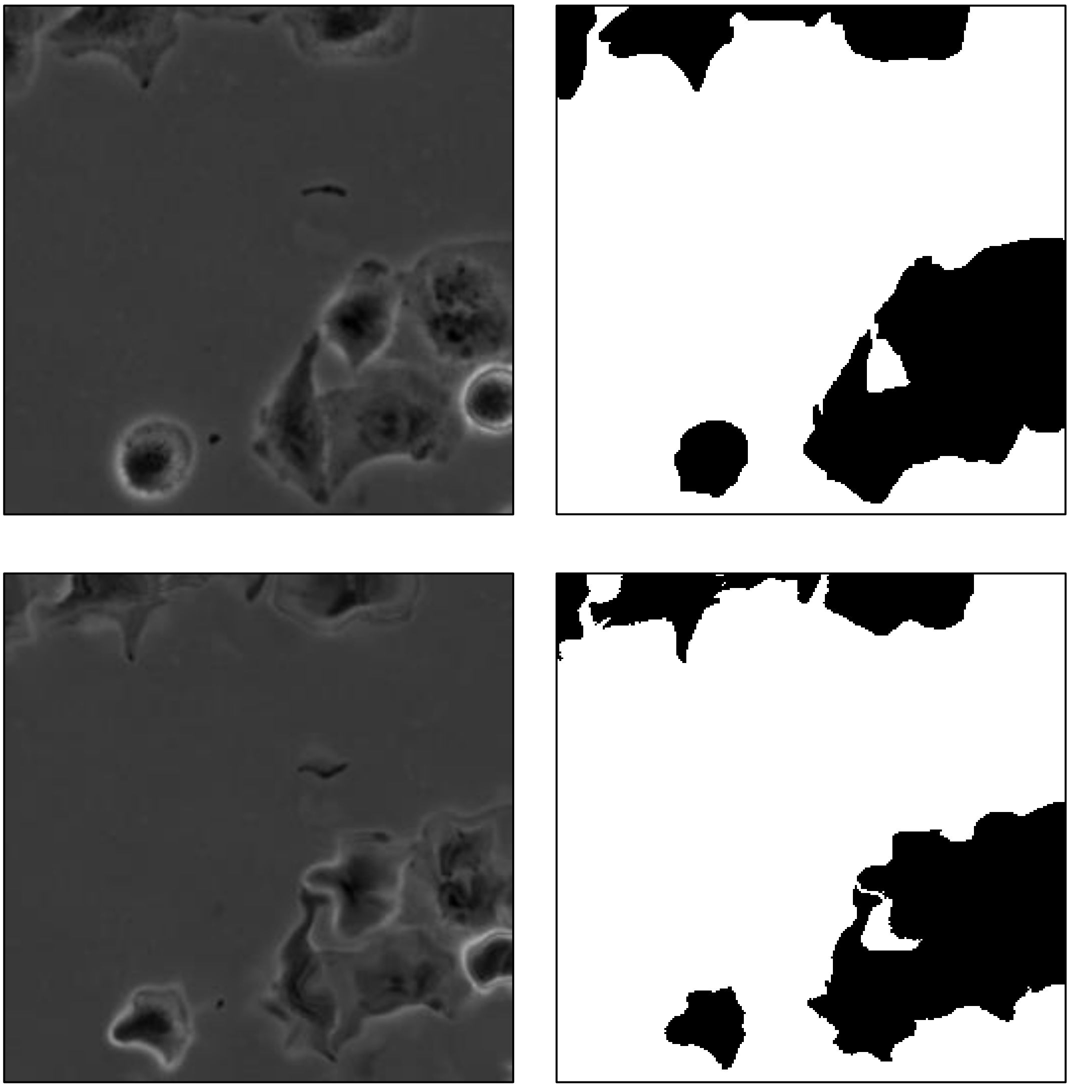


Figure 13: Stretch Operator

The third approach used the rotate and flip operator, rotating the images into three additional positions and flipping them horizontally and vertically. These transformations were applied to both the original images and their ground truths. Such operators do not add new information to the dataset (differently from stretch operator), but when dealing with a small dataset they expose the model to different orientations and perspectives of the images.

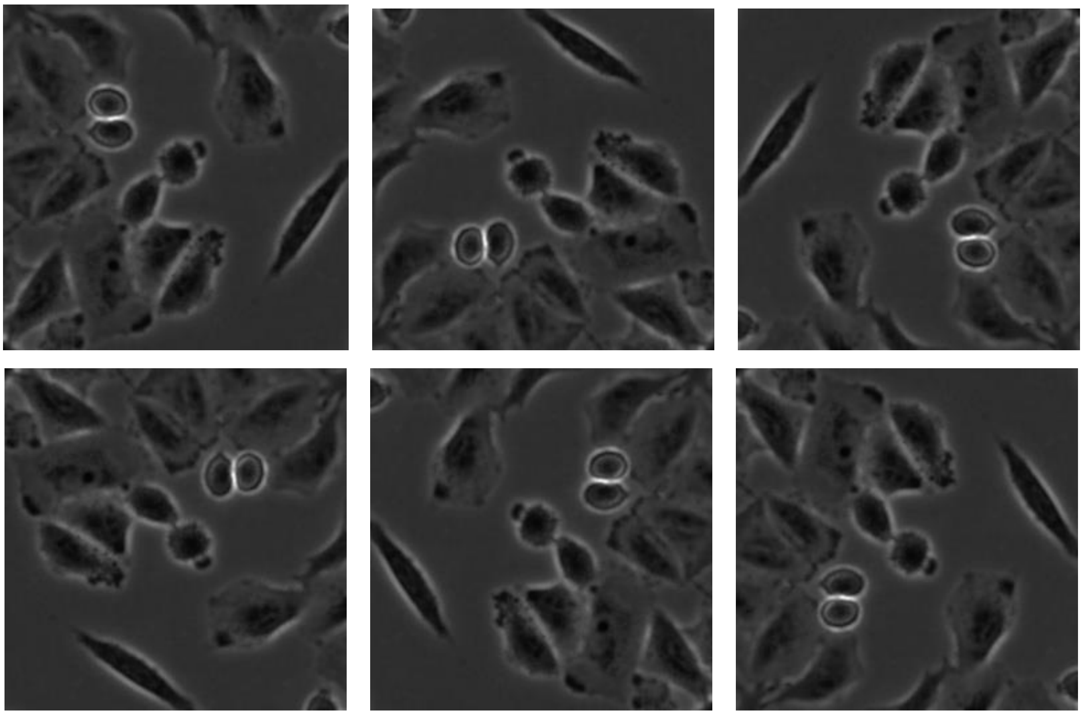


Figure 14: Flip and Rotate

3.4 Segmentation Phase

Semantic segmentation was the first phase of this study, to be performed after preprocessing, annotation, and augmentation. This task was to separate cellular features from the background. This goal was accomplished by applying a UNet model, which is well-known for doing well in semantic segmentation tasks.

A description of the best Unet architecture in this study:

Input layer: Takes input images of size 256x256 pixels with a single channel (grayscale).

Encoding path: Starts with a convolutional layer (conv2d_145) followed by batch normalization (batch_normalization_138) and dropout (dropout_69). Then, another convolutional layer (conv2d_146) followed by batch normalization (batch_normalization_139) and max pooling (max_pooling2d_31). This pattern repeats with increasing filter sizes and decreasing spatial dimensions through max pooling. This sequence creates a contracting path that extracts features at multiple levels of abstraction.

Bottleneck: After several convolutional and pooling layers, the network reaches a bottleneck layer (conv2d_155) where the spatial dimensions are greatly reduced, but the depth (number of channels) is increased.

Decoding path: Starts with transposed convolutional layers (conv2d_transpose_31, conv2d_transpose_32, etc.) which increase the spatial dimensions. Concatenation (concatenate_31, concatenate_32, etc.) is performed with the corresponding feature maps from the encoding path to provide detailed localization information. Then, convolutional layers (conv2d_157, conv2d_158, etc.) followed by batch normalization and dropout are applied to refine the segmentation.

Output layer: The final layer (conv2d_167) produces segmentation masks with the same dimensions as the input images, where each pixel represents a class probability (in this case, binary segmentation).

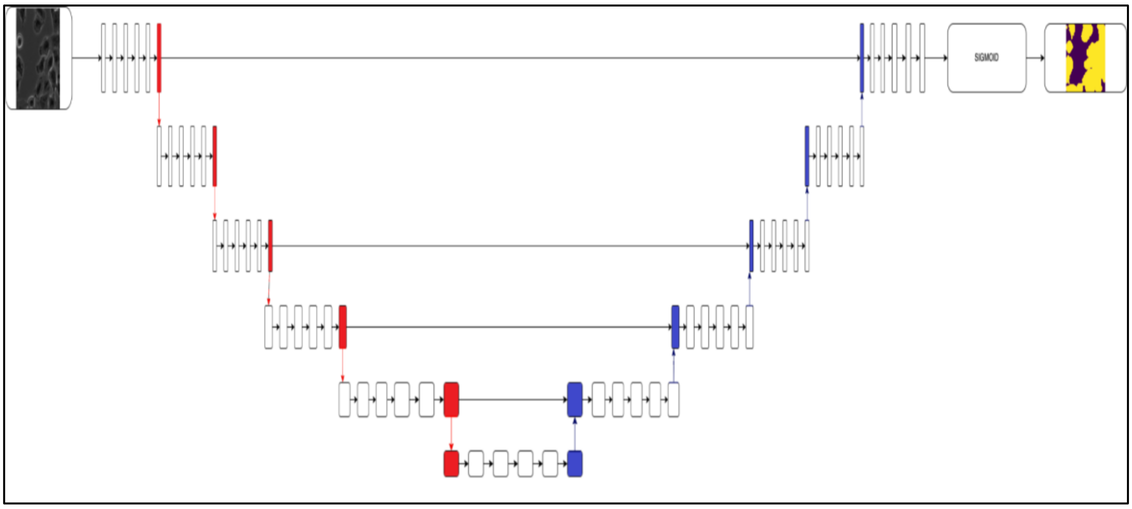


Figure 15: UNET Architecture

For the Segmentation phase, 20 different Unet Models were used. They had different training sizes, different batch sizes, different epochs sizes, and different loss functions. Not only simple Unet was used in these trials, Stacked Unet, U2Net and CellSegUnet can be mentioned.

After training the models, first the accuracy of the model was found. It was done by testing 19 images and comparing their ground truth with the predicted image. The best performing models are used for further work to find the confluency of the cells in the image.

3.5 Classification Phase

The second phase of this study is the classification task. Different combinations were used with unsupervised learning KMeans, to cluster the images.

The first try was to use KMeans to cluster two different cells, A549 and Balb 3T3. For these two cells, the dataset was acquired from D1 and the original image was cropped. Two normalization techniques were used: Min-Max and Z-score.

The second try was to use KMeans to cluster between the 3 densities of biomaterials used in A549. Since the biomaterial was injected on the second day, D2 was also included into the classification, which means 4 classes. PCA was used and different trials were conducted with VGG16 and ResNet50.

The same approach was implemented, but this time with supervised learning LeNet.

The third try was to use KMeans to cluster between the 2 densities of biomaterials used in A549. The densities of the biomaterial taken into consideration for the classification were 20ug and 50ug. PCA was used and different trials were conducted with VGG16 and ResNet50. Again, the same approach was used with supervised learning LeNet. The first image is the raw image, the second is the manual annotation performed on it, and the last image is the mask. For a short training test the images with the best masks will be cropped in a new format: 255x255. This will generate twenty images out of one.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Segmentation Phase

28 UNet models and architectures were trained and tested over different hyperparameters. The table details the performance of 28 different UNet models trained and tested with varying hyperparameters to determine their effectiveness in cell confluence determination. Each model is evaluated based on its architecture, loss function, number of images, batch size, epochs, and accuracy metrics including train, validation, and test accuracy. Among these, models number 23 and 28 have been highlighted as the best-performing models. The best two performing models, highlighted below, are used to find the confluence of the cells.

Table 2: UNET Model Results

Model Number	**UNET Models**	**Loss Function**	**Nr of Images**	**Batch Size**	**Epochs**	**Train Accuracy**	**Validation Accuracy**	**Test Accuracy**
1	unet2	binary_crossentropy	19	2	100			48.03%
2	unet2	binary_crossentropy	19	8	100			44.42%
3	unet2	binary_crossentropy	19	16	100			
4	unet2	binary_crossentropy	38	2	100			95.83%
5	unet2	binary_crossentropy	38	8	100			96.06%
6	unet2	binary_crossentropy	38	16	100			
7	unet2	binary_crossentropy	76	2	100	0.944	0.955	94.22%
8	unet2	binary_crossentropy	76	8	100	0.890	0.647	75.63%
9	unet2	binary_crossentropy	76	16	100	0.899	0.655	68.47%
10	SUNET	binary_crossentropy	76	2	100	0.829	0.784	94.79%
11	unet2	binary_crossentropy	285	2	100	0.832	0.792	95.27%
12	unet2	binary_crossentropy	285	8	100	0.982	0.969	95.63%
13	unet2	binary_crossentropy	285	16	100	0.860	0.601	78.45%
14	unet2	binary_crossentropy	285	24	100	0.932	0.567	86.21%
15	U2NET	binary_crossentropy	285	2	100			95.22%
16	SUNET2	binary_crossentropy	285	2	100			95.96%
17	SUNET2	binary_crossentropy	285	8	100	0.977	0.967	95.22%
18	CellSegUne	binary_crossentropy	285	2	100	0.014	0.0000158	
19	unet2	binary_crossentropy	1710	2	100	0.906	0.739	
20	unet2	binary_crossentropy	1710	8	100	0.849	0.498	
21	unet2	binary_crossentropy	1710	16	100	0.893	0.769	
22	unet3	combined_dice_bce_loss	285	2	100	0.990	0.967	
23	***unet3***	***combined_dice_bce_loss***	***285***	***8***	***100***	***0.996***	***0.970***	
24	unet3	combined_dice_bce_loss	285-gauss	8	100	0.865	0.532	
25	unet3	combined_dice_bce_loss	285-median	8	100	0.874	0.544	

Model Number	UNET Models	Loss Function	Nr of Images	Batch Size	Epochs	Train Accuracy	Validation Accuracy	Test Accuracy
26	unet3	combined_dice_bce_loss	285-CLAHE	8	100	0.903	0.462	
27	unet3	combined_dice_bce_loss	285-histnorm	8	100	0.940	0.553	
28	***unet3***	***combined_dice_bce_loss***	***285***	***8***	***200***	***0.993***	***0.966***	

The simple UNet (unet2) models are trained with a binary cross-entropy loss function and cover a range of datasets from 19 to 1710 images. These models exhibit significant variability in performance, with train accuracy ranging from 82.9% to 98.2%, validation accuracy from 56.7% to 96.9%, and test accuracy from 44.42% to 95.63%. The results indicate that while some configurations of the simple UNet can achieve high accuracy, the performance is highly sensitive to changes in batch size and the number of images used.

The stacked UNet architectures (SUNET and SUNET2) show a more consistent performance across different configurations. These models, also using a binary cross-entropy loss function, were trained on datasets ranging from 76 to 285 images with batch sizes between 2 and 8. They demonstrate high train accuracy, from 82.9% to 97.7%, and validation accuracy between 78.4% and 96.7%, with test accuracy consistently above 94%.

In contrast, the CellSegUnet model, which is designed specifically for cell segmentation, performed poorly in this experiment. Despite being trained on 285 images with a batch size of 2, it achieved an exceptionally low train accuracy of 1.4% and a validation accuracy of just 0.001508%, indicating potential issues with either the implementation or suitability of this model for the specific task.

The U2NET model, also trained on 285 images with a binary cross-entropy loss function, showed promising test accuracy of 95.22%. However, the lack of detailed train and validation accuracy metrics limits a comprehensive evaluation of its performance. Nonetheless, the test accuracy suggests that U2NET can be effective for segmentation tasks with a smaller batch size.

The UNet3 models, which utilize a combined Dice and Binary Cross-Entropy loss function, stand out for their exceptional performance. These models were trained on datasets of 285 images, with batch sizes between 2 and 8, and epochs ranging from 100 to 200. They consistently exhibit high train accuracy (86.5% to 99.6%) and validation accuracy (46.2% to 97.0%). The variability in validation accuracy based on preprocessing techniques (Gaussian, Median, CLAHE, and Histogram Normalization) highlights the model's sensitivity to input data quality. Despite this, the superior performance of models 23 and 28, with train accuracy of 99.6% and 99.3% and validation accuracy of 97.0% and 96.6% respectively, underscores the effectiveness of the combined loss function in enhancing segmentation accuracy.

Below, 20 unseen images were used to test the two best models. It uses several evaluation metrics such as Accuracy, Intersection over Union (IoU), Dice Coefficient, and F1-Score. After being tested on several images, the mean of each evaluation metrics is calculated, for a conclusion to be drawn regarding each model. This is done for both Model_1 (model number 23) and Model_2 (model number 28).

4.1.1 UNet Model_1

Table 3: Model 1 Results

		Model_1		
Image	***Accuracy***	***IoU***	***Dice Coefficient***	***F1-Score***
image_0_0.png	0.96	0.92	0.96	0.96
image_0_1.png	0.97	0.95	0.97	0.97
image_0_2.png	0.96	0.91	0.96	0.96
image_0_3.png	0.96	0.92	0.96	0.96
image_0_4.png	0.98	0.97	0.99	0.99
image_1_0.png	0.96	0.94	0.97	0.97
image_1_1.png	0.95	0.89	0.94	0.94
image_1_2.png	0.94	0.90	0.95	0.95
image_1_3.png	0.97	0.95	0.98	0.98

image_1_4.png	0.96	0.95	0.97	0.97
image_2_0.png	0.96	0.94	0.97	0.97
image_2_1.png	0.93	0.81	0.90	0.90
image_2_2.png	0.95	0.86	0.92	0.92
image_2_3.png	0.96	0.91	0.96	0.96
image_2_4.png	0.98	0.97	0.99	0.99
image_3_0.png	0.98	0.97	0.98	0.98
image_3_1.png	0.97	0.94	0.97	0.97
image_3_2.png	0.94	0.86	0.92	0.92
image_3_3.png	0.96	0.95	0.97	0.97
image_3_4.png	0.92	0.87	0.93	0.93

Model_1 demonstrates high consistency and robustness in performance across all unseen images. The Accuracy values range from 0.92 to 0.98, indicating that the model is highly reliable in predicting cell confluence accurately. The IoU values, which range from 0.81 to 0.97, show that the model achieves a high degree of overlap between predicted and actual segmented areas. This is further supported by the Dice Coefficient and F1-Score, both ranging from 0.90 to 0.99, which confirm the model's effectiveness in achieving precise and accurate segmentation.

Table 4: Model 1 Mean Results

Model_1: Mean Results	
Model Accuracy	0.96
Model IoU	0.92
Model Dice Coefficient	0.96
Model F1-Score	0.96

The consistently high performance of Model_1 underscores its capability to generalize well to new, unseen images. The few instances of slightly lower performance (e.g., image_2_1 and image_3_4) still give good results, suggesting minor variability that could be attributed to differences in image quality or illumination problems.

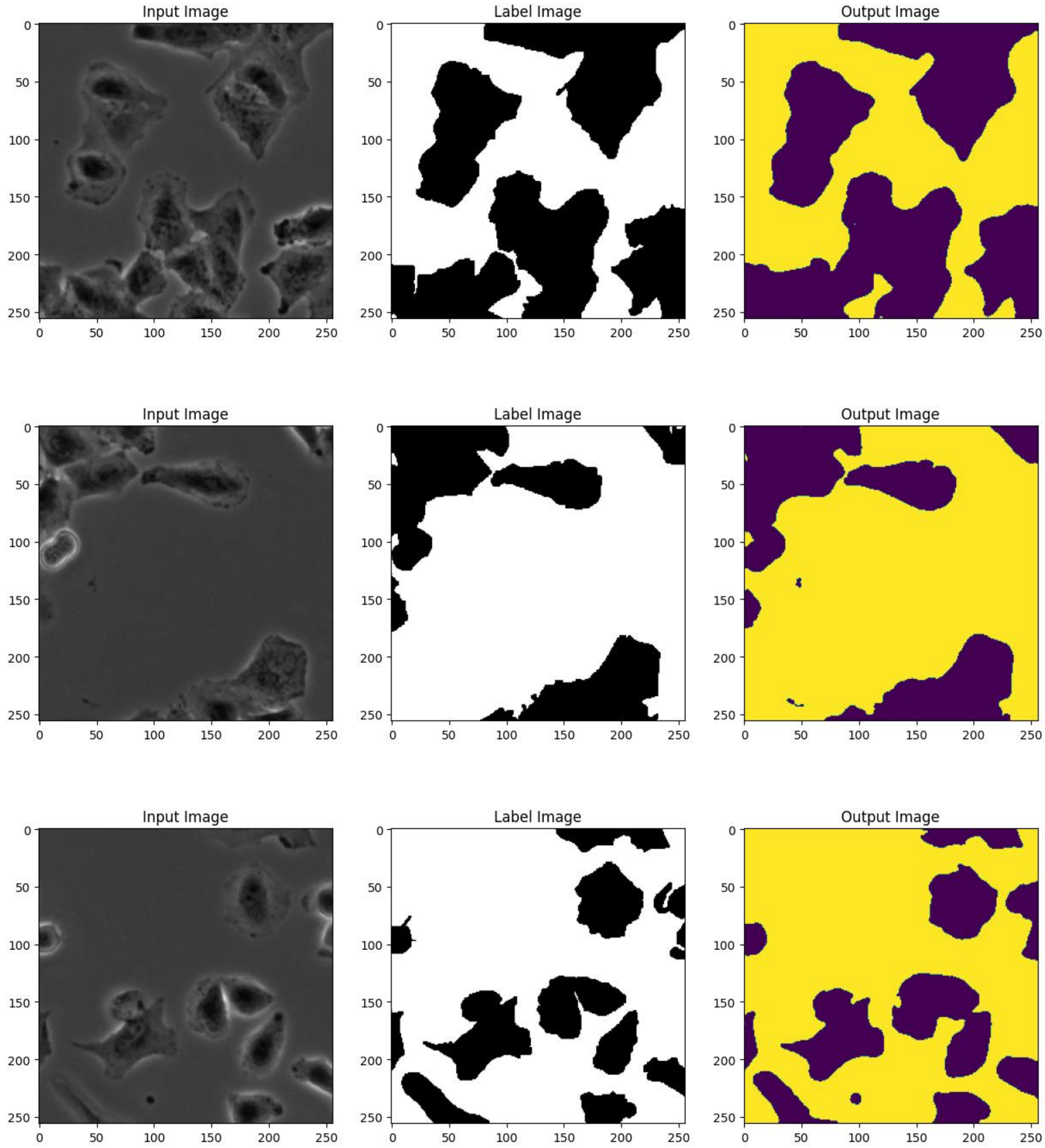


Figure 16: Predictions using Model_1

The pipeline for predicting the label start by getting a big image 1280x1024. The next step is cropping it to 256x256 since that is the size of image the model is trained on. Some predicting images are also shown above. In the end the inputted image is reconstructed. Below, four images will be illustrated. Image (a) is the original image in the original size; image (b) is the manually labeled mask of the original image; image (c) is the predicted mask of the original image; and image (d) is a visual XOR of the manually annotated mask and predicted mask.

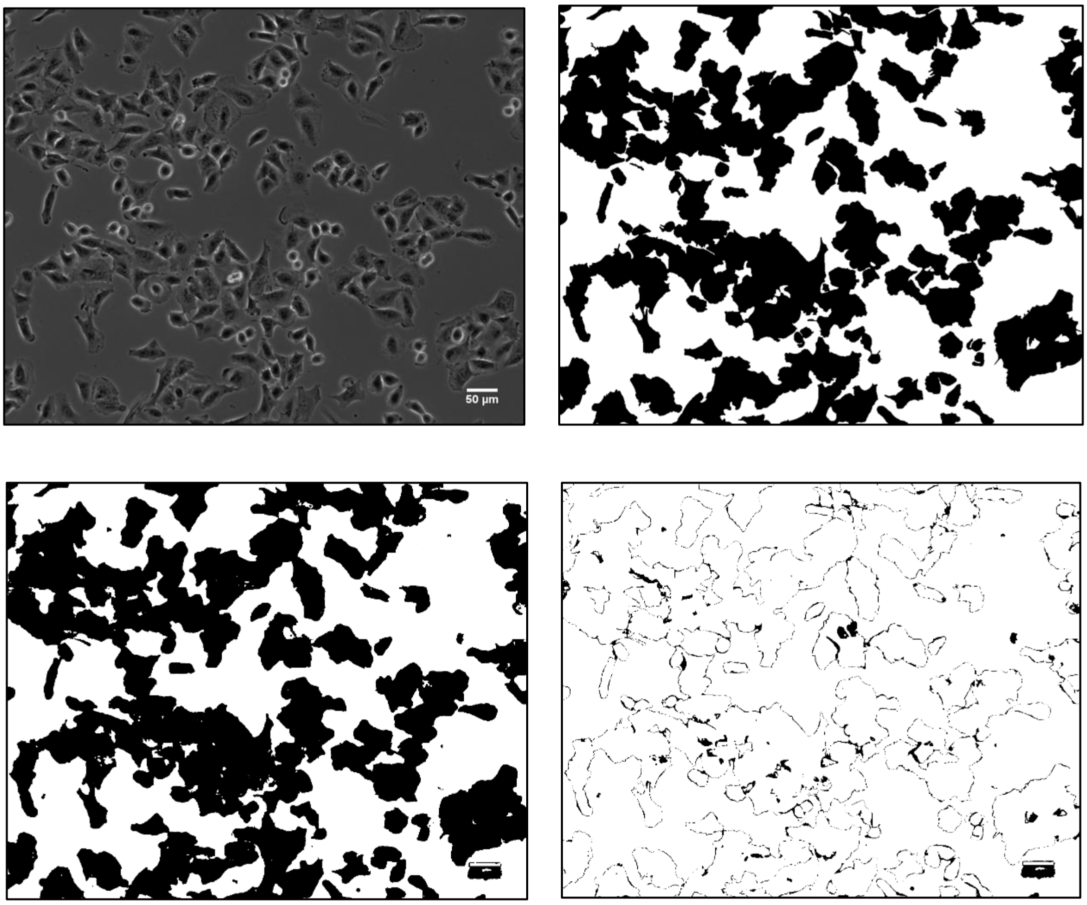


Figure 17: Predictions using Model_1

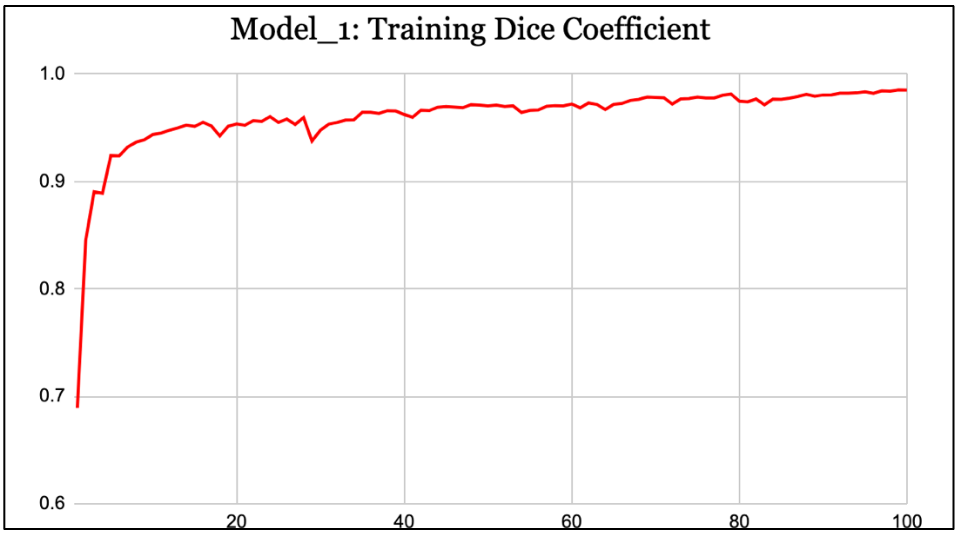


Figure 18: Training Dice Coefficient for Model_1

4.1.2 UNet Model_2

Table 5: Model 2 Evaluation Metrics

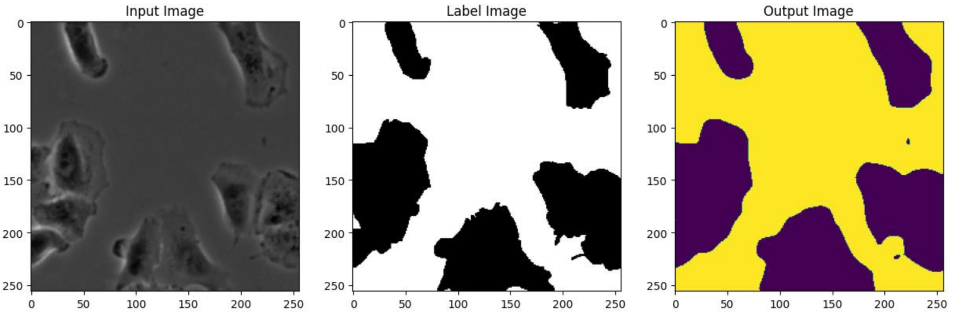
		Model_2		
Image	*Accuracy*	*IoU*	*Dice Coefficient*	*F1-Score*
image_0_0.png	0.95	0.91	0.95	0.95
image_0_1.png	0.97	0.93	0.96	0.96
image_0_2.png	0.95	0.89	0.94	0.94
image_0_3.png	0.95	0.92	0.96	0.96
image_0_4.png	0.97	0.96	0.98	0.98
image_1_0.png	0.95	0.92	0.96	0.96
image_1_1.png	0.95	0.88	0.94	0.94
image_1_2.png	0.94	0.90	0.95	0.95
image_1_3.png	0.97	0.95	0.98	0.98
image_1_4.png	0.97	0.95	0.98	0.98
image_2_0.png	0.96	0.93	0.97	0.97
image_2_1.png	0.94	0.82	0.90	0.90
image_2_2.png	0.95	0.84	0.91	0.91
image_2_3.png	0.95	0.90	0.95	0.95
image_2_4.png	0.98	0.97	0.98	0.98
image_3_0.png	0.97	0.95	0.98	0.98

image_3_1.png	0.95	0.92	0.96	0.96
image_3_2.png	0.93	0.85	0.92	0.92
image_3_3.png	0.96	0.95	0.97	0.97
image_3_4.png	0.92	0.85	0.92	0.92

Model_2 showcases high consistency and robustness in its segmentation performance on unseen images. The model maintains a mean Accuracy of 0.95. The mean IoU of 0.91 reflects a high degree of overlap between the predicted and actual segmented areas, while it achieves a mean Dice Coefficient and F1-Score of 0.95. Images such as image_2_1 and image_3_4 show slightly lower performance with IoU values of 0.82 and 0.85, and Dice Coefficients of 0.90 and 0.92, respectively. These instances, although still within acceptable ranges, highlight potential areas for further refinement or indicate differences in image complexity or quality.

Table 6: Model 2 Mean Results

Model_2: Mean Results	
Model Accuracy	0.95
Model IoU	0.91
Model Dice Coefficient	0.95
Model F1-Score	0.95



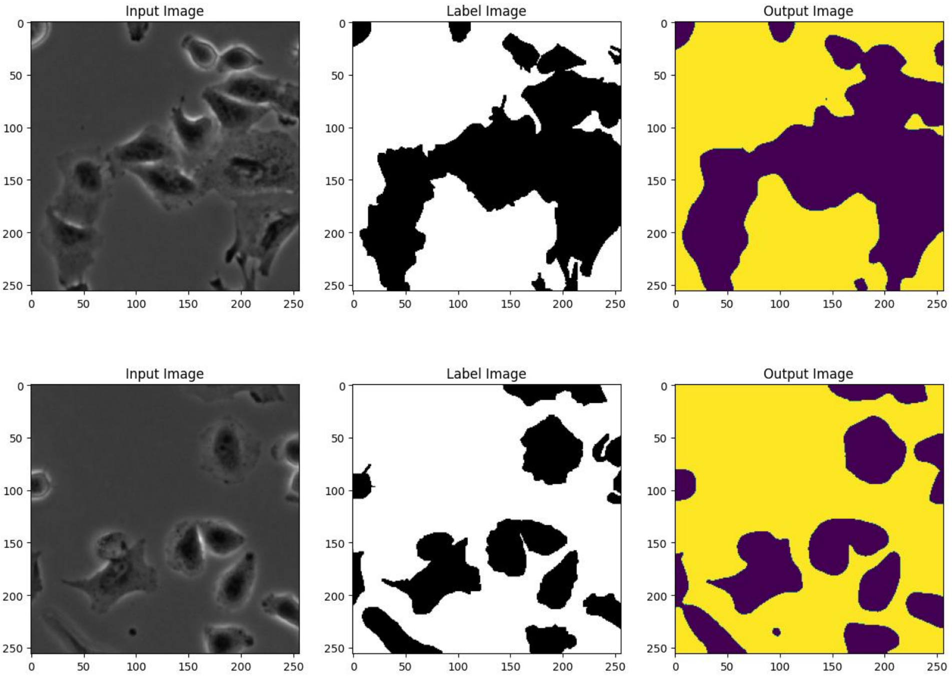
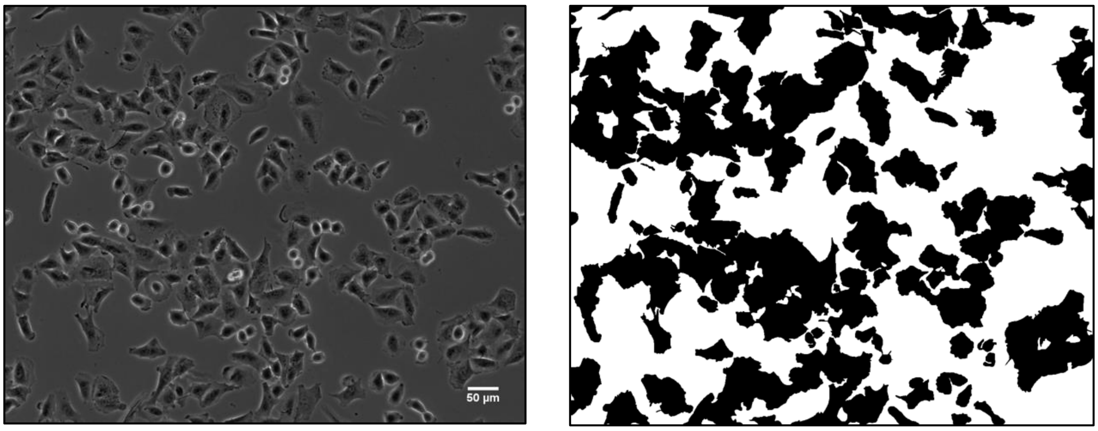


Figure 19: Predictions using Model_2

The pipeline for predicting the label start by getting a big image 1280x1024. The next step is cropping it to 256x256 since that is the size of image the model is trained on. Some predicting images are also shown above. In the end the inputted image is reconstructed. Below, four images will be illustrated. Image (a) is the original image in the original size; image (b) is the manually labeled mask of the original image; image (c) is the predicted mask of the original image; and image (d) is a visual XOR of the manually annotated mask and predicted mask.



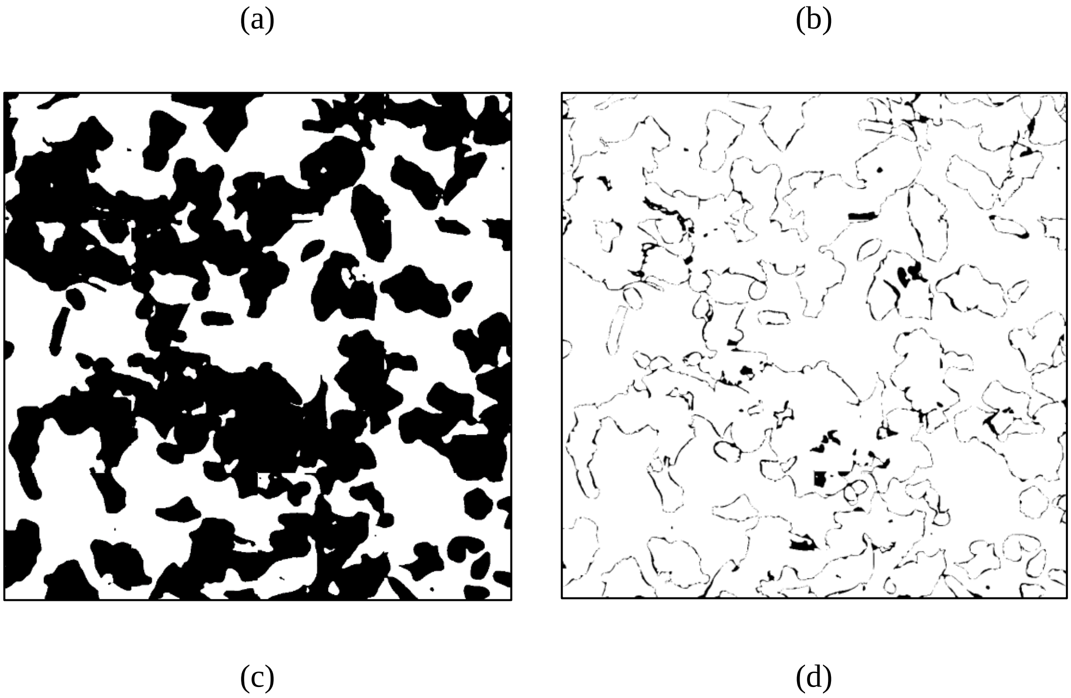


Figure 20: Predictions using Model_2

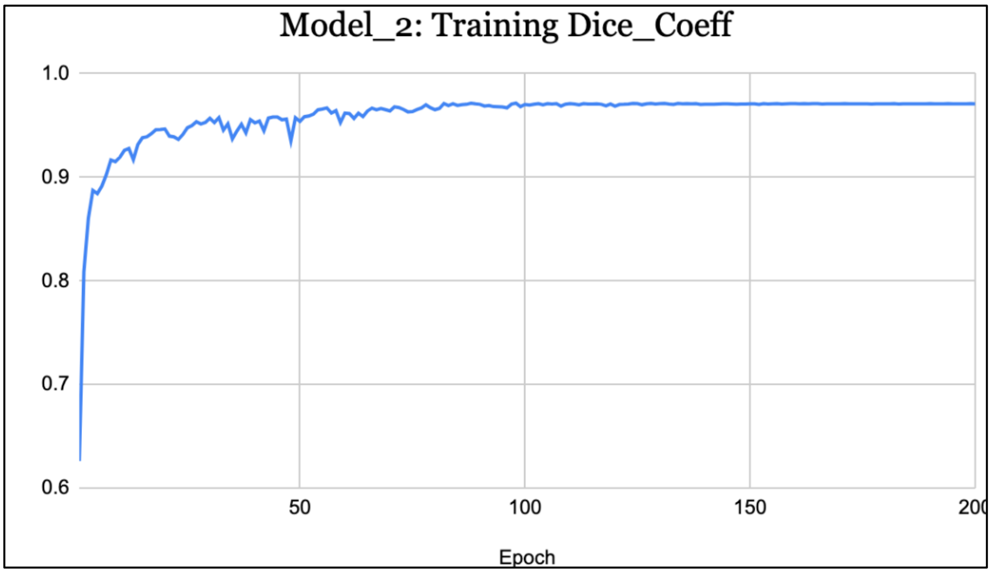


Figure 21: Training Dice Coefficient for Model_2

In conclusion, both Model_1 and Model_2 are highly effective for segmentation tasks, achieving high accuracy and precision on unseen images. Model_1 has a slight overall advantage in all metrics—Accuracy, IoU, Dice Coefficient, and F1-Score. This indicates that Model_1 may be marginally more reliable and precise in its predictions, particularly for complex segmentation scenarios. Nonetheless, the differences are minimal, and both models are strong candidates for practical applications in cell segmentation tasks.

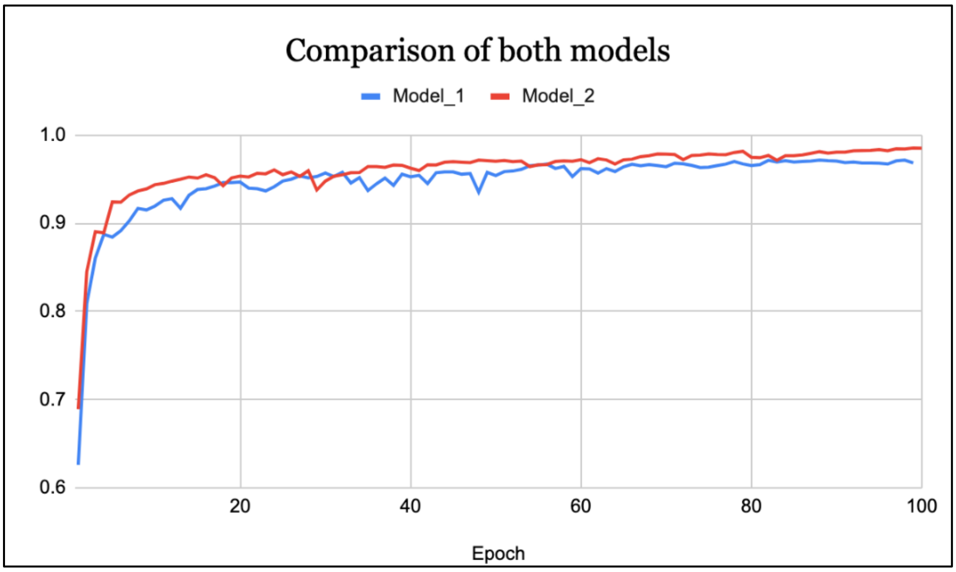


Figure 22: Training Dice Coefficient for both models

4.1.3 Confluency Results

Cell confluency was calculated using two selected models from a set of 28, focusing on their performance across three distinct datasets. The datasets comprised two groups containing biomaterial PAR30 and one group with biomaterial PLL250. Each dataset was further subdivided into two main folders: day 1 images and day 2 images. For the biomaterial PAR30, images were categorized by varying densities of 5 µg, 20 µg, and 50 µg. In the case of biomaterial PLL250, the categories included densities of 5 µg and 500 µg. The cell confluency was determined using the predicting models, which employed the U-Net architecture. Subsequently, the average confluency for each folder was calculated. This comprehensive analysis helped with the assessment of the risk associated with different biomaterial densities.

4.1.3.1 Dataset_1: PAR30

The images in this dataset contain several folders: D1- Day 1 of the measurements; D2-Day 2 which means monitoring the cells in the second day; D2_PAR30_5ug – cells in day 2 but the biomaterial with density 5ug is injected; D2_PAR30_20ug – cells in day 2 but the biomaterial with density 20ug is injected; D2_PAR30_50ug – cells in day 2 but the biomaterial with density 50ug is injected.

For each of the above folders, the images were tested using both UNet Model_1 and Model_2.

After calculating the cell confluency for each folder using both models, the average cell confluency is found. Also, the standard deviation for each folder is found. The averages are compared between the models, but also between the folders to examine the trend.

In the end, the averages are normalized and the assessment of biomaterial is conducted. The threshold used is 70%. If the area covered by the cells or differently mentioned as cell confluency, is greater than 70% it means that the biomaterial is nontoxic. Consequently, if the confluency is below that threshold, the biomaterial is considered toxic.

Table 7: D1 Confluency of both models

D1			
Cell_Name	***C_Unet100***	***C_Unet200***	***Difference***
001	62.461	60.135	2.325
002	52.321	54.553	2.232
003	58.223	59.186	0.963
004	40.793	42.509	1.715
005	47.274	46.481	0.793
006	43.166	44.031	0.865
007	59.253	60.118	0.865
008	55.525	56.870	1.346
009	41.405	42.930	1.526
010	32.332	31.996	0.336
011	63.910	64.756	0.846
012	55.107	56.204	1.097
013	39.883	40.194	0.311
014	45.931	46.753	0.821
015	38.342	34.700	3.642
016	43.032	44.608	1.576
017	49.817	51.922	2.105
018	44.842	44.844	0.002
019	33.534	34.355	0.821
020	41.666	42.162	0.496
021	39.107	38.260	0.847
022	32.352	31.824	0.528
023	51.830	54.300	2.470
024	42.394	42.501	0.108
025	51.466	51.253	0.213
026	38.231	38.691	0.460
027	48.996	48.640	0.356
028	45.152	45.786	0.634
029	45.889	46.463	0.574
030	33.704	31.411	2.293
Average	45.931	46.281	
Standard Deviation	8.713500185	9.160313082	

Table 8: D2 Confluency of both models

D2			
Cell_Name	***C_Unet100***	***C_Unet200***	***Difference***
001	93.988	97.058	3.070
002	82.002	84.270	2.268
003	96.393	98.016	1.623
004	96.707	98.768	2.061
005	39.230	44.861	5.631
006	73.691	78.866	5.175
007	86.926	86.730	0.195
008	87.566	91.375	3.809
009	95.629	94.929	0.700
010	93.060	92.528	0.532
011	92.860	93.568	0.708
012	78.490	84.409	5.919
Average	84.712	87.115	
Standard Deviation	16.1549907	14.68651317	

Table 9: D2_5ug Confluency of both models

D2_5ug			
Cell_Name	***C_Unet100***	***C_Unet200***	***Difference***
001	61.739	59.340	2.399
002	81.377	85.400	4.023
003	84.250	82.893	1.357
004	92.064	91.836	0.228
005	88.897	90.740	1.842
006	85.503	85.798	0.295
007	89.151	86.578	2.573
008	86.889	91.191	4.302
009	40.120	46.001	5.881
010	91.053	91.769	0.716
011	83.251	82.254	0.998
012	92.245	98.364	6.119
Average	81.378	82.680	
Standard Deviation	15.34412322	14.99223137	

Table 10: D2_20ug Confluency of both models

D2_20ug			
Cell_Name	***C_Unet100***	***C_Unet200***	***Difference***
001	65.960	63.309	2.651
002	55.895	59.127	3.232
003	39.859	42.418	2.559
004	56.136	57.136	1.001
005	50.870	43.967	6.903
006	64.534	61.906	2.628
007	67.733	71.834	4.101
008	52.220	55.804	3.583
009	53.256	52.364	0.892
010	52.393	51.124	1.269
011	47.545	49.698	2.153
Average	55.127	55.335	
Standard Deviation	8.333729175	8.667446148	

Table 11:D2_50ug Confluency of both models

D2_50ug			
Cell_Name	***C_Unet100***	***C_Unet200***	***Difference***
001	21.633	22.705	1.072
002	32.398	30.303	2.096
003	40.192	38.314	1.878
004	27.397	27.175	0.222
005	31.501	30.923	0.578
006	34.076	32.791	1.285
007	40.402	43.070	2.668
008	35.549	26.727	8.822
009	21.048	20.663	0.385
010	32.655	22.899	9.755
011	33.498	32.364	1.134
Average	31.850	29.812	
Standard Deviation	6.372221468	6.807383643	

Table 12: Average and STDev (a) model_1 and (b) model_2

			Model_1		
Day	***D1***	***D2***	***D2_5***	***D2_20***	***D2_50***
Average	45.931	84.712	81.378	55.127	31.850
STD	8.714	16.155	15.344	8.334	6.372

(a)

			Model_2		
Day	***D1***	***D2***	***D2_5***	***D2_20***	***D2_50***
Average	46.281	87.115	82.680	55.335	29.812
STD	9.160	14.687	14.992	8.667	6.807

(b)

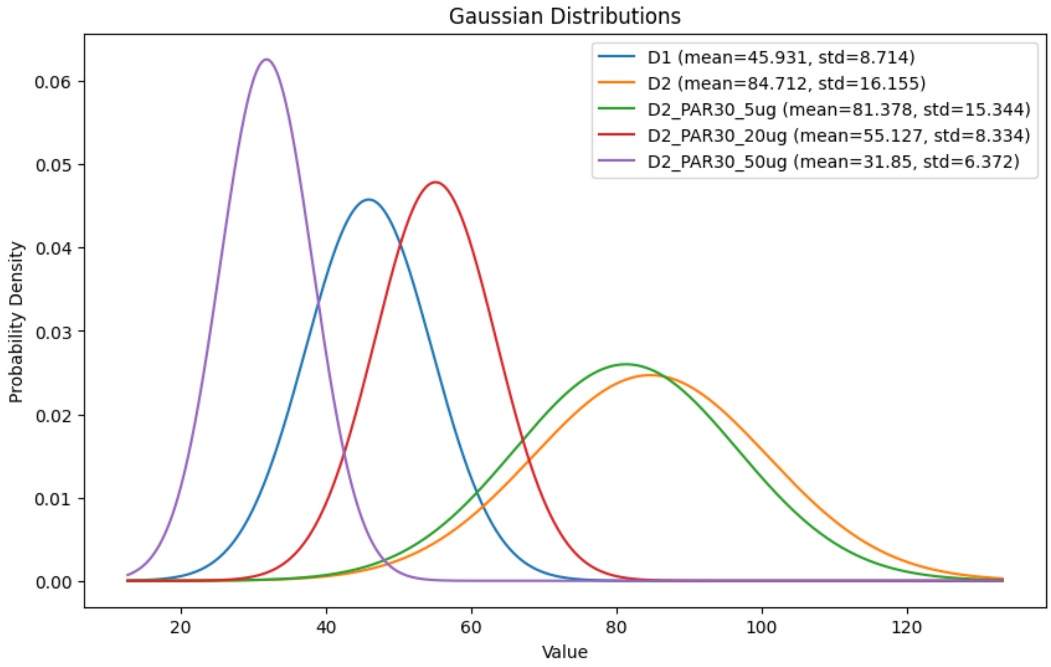


Figure 23: Gaussian Distribution for Dataset_1

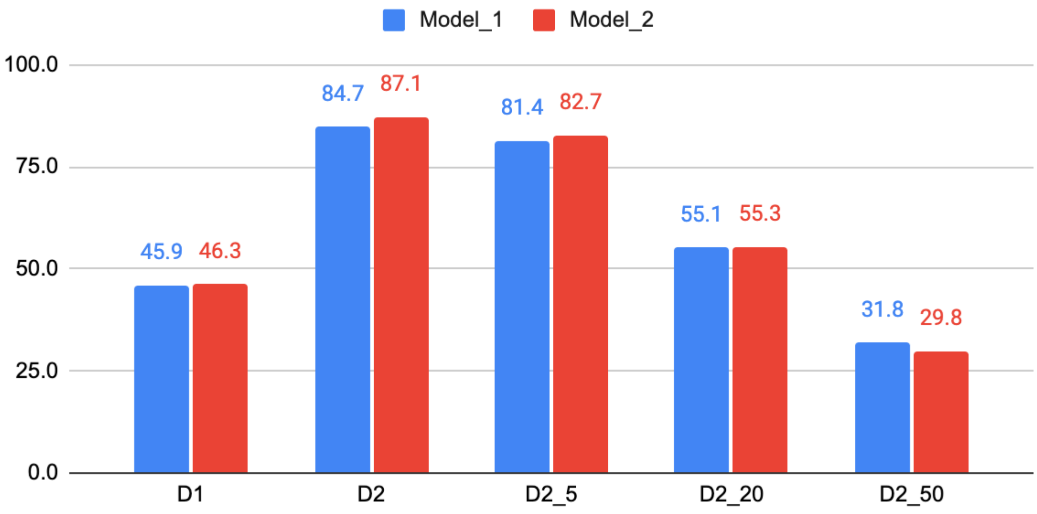


Figure 24: Comparison of Cell Confluency between models

In the above graph, the comparison of the average confluency for each day is shown. When comparing the models, they are quite similar. The models generally agree on the confluency predictions with small differences, usually around 2-3%. This suggests that both models have similar capabilities in predicting cell confluency.

The confluency of Day 1 serves as a baseline for comparison on Day 2 without biomaterial. For Day 2, a significant increase can be seen, indicating considerable cell growth over time without the presence of additional biomaterial. For biomaterial with density 5ug, the results suggest that the presence of it might slightly decrease cell confluency, though the impact is relatively minor. There is a drop of 30% from Day 2 to density 20ug which indicates more significant inhibitory effect on cell confluency. For density 50ug, the lowest confluency is observed. This indicates a strong negative impact of the high concentration of biomaterial on cell confluency. One observation is that higher biomaterial concentrations may severely limit cell health or cause significant cell death.

The presented tables illustrate the results of cell confluency measurements from segmentation, which are used to determine the toxicity of varying densities of biomaterial PAR30 over Day 2 (D2). The threshold for toxicity is defined as 70% confluency; values above this threshold indicate nontoxicity, while values below signify toxicity.

Table 13: Ratios of biomaterials over Day2

Ratio	***Model_1***	
5/D2	0.961	Nontoxic
20/D2	0.651	Toxic
50/D2	0.376	Toxic

(a)

Ratio	***Model_2***	
5/D2	0.949	Nontoxic
20/D2	0.635	Toxic
50/D2	0.342	Toxic

(b)

In Table (a), the confluency results for Model_1 are examined. At a density of 5ug of PAR30, the confluency is measured at 0.961. This high confluency value indicates that the condition is nontoxic, as it is well above the 70% threshold. However, as the density increases to 20ug, the confluency drops to 0.651. This lower value falls below the toxicity threshold, indicating that the condition has become toxic. The trend continues at a density of 50ug, where the confluency further decreases to 0.376, reinforcing the indication of toxicity.

Similarly, in Table (b), the results for Model_2 show a comparable pattern. At a density of 5ug, the confluency is 0.949, which is still above the 70% threshold and thus classified as nontoxic. When the density is increased to 20ug, the confluency reduces to 0.635, falling below the toxicity threshold and indicating toxicity. At the highest density of 50ug, the confluency drops significantly to 0.342, confirming the toxic nature of this condition.

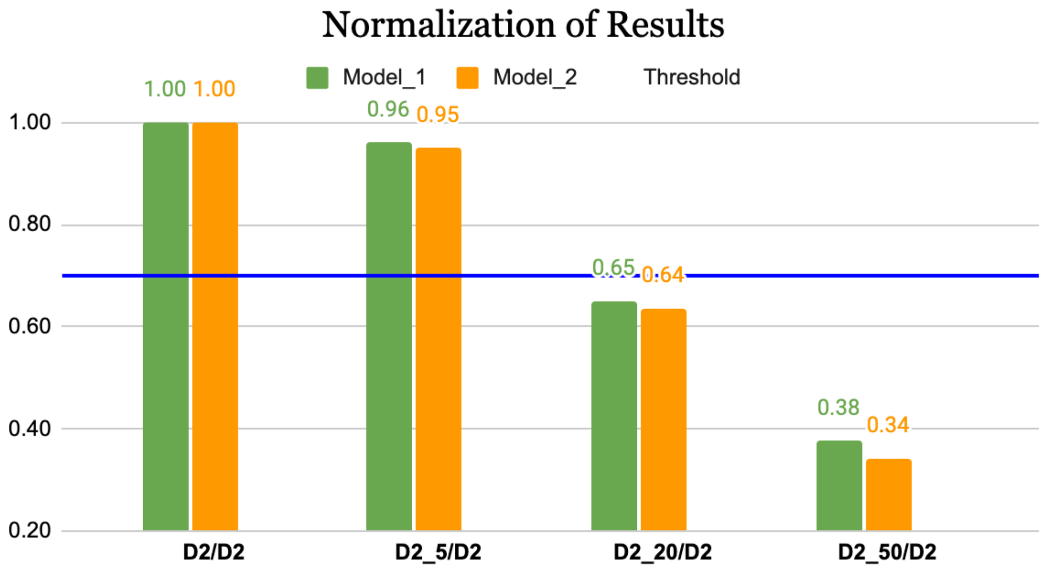


Figure 25: Normalization of PAR30 results dataset_1

Both Model_1 and Model_2 exhibit a clear trend regarding the toxicity of PAR30 at varying densities. At the lower density of 5ug per Day 2, the confluency is high, and the condition is deemed nontoxic in both models. However, as the density increases to 20ug and further to 50ug, the confluency values decrease significantly, falling below the 70% threshold and indicating toxicity. It can be said that with density of 20ug the cell is close to the threshold where it is starting to become toxic. It can be concluded that higher densities result in reduced confluency and increased toxicity. This pattern is consistently observed across both models, reinforcing the reliability of the confluency measurements in determining the toxicity of biomaterial densities.

4.1.3.2 Dataset_2: PAR30

Table 14: Average and STDev (a) model_1 and (b) model_2

			Model_1		
Day	***D1***	***D2***	***D2_5***	***D2_20***	***D2_50***
Average	45.284	79.880	83.055	45.054	41.255
STDev	18.408	19.675	14.918	24.552	22.533

(a)

Model_2					
Day	***D1***	***D2***	***D2_5***	***D2_20***	***D2_50***
Average	46.878	80.874	84.467	43.739	39.575
STDev	19.735	19.355	14.998	25.271	23.249

(b)

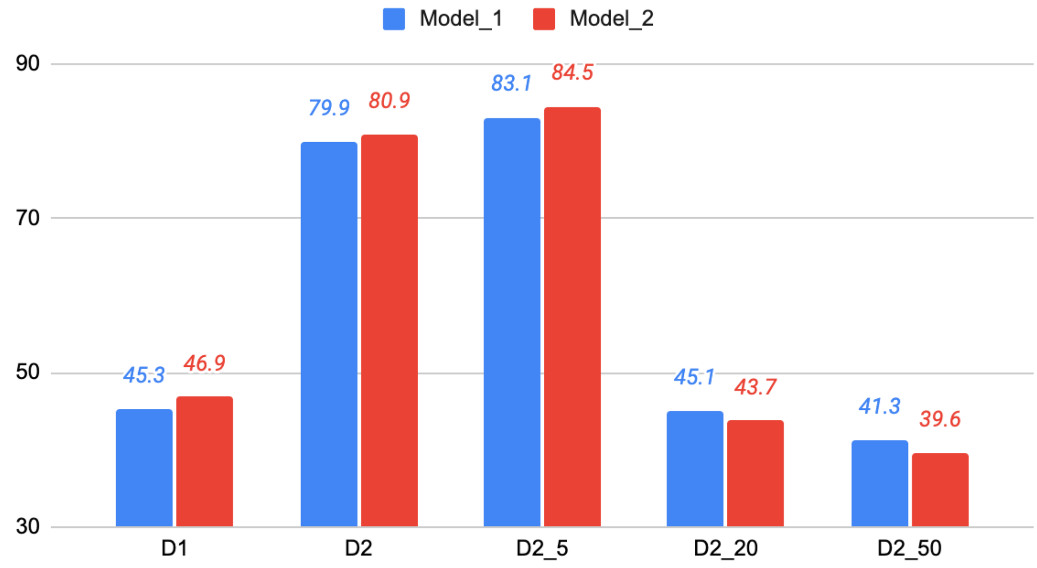


Figure 26: Average Cell Confluency Models

The table above compares the confluency results for each day across two models. Both models show similar trends in their confluency predictions, with differences generally around 1-2%. This indicates that both models have comparable capabilities in predicting cell confluency.

From the table, we can analyze the trends observed for each model with respect to different biomaterial densities over time. On Day 1, the baseline confluency is approximately 45% for both models, which serves as a starting point for comparison. On Day 2, there is a significant increase in confluency, with Model_1 reaching 79.9% and Model_2 at 80.9%. This indicates considerable cell growth over time without the presence of additional biomaterial, reflecting healthy cell growth.

When examining the effect of different biomaterial densities, we observe that for the 5ug density, both models show high confluency values of 83.1% for Model_1 and 84.5% for Model_2 on Day 2.5. This suggests that a low density of biomaterial does not significantly hinder cell growth, maintaining high confluency levels.

However, at a density of 20ug, there is a notable drop in confluency, with Model_1 at 45.1% and Model_2 at 43.7%. This substantial decrease, approximately 30% from Day 2, indicates that a medium density of biomaterial has a significant inhibitory effect on cell confluency, suggesting potential negative impacts on cell health or viability.

At the highest density of 50ug, the lowest confluency values are observed: 41.3% for Model_1 and 39.6% for Model_2. This further reduction in confluency indicates a strong negative impact of the high concentration of biomaterial on cell confluency. The results imply that higher biomaterial concentrations severely limit cell health or cause significant cell death.

Table 15: Ratios of biomaterials over Day2

Ratio	***Model_1***	
5/D2	1.040	Nontoxic
20/D2	0.564	Toxic
50/D2	0.516	Toxic

(a)

Ratio	***Model_2***	
5/D2	1.044	Nontoxic
20/D2	0.541	Toxic
50/D2	0.489	Toxic

(b)

The tables provided present the ratios of biomaterial densities D2_5, D2_20, and D2_50 relative to the baseline density D2, along with the confluency results for two models, Model_1 and Model_2. These ratios serve as indicators of the impact of different biomaterial densities on cell confluency.

In the first table, the ratios for Model_1 suggest that a density of 5ug leads to a slight increase in confluency, as the ratio is slightly above 1. However, for densities of 20ug and 50ug, there is a substantial decrease in confluency, with the ratios well below 1.

In the second table, the ratios for Model_2 show a similar pattern to Model_1, with a slight increase in confluency at a density of 5ug, followed by significant decreases at densities of 20ug and 50ug. Again, this suggests that higher densities of biomaterial are associated with reduced cell confluency, indicating potential toxicity.

It can be concluded that a density of 5ug is nontoxic, while densities of 20ug and 50ug are toxic. From the results, it can be seen that when increasing the density, the cells become more toxic.

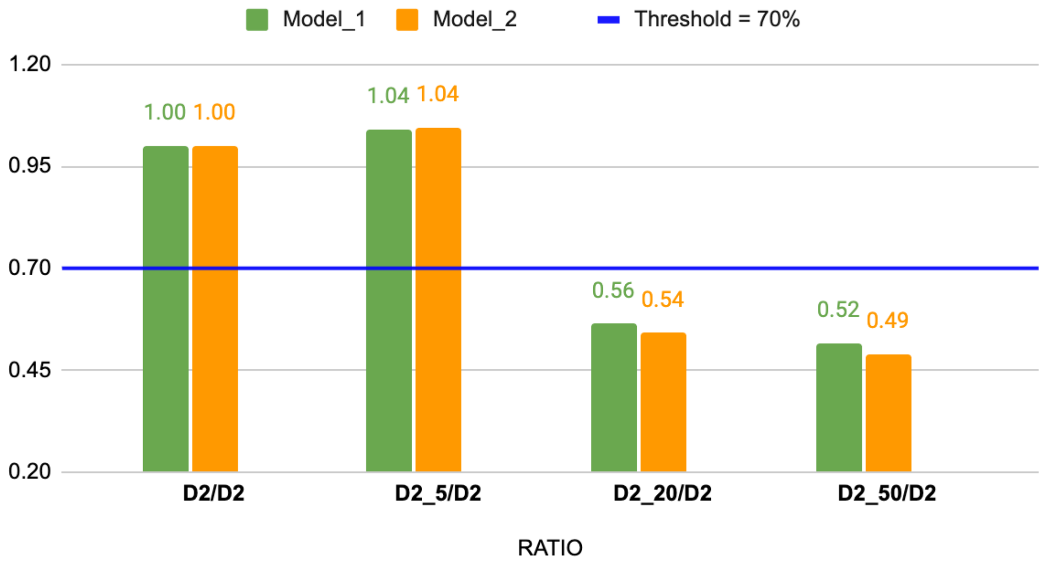


Figure 27: Ratio of Cell Confluence using Model1 and Model2

4.1.3.3 Comparison between PAR30

The confluency results obtained from cell images captured on two different dates but of the same biomaterial, PAR30, are being compared. By comparing the confluency results from these images, insights can be gained into how the biomaterial and its density affects cell growth and health over time. The comparison aims to determine if there are any significant differences in cell confluency between the two dates.

Table 16: Cell Confluence Comparison

Ratio: Model_1			
PAR30 :	***Dataset_1***	***Dataset_2***	***MTT***
5/D2	0.961	1.040	Healthy
20/D2	0.651	0.564	Nonhealthy
50/D2	0.376	0.516	Nonhealthy

(a)

Ratio: Model_2			
PAR30 :	***Dataset_1***	***Dataset_2***	***MTT***
5/D2	0.949	1.044	Healthy
20/D2	0.635	0.541	Nonhealthy
50/D2	0.342	0.489	Nonhealthy

(b)

The provided tables present the confluency results from cell images captured on two different dates, Dataset_1, Dataset_2, for two models, Model_1 and Model_2. The confluency values represent the proportion of the cell culture surface covered by cells, with values above 70% indicating nontoxicity and values below indicating toxicity.

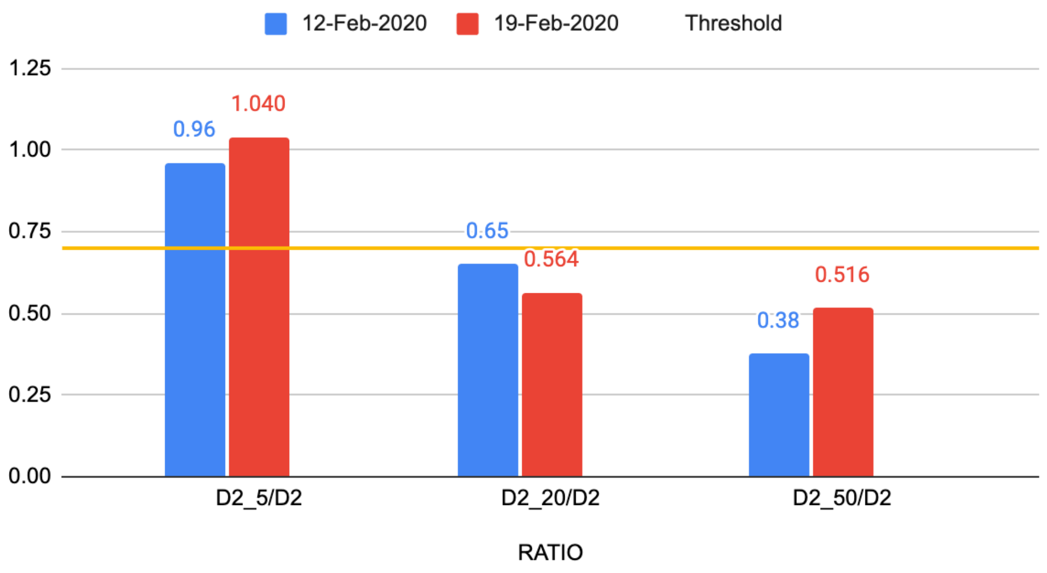


Figure 28: Results Model1

In the above graph, model_1 is used to compare the confluency results (normalized) over two datasets. On the first dataset, the difference between density 20ug to 50 ug is a decrease of 28%. While on the second dataset, the difference between them is a smaller decrease of roughly 5%.

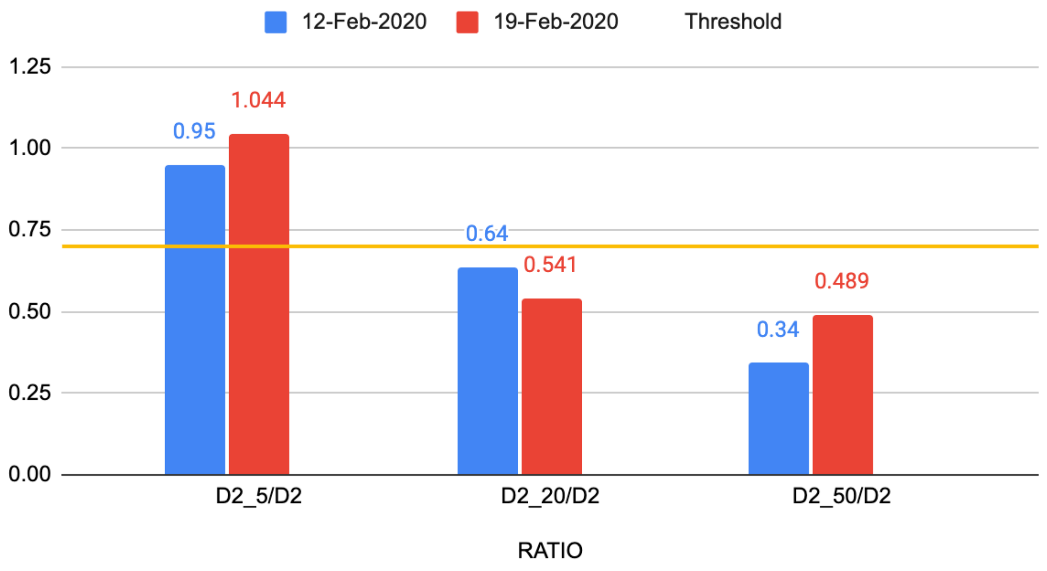


Figure 29: Results Model2

In the above graph, model_2 is used to compare the confluency results (normalized) over two datasets. On the first dataset, the difference between density 20ug to 50 ug is a decrease of 30%. While on the second dataset, the difference between them is a smaller decrease of 5%.

4.1.3.4 PLL250

Table 17: Average and STdev using both models

Model_1				
Day	***D1***	***D2***	***D2_5***	***D2_500***
Average	45.284	79.880	59.371	31.916
STD	18.245	19.675	16.421	21.840

(a)

Model_2				
Day	***D1***	***D2***	***D2_5***	***D2_500***
Average	46.878	80.874	60.891	28.097
STD	19.560	19.355	17.035	20.007

Table 18: Toxicity results for dataset 3

Ratio	***Model_1***	
5/D2	0.743	Nontoxic
500/D2	0.400	Toxic

(a)

Ratio	***Model_1***	
5/D2	0.753	Nontoxic
500/D2	0.347	Toxic

The table presents the ratio of cell confluency on Day 2 under different concentrations of PLL250 to the baseline confluency on Day 2 without PLL250 for two models. The ratios are calculated to assess the potential cytotoxicity of PLL250 based on a toxicity threshold, where a ratio greater than 0.7 indicates non-toxicity and a ratio of 0.7 or below suggests toxicity.

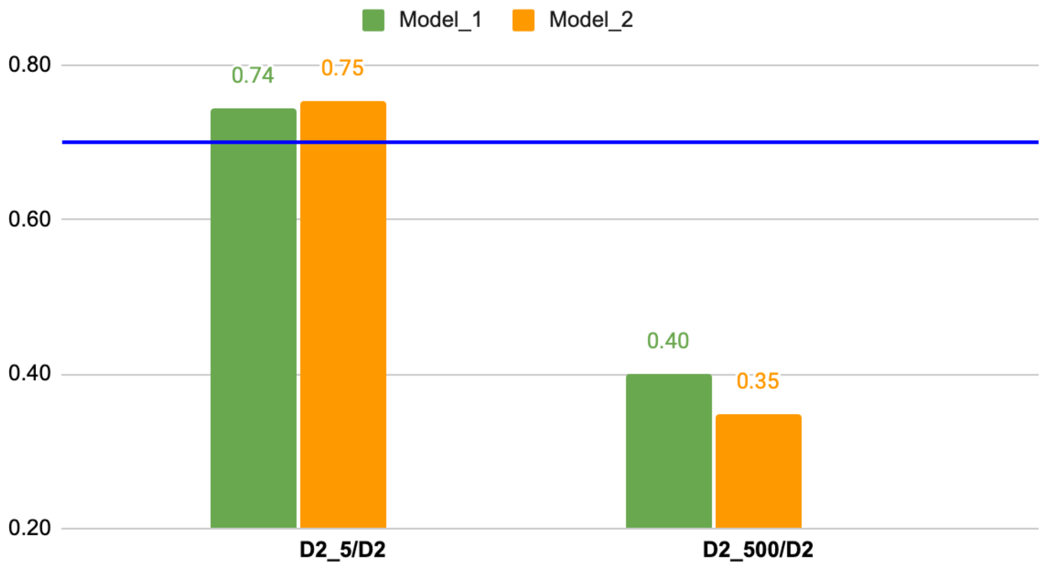


Figure 30: Normalization for dataset 3

For the condition with 5 µg of PLL250, the ratio of confluency to the baseline Day 2 confluency is 0.74 for Model 1 and 0.75 for Model 2. These ratios are both above the 0.7 threshold, indicating that 5 µg is considered non-toxic by both models, suggesting that this concentration does not adversely affect cell growth to a level of concern.

On the other hand, for a density of 500 µg, the ratios drop significantly, with Model 1 showing a ratio of 0.40 and Model 2 a ratio of 0.35. These ratios are well below the 0.7 threshold, indicating that 500 µg of PLL250 is toxic to the cells.

4.1.3.5 Comparison of all results

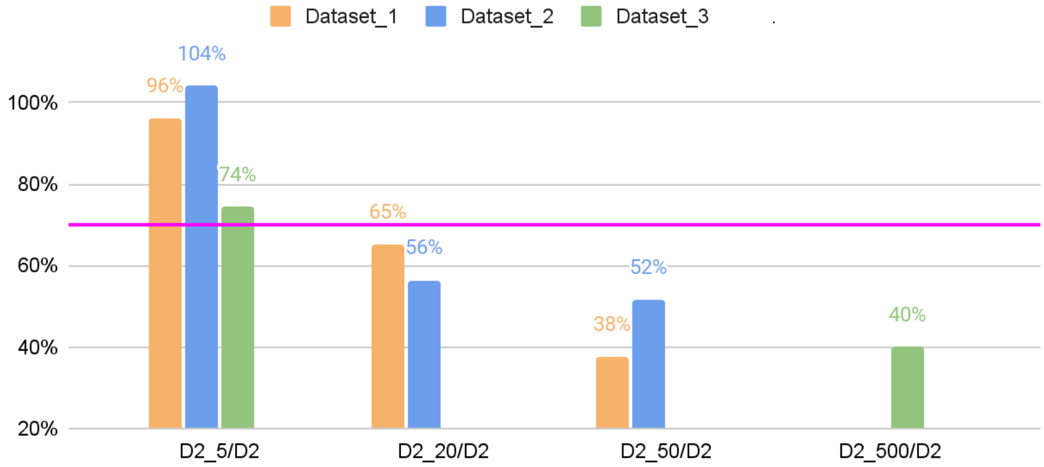


Figure 31: Comparison all three datasets, based on their confluence ratio, with model_1

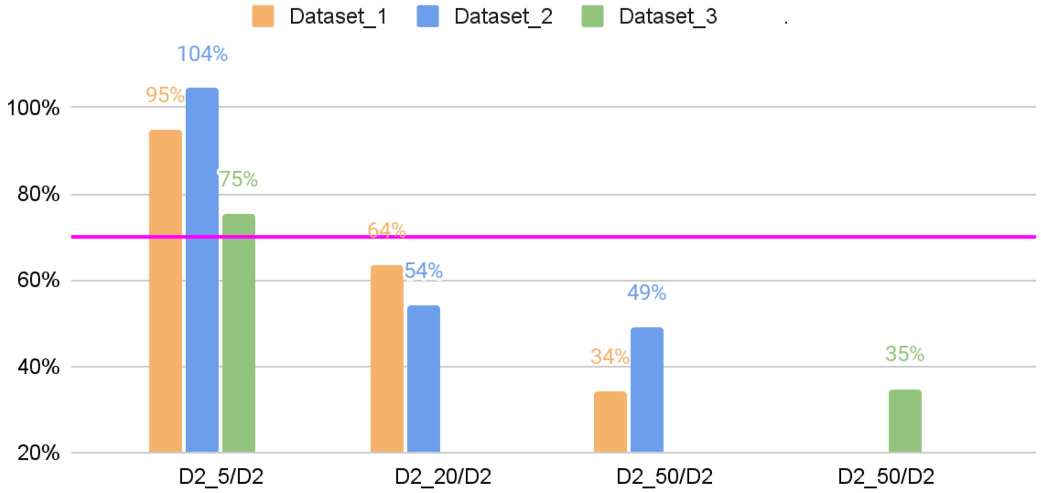
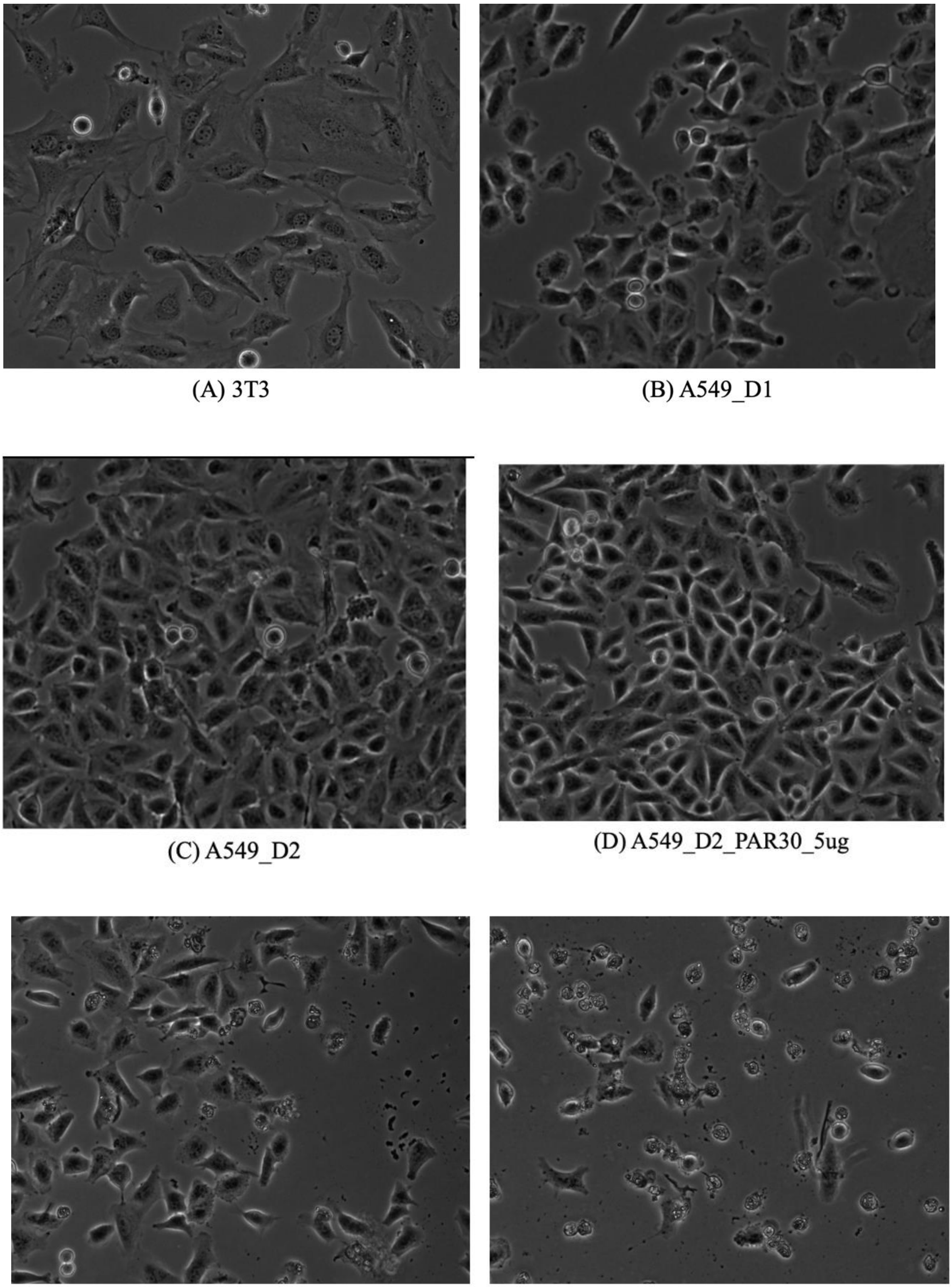


Figure 32: Comparison all three datasets, based on their confluence ratio, with model_2

4.2 Classification Phase

Representative images used for classification.



(A) 3T3

(B) A549_D1

(C) A549_D2

(D) A549_D2_PAR30_5ug

(E) A549_D2_PAR30_20ug

(F) A549_D2_PAR30_50ug

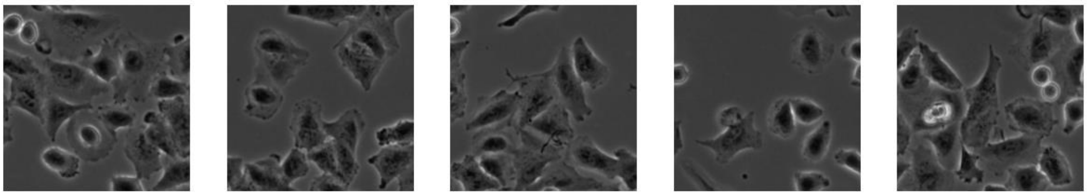
4.2.1 A549 and 3T3

4.2.1.1 KMeans

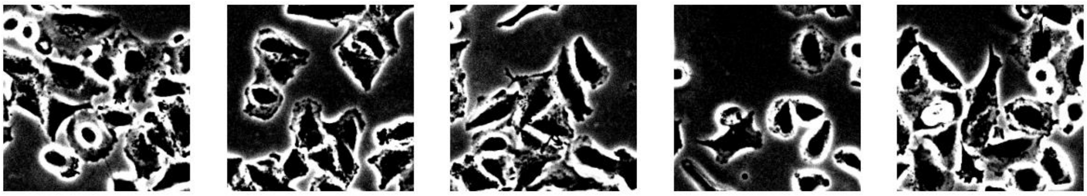
The first approach was to classify with unsupervised learning cell A549 and cell 3t3.

Using simply Kmeans to classify between two classes, without any preprocessing techniques it was hard to achieve high accuracy results. In this case, the highest accuracy achieved was 12%.

On the other hand, when implementing t-SNE and PCA for feature extraction, and Z-score normalization technique the accuracy results increased.



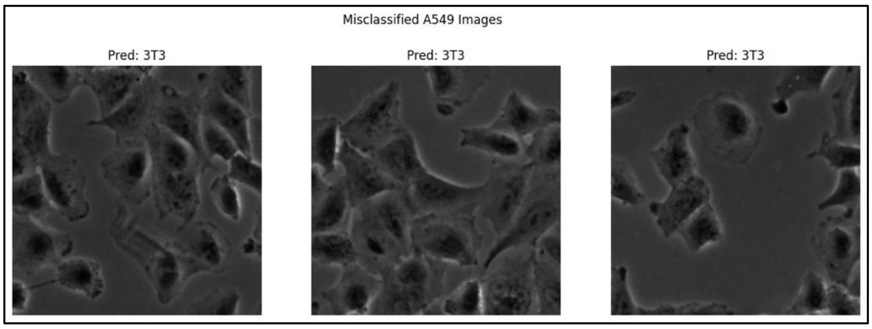
(a) Min-Max Normalization Technique

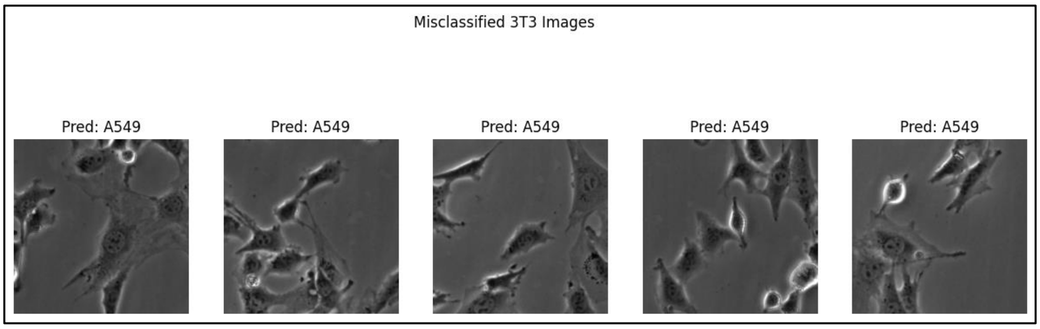


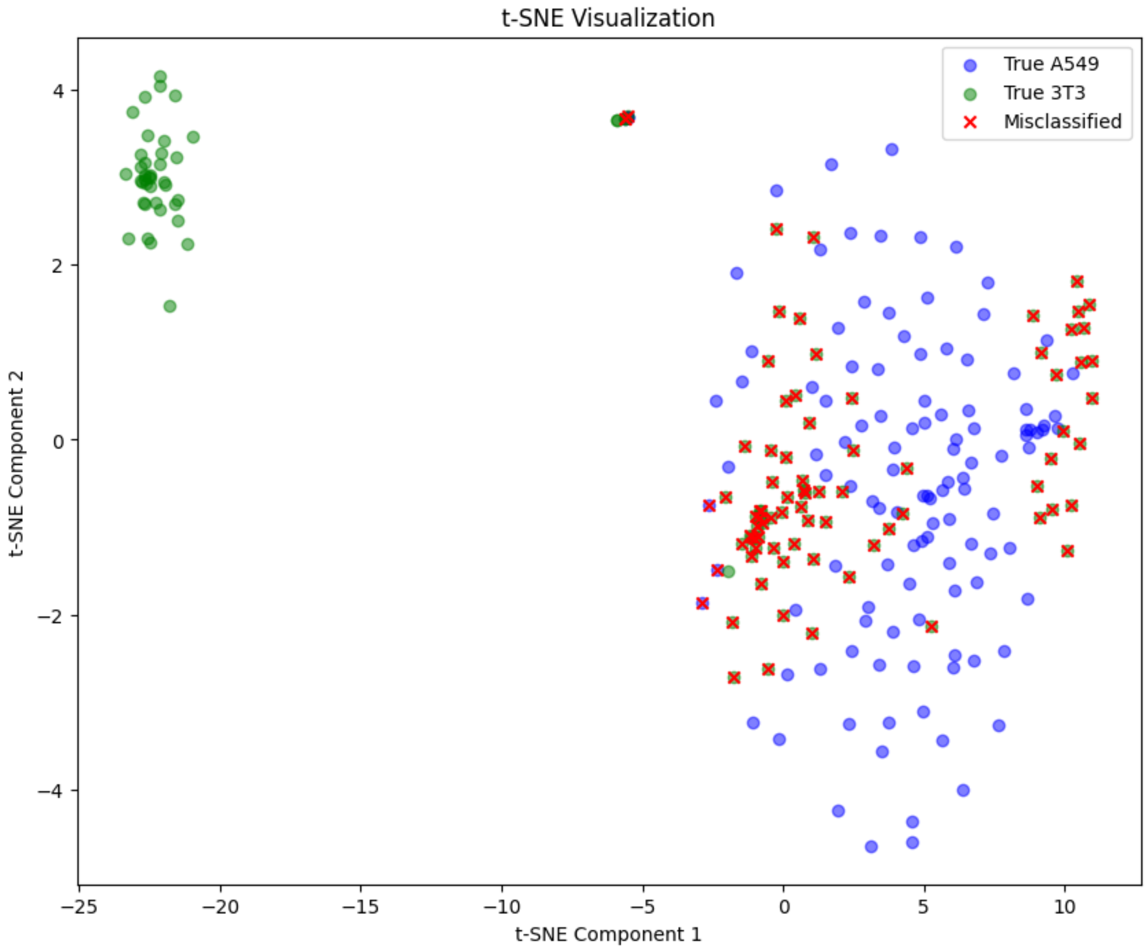
(a) Z-Score Normalization Technique

```
Classification Report Using KMeans for A549 and 3T3:
precision    recall  f1-score   support


A549         0.60       0.97       0.74        119
3T3          0.93       0.34       0.50        119


accuracy                                      0.66         238
macro avg              0.76       0.66       0.62         238
weighted avg           0.76       0.66       0.62         238
```

Model_1_1



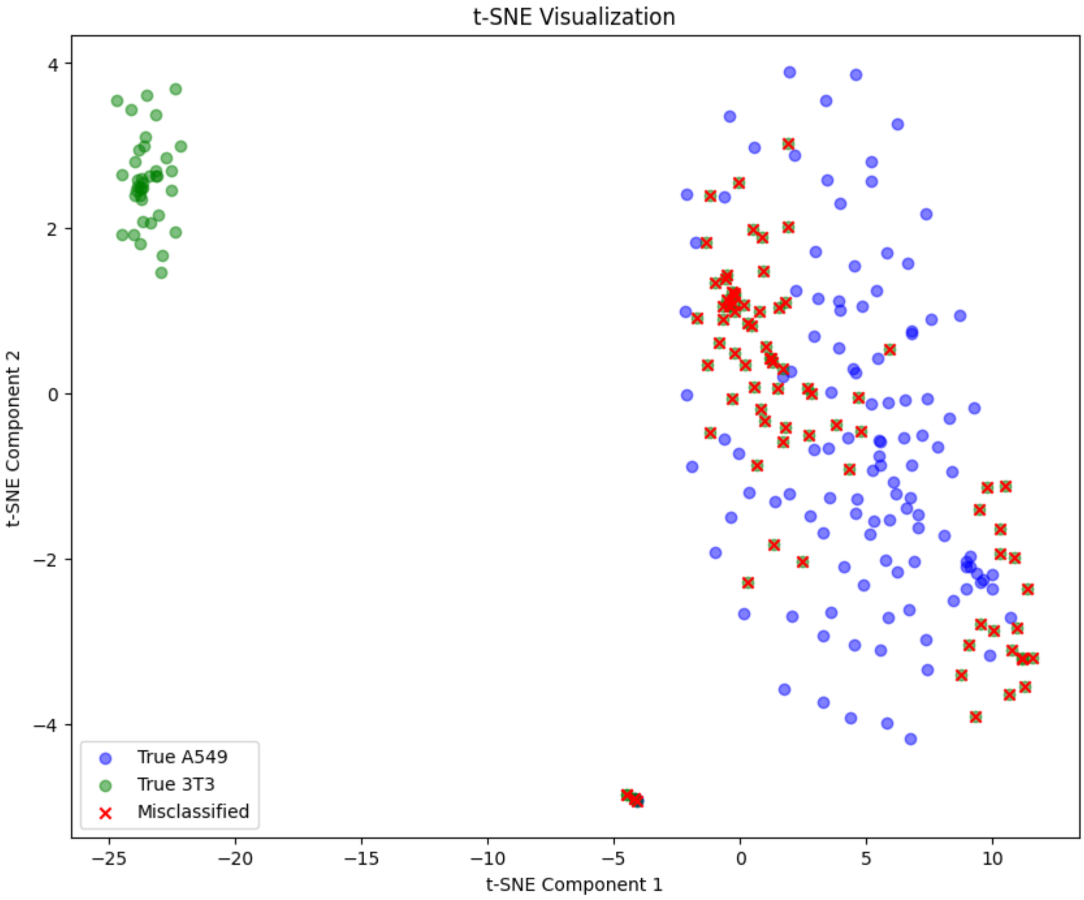


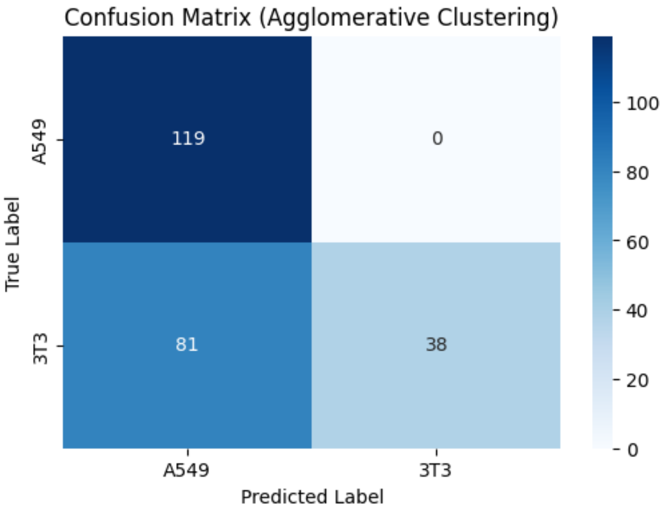


In the above model with KMeans, a total of 81 images were misclassified. 3 of which were A549 cells and 78 images were 3T3 cells. Several combination of normalization techniques, feature extractions and parameters were tried, but it came to a point where the accuracy was not changing.

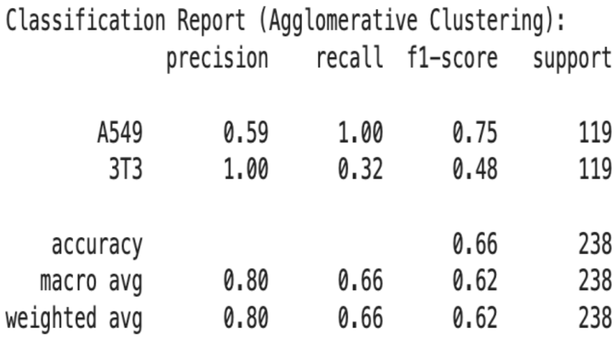
4.2.1.2 Agglomerative Clustering

In this section, the best model from the above section is used, but with another unsupervised classifier: Agglomerative Clustering. Using this classifier, there were a total of 81 misclassified images, but A549 was classified correctly, and all the misclassified images were from the 3T3 class.





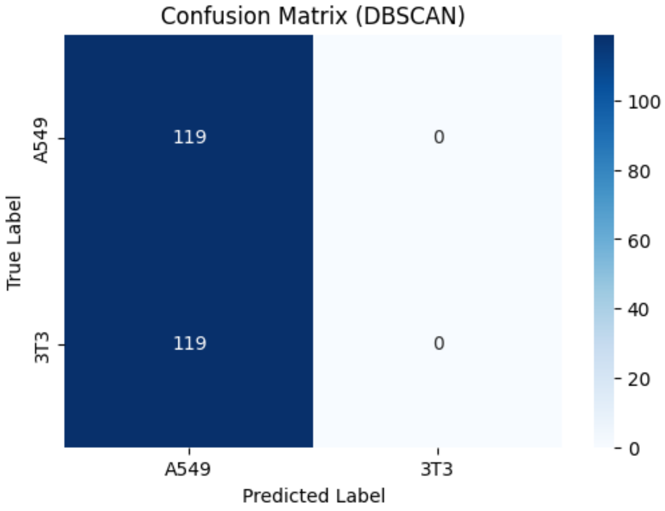
Model_1_2

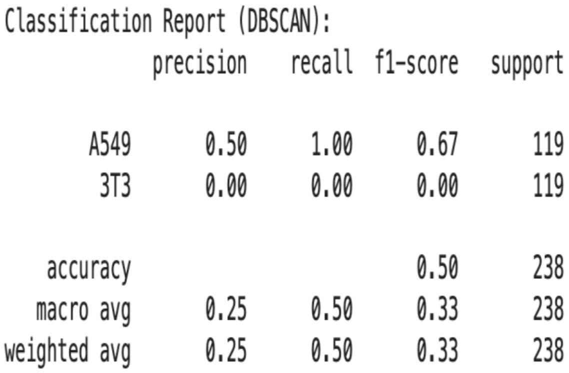


```
Classification Report (Agglomerative Clustering):
              precision    recall  f1-score   support

        A549       0.59      1.00      0.75       119
         3T3       1.00      0.32      0.48       119

    accuracy                           0.66       238
   macro avg       0.80      0.66      0.62       238
weighted avg       0.80      0.66      0.62       238
```

4.2.1.2 DBSCAN Clustering

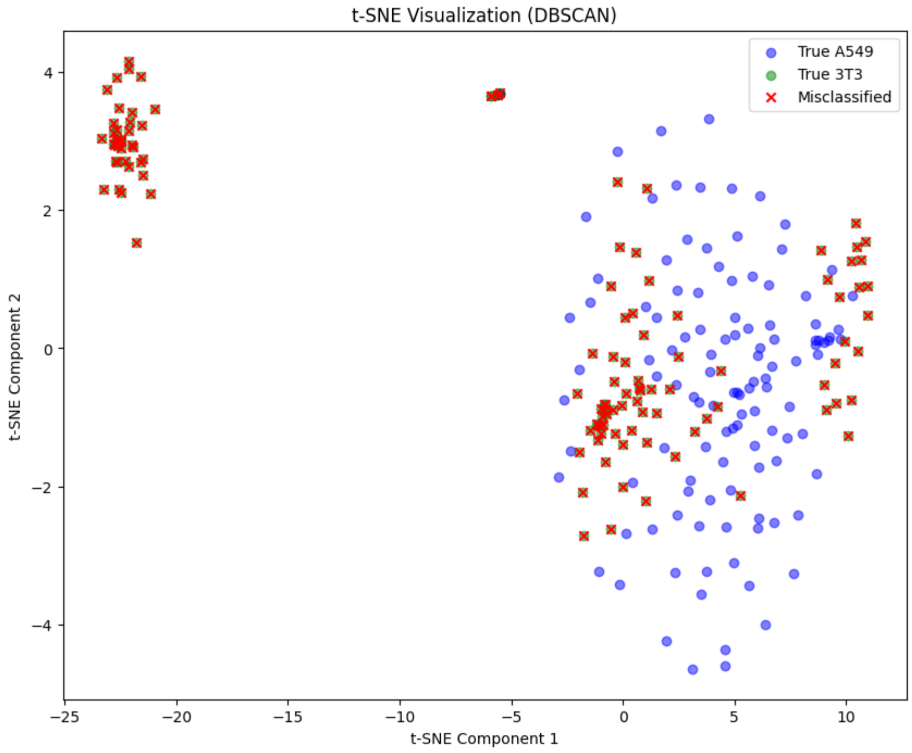
In this section, the best model from the above section is used, but with another unsupervised classifier: DBSCAN. Using this classifier, there were a total of 119 misclassified images, which means one class (in this case 3T3) was not classified as different at all. Both classes were classified as one with DBSCAN.





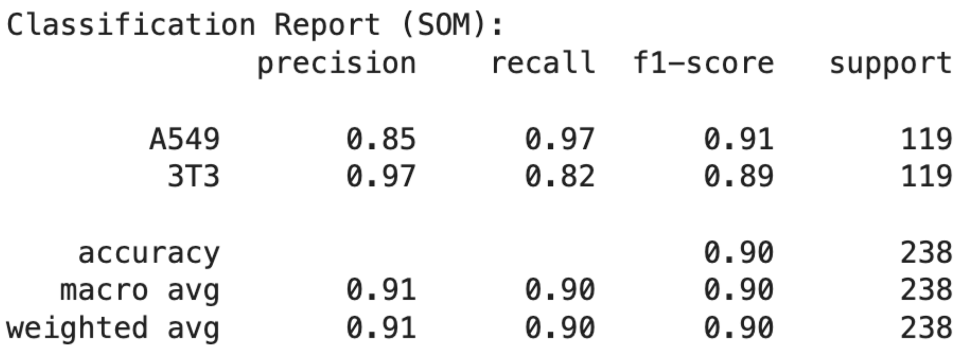
```
Classification Report (DBSCAN):
              precision    recall  f1-score   support

        A549       0.50      1.00      0.67       119
         3T3       0.00      0.00      0.00       119

    accuracy                           0.50       238
   macro avg       0.25      0.50      0.33       238
weighted avg       0.25      0.50      0.33       238
```

Model_1_3



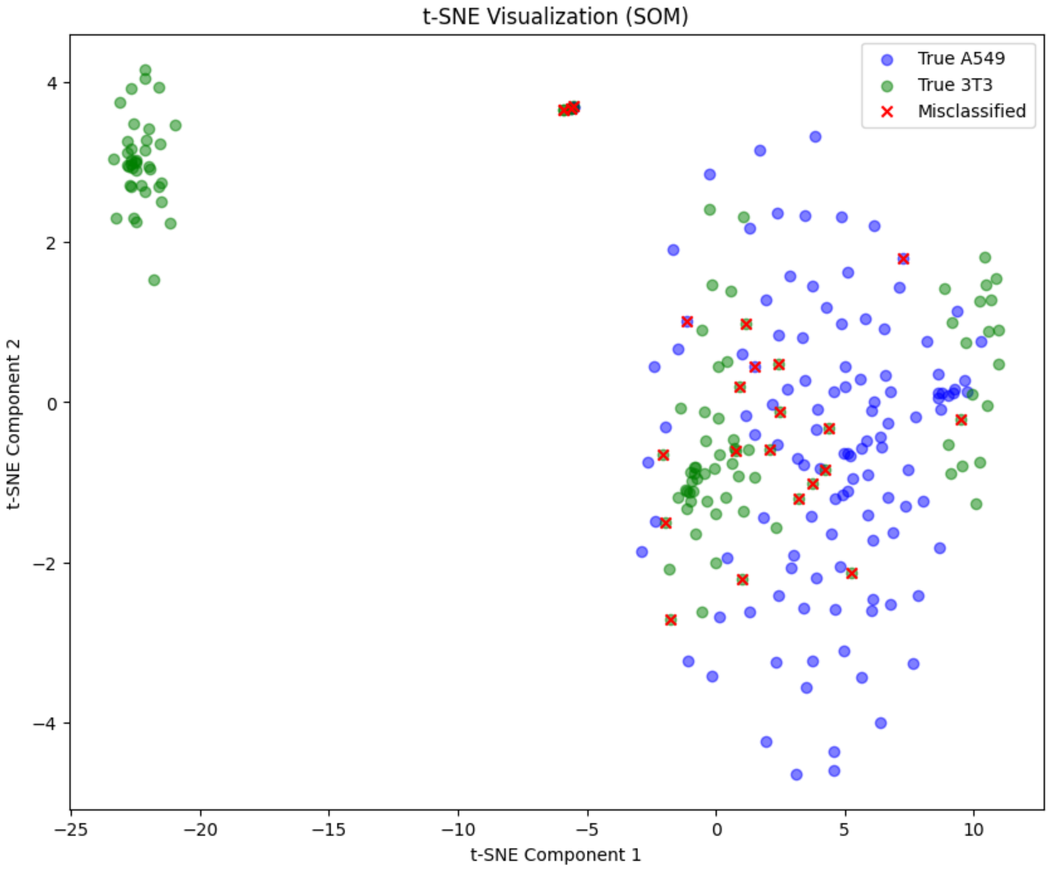
4.2.1.2 SOM Clustering

In this section, the best model from the above section is used, but with another unsupervised classifier: SOM (Self-Organizing Maps). Self-Organizing Maps (SOM) are a type of unsupervised learning neural network that can be used for clustering. Using this classifier, there were a total of 24 misclassified images, 3 from A549 class and 21 from 3T3 class. This unsupervised method yielded an accuracy of 90%.



```
Classification Report (SOM):
              precision    recall  f1-score   support

        A549       0.85      0.97      0.91       119
         3T3       0.97      0.82      0.89       119

    accuracy                           0.90       238
   macro avg       0.91      0.90      0.90       238
weighted avg       0.91      0.90      0.90       238
```

Model_1_4



4.2.2 Classification Between Different Densities - 4 classes

4.2.1.1 VGG16+KMeans

The second try was to use KMeans to cluster between the 3 densities of biomaterials used in A549. In this model, VGG16 is used for feature extraction.

Adjusted Rand Index: 0.31

Accuracy: 0.22

```
Classification Report:

             precision    recall  f1-score   support



          d2       0.42      0.76      0.54       120

        d2-5       0.00      0.00      0.00       120

       d2-20       0.11      0.12      0.11       120

       d2-50       0.00      0.00      0.00       120



    accuracy                           0.22       480

   macro avg       0.13      0.22      0.16       480

weighted avg       0.13      0.22      0.16       480
```

Model_2

4.2.1.2 VGG16+PCA+KMeans

Model_3 is the same as the above one, but it also has PCA for feature extraction and dimensionality reduction.

Adjusted Rand Index: 0.52

Accuracy: 0.67

```
Classification Report:
              precision    recall  f1-score   support

          d2       0.27      0.05      0.08       120
        d2-5       0.47      0.91      0.62       120
       d2-20       0.85      0.74      0.79       120
       d2-50       0.95      0.97      0.96       120

    accuracy                           0.67       480
   macro avg       0.64      0.67      0.61       480
weighted avg       0.64      0.67      0.61       480
```

Model_3

4.2.1.3 ResNet50+PCA+KMeans

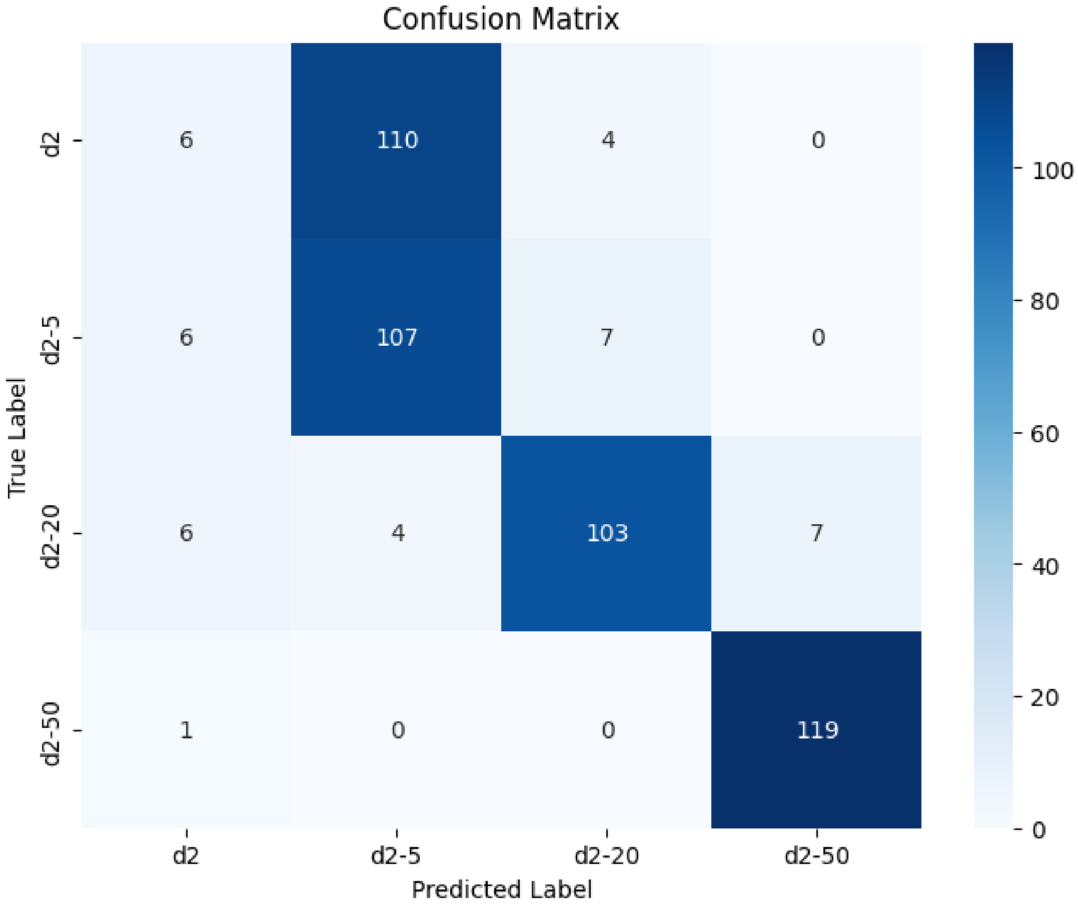
Model_4 uses ResNet 50 instead of VGG16.

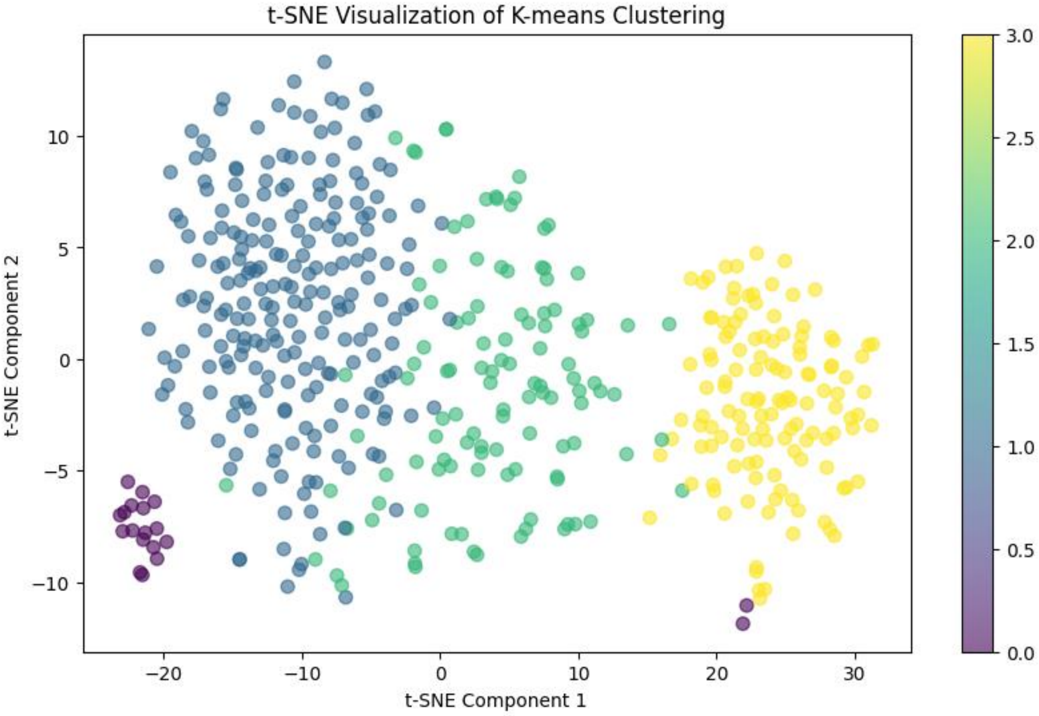
Adjusted Rand Index: 0.60

Accuracy: 0.70

```
Classification Report:

              precision    recall  f1-score   support


          d2       0.33      0.05      0.09       120

        d2-5       0.48      0.89      0.63       120

       d2-20       0.90      0.85      0.88       120

       d2-50       0.94      1.00      0.97       120


    accuracy                           0.70       480

   macro avg       0.67      0.70      0.64       480

weighted avg       0.67      0.70      0.64       480
```

Model_4


Confusion Matrix
True Label
Predicted Label
d2
d2-5
d2-20
d2-50
6
110
4
0
6
107
7
0
6
4
103
7
1
0
0
119
0
20
40
60
80
100


t-SNE Visualization of K-means Clustering
t-SNE Component 1
t-SNE Component 2
−20
−10
0
10
20
30
10
5
0
−5
−10
0.0
0.5
1.0
1.5
2.0
2.5
3.0

1.2.2 4.2.1.4 ResNet101+PCA+KMeans

Model_5 is the same as Model_4, but instead of ResNet50 uses ResNet101.

Adjusted Rand Index: 0.52

Accuracy: 0.31

```
Classification Report:
              precision    recall  f1-score   support

          d2       0.49      0.45      0.47       120
        d2-5       0.00      0.00      0.00       120
       d2-20       0.91      0.78      0.84       120
       d2-50       0.00      0.00      0.00       120

    accuracy                           0.31       480
   macro avg       0.35      0.31      0.33       480
weighted avg       0.35      0.31      0.33       480
```

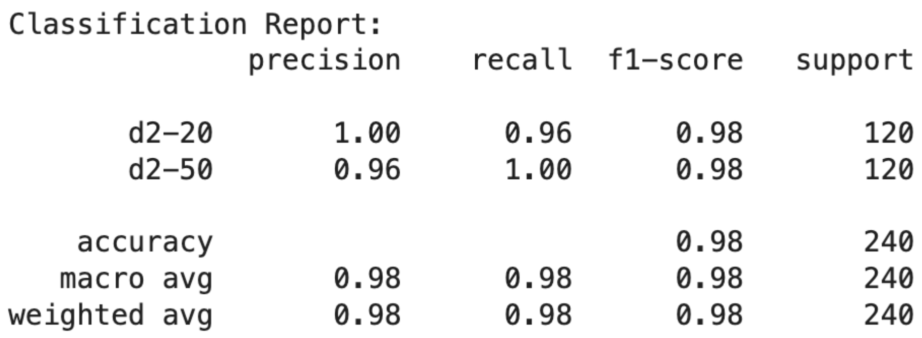
Model_5

4.2.3 Classification between 20ug and 50 ug

4.2.3.1 KMeans+ VGG16+ PCA

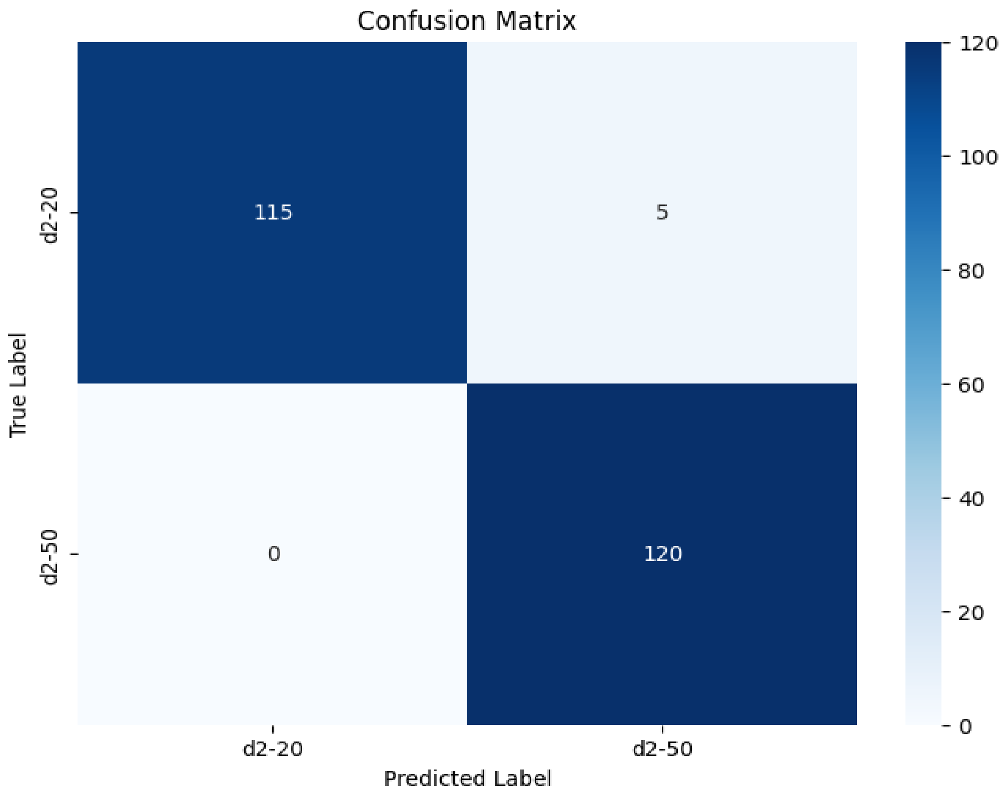
Adjusted Rand Index: 0.92

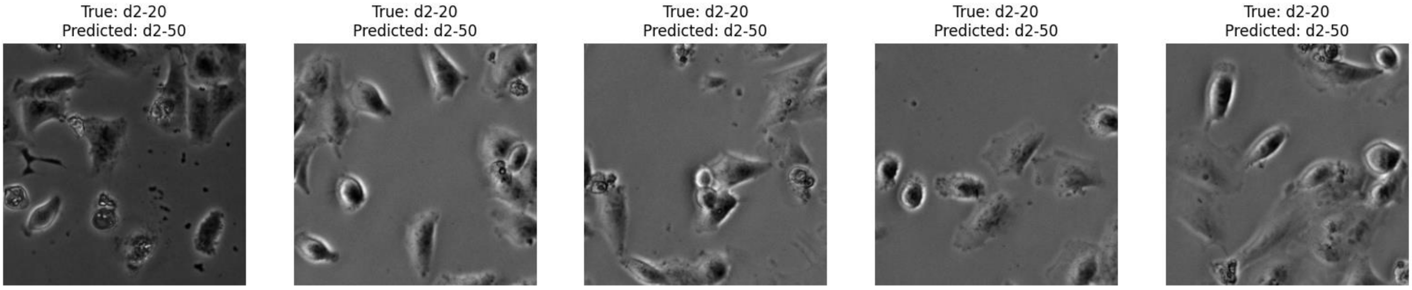
Accuracy: 0.98



```
Classification Report:
              precision    recall  f1-score   support

       d2-20       1.00      0.96      0.98       120
       d2-50       0.96      1.00      0.98       120

    accuracy                           0.98       240
   macro avg       0.98      0.98      0.98       240
weighted avg       0.98      0.98      0.98       240
```

Model_6





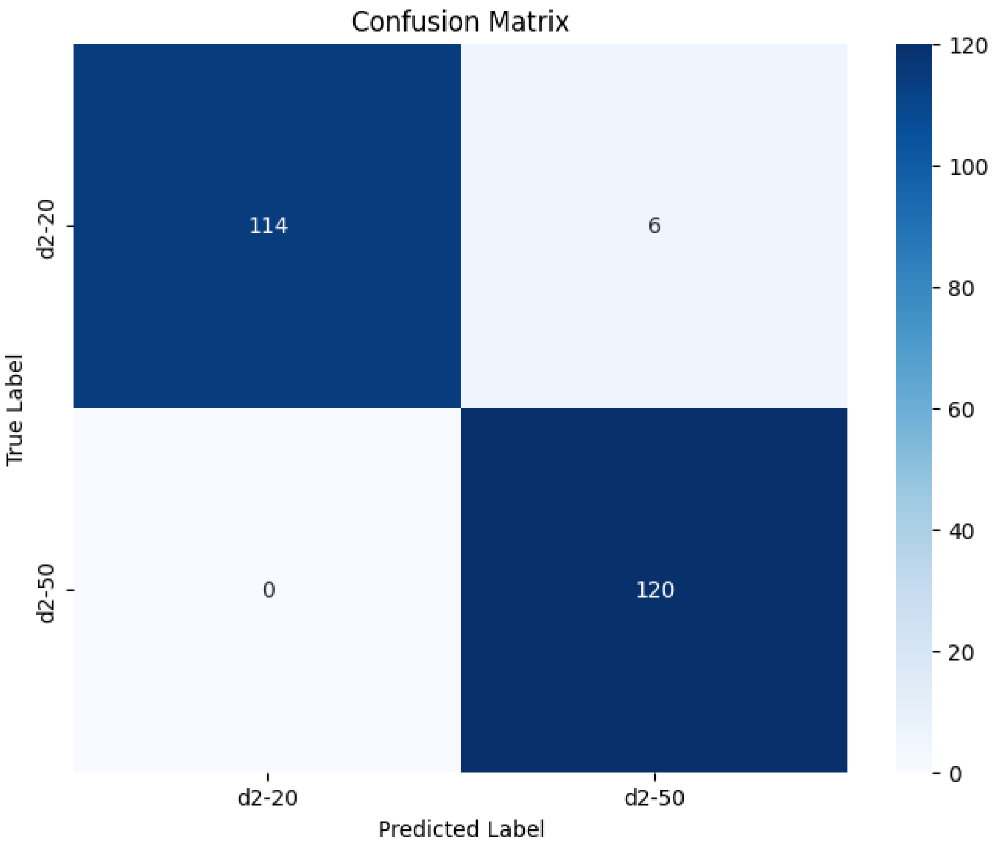
4.2.3.2 KMeans+ ResNet50+ PCA

Adjusted Rand Index: 0.91

Accuracy: 0.975

```
Classification Report:

              precision    recall  f1-score   support


       d2-20       1.00      0.95      0.97       120

       d2-50       0.95      1.00      0.98       120


    accuracy                           0.97       240

   macro avg       0.98      0.97      0.97       240

weighted avg       0.98      0.97      0.97       240
```

Model_7

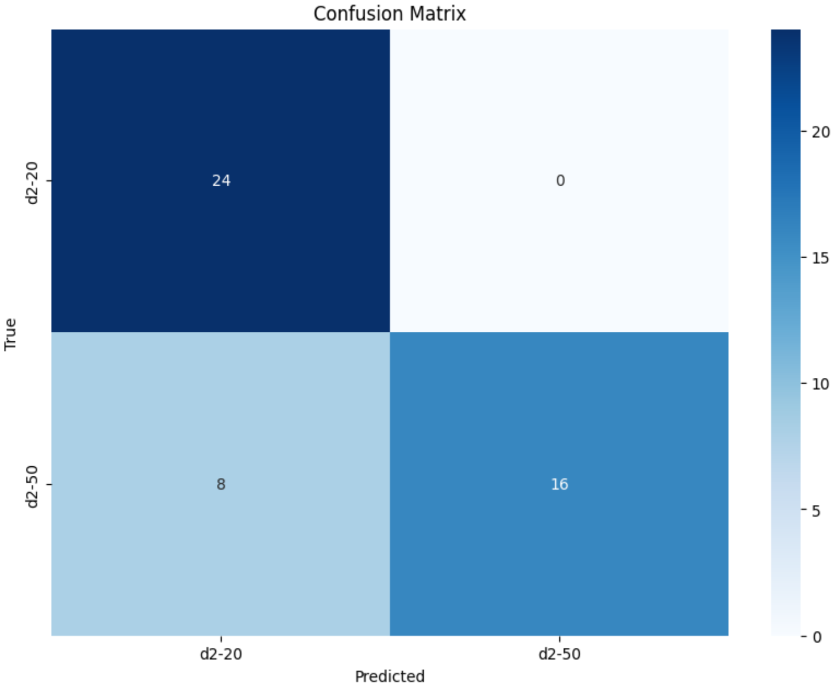


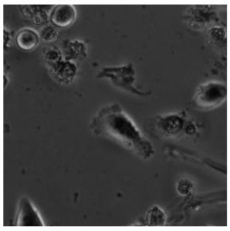
4.2.3.3 LeNet, 10 Epochs

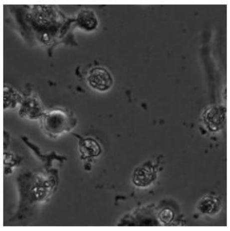
Test Accuracy: 0.83

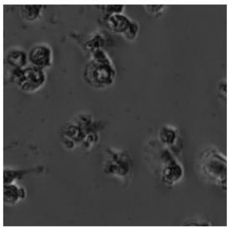
```
              precision    recall  f1-score   support

       d2-20       0.75      1.00      0.86        24

       d2-50       1.00      0.67      0.80        24

    accuracy                           0.83        48

   macro avg       0.88      0.83      0.83        48

weighted avg       0.88      0.83      0.83        48
```

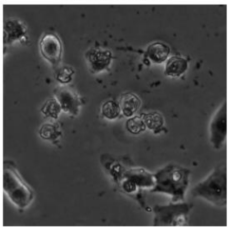
Model_8

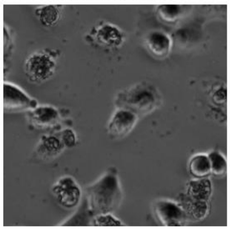


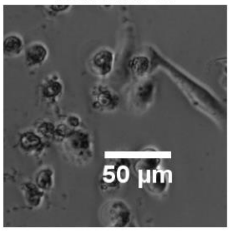

True: d2-50
Predicted: d2-20


True: d2-50
Predicted: d2-20

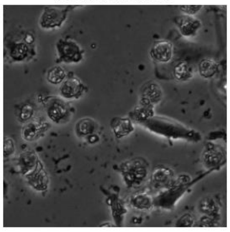

True: d2-50
Predicted: d2-20

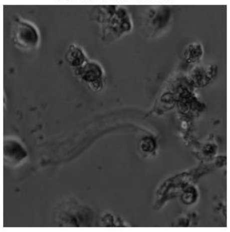

True: d2-50
Predicted: d2-20


True: d2-50
Predicted: d2-20



True: d2-50
Predicted: d2-20


True: d2-50
Predicted: d2-20

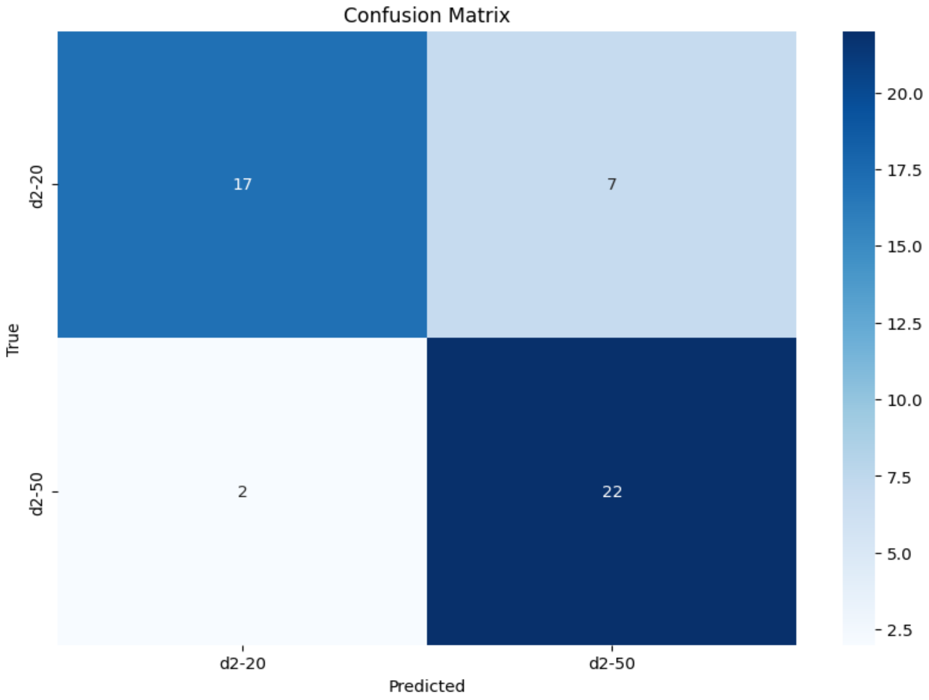

True: d2-50
Predicted: d2-20

4.2.3.3 LeNet, 100 Epochs

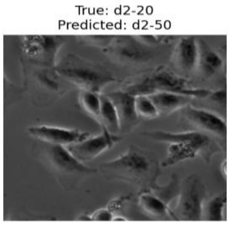
Test Accuracy: 0.81

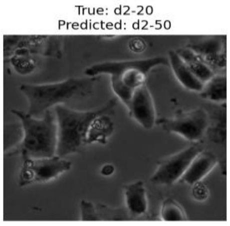
```
               precision    recall  f1-score   support

       d2-20       0.89      0.71      0.79        24

       d2-50       0.76      0.92      0.83        24

    accuracy                           0.81        48

   macro avg       0.83      0.81      0.81        48

weighted avg       0.83      0.81      0.81        48
```

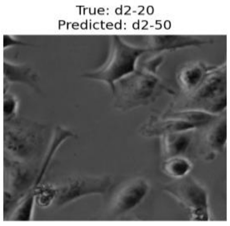
Model_9

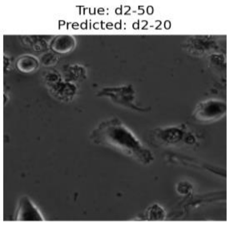


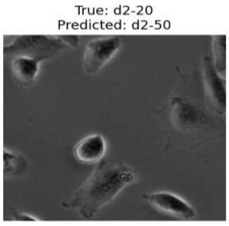
Total misclassified images: 9

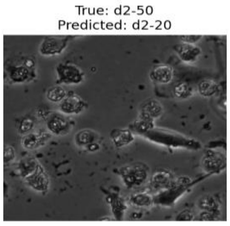


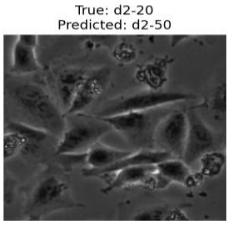


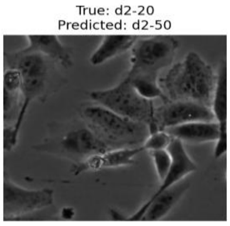


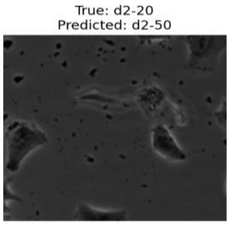












4.2.4 Classification between day2 and density 5

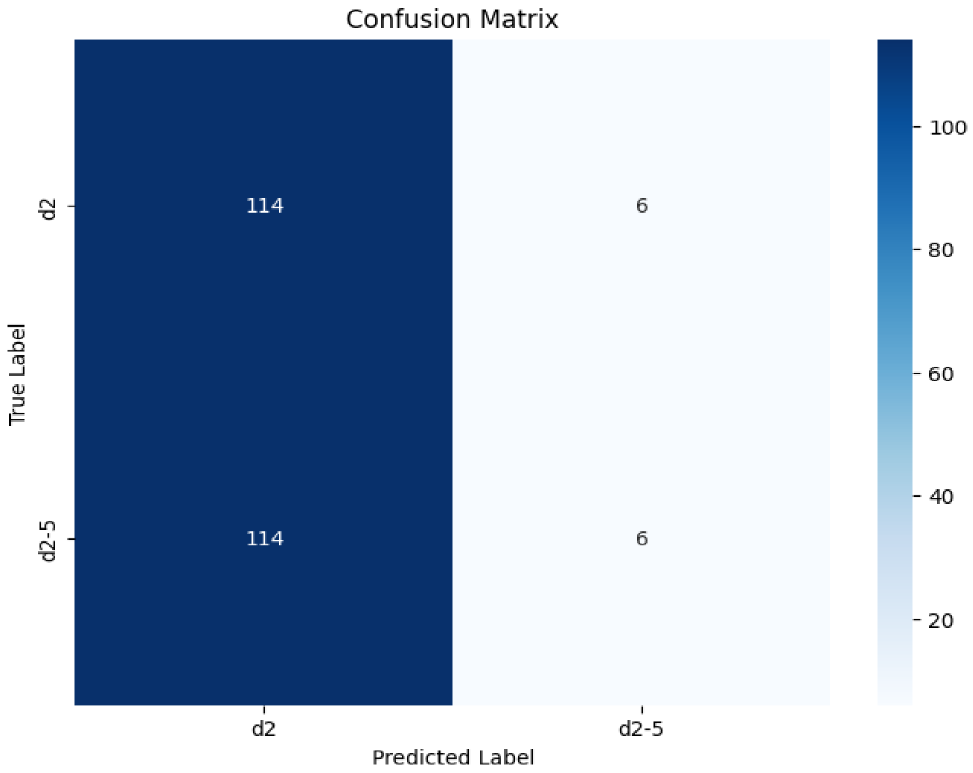
4.2.4.1 KMeans + VGG16 + PCA

Adjusted Rand Index: -0.0007956115740546878

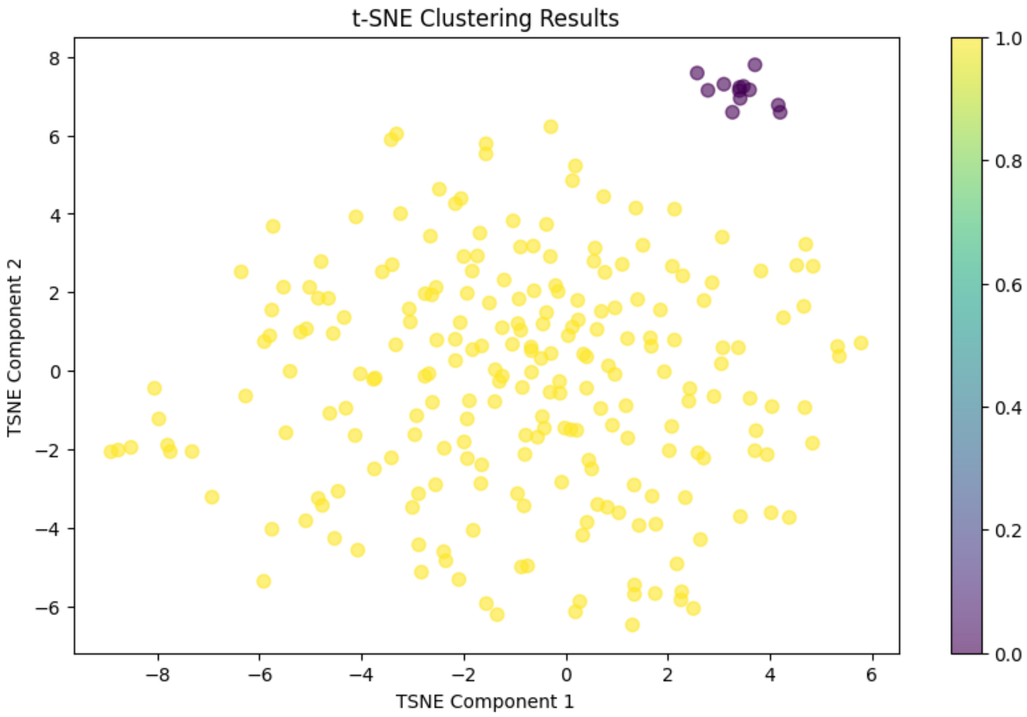
Accuracy: 0.5

```
Classification Report:

              precision    recall  f1-score   support

          d2       0.50      0.95      0.66       120

        d2-5       0.50      0.05      0.09       120

    accuracy                           0.50       240

   macro avg       0.50      0.50      0.37       240

weighted avg       0.50      0.50      0.37       240
```

Model_10



We can say that classifying between these two classes is quite hard, since all 240 images are quite similar to each other.



4.2.5 Classification between density 5ug and 20ug

4.2.5.1 KMeans + VGG16 + PCA

Adjusted Rand Index: 0.61

Accuracy: 0.11

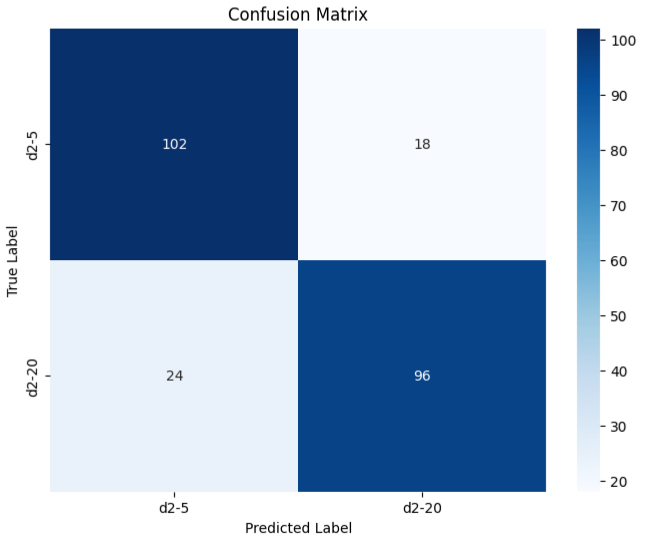
```
Classification Report:

              precision    recall  f1-score   support

        d2-5       0.05      0.04      0.04       120

       d2-20       0.15      0.17      0.16       120

    accuracy                           0.11       240

   macro avg       0.10      0.11      0.10       240

weighted avg       0.10      0.11      0.10       240
```

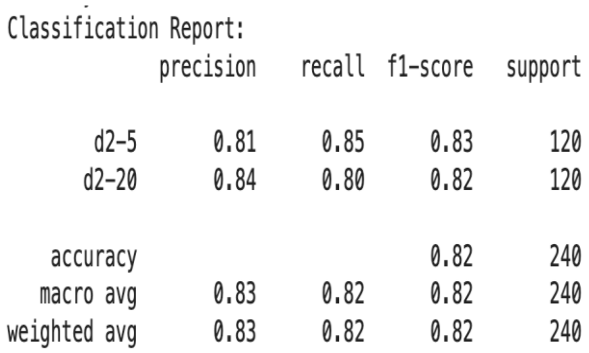
Model_11

4.2.5.2 KMeans + ResNet50 + PCA + tSNE

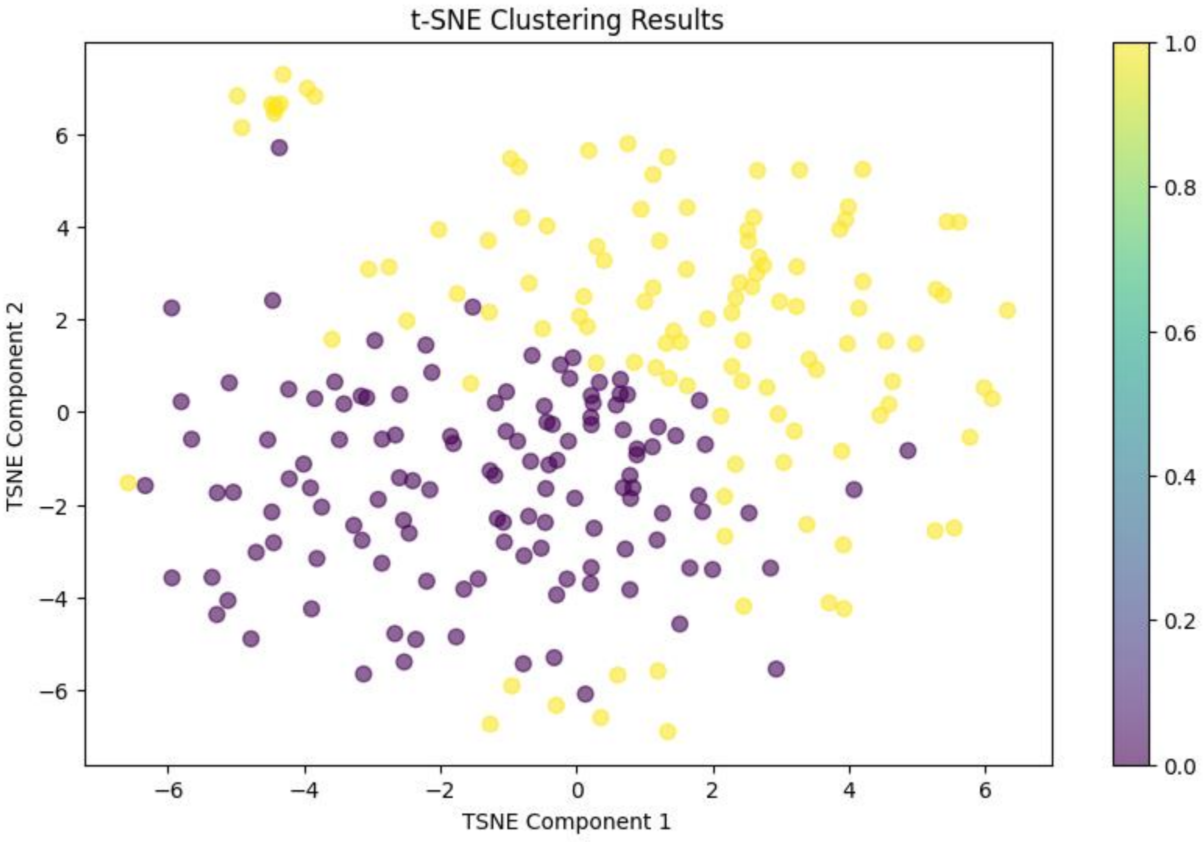
Adjusted Rand Index: 0.42

Accuracy: 0.825





```
Classification Report:
              precision    recall  f1-score   support

        d2-5       0.81      0.85      0.83       120
       d2-20       0.84      0.80      0.82       120

    accuracy                           0.82       240
   macro avg       0.83      0.82      0.82       240
weighted avg       0.83      0.82      0.82       240
```



4.3 Classification Results

Table 19: Classification of D2, D2_5ug, D2_20ug, and D2_50ug

Model_Name	Description	Model_Combination	Classes	Accuracy
Model_2	D2-D2_50	KMeans + VGG16	4	22%
Model_3	D2-D2_50	KMeans+PCA+VGG16	4	67%
Model_4	**D2-D2_50**	**KMeans+PCA+ResNet50**	**4**	**70%**
Model_5	D2-D2_50	KMeans+PCA+ResNet101	4	30.8%

The above table focuses on the classification of four specific classes (D2, D2_5ug, D2_20ug, and D2_50ug). Model 2, which uses KMeans and VGG16, starts with a relatively low accuracy of 22%, underscoring the need for additional improvements. When PCA is added in Model 3 alongside VGG16, the accuracy improves dramatically to 67%. This confirms the positive impact of PCA in enhancing the model's performance. Model 4, which employs KMeans, PCA, and ResNet50, achieves the highest accuracy in this set at 70%. This suggests that ResNet50, with its deeper architecture, is more effective for these classes than VGG16. However, Model 5, which uses ResNet101 instead of ResNet50, sees its accuracy fall to 30.8%, likely due to the increased model complexity leading to overfitting or other issues.

Table 20: Classification of D2_20ug and D2_50ug

Model_Name	Description	Model_Combination	Classes	Accuracy
Model_6	**20-50ug**	**KMeans+PCA+VGG16**	**2**	**98%**
Model_7	20-50ug	KMeans+PCA+ResNet50	2	97.5%
Model_8	20-50ug	LeNet - 10 epochs	2	83%
Model_9	20-50ug	LeNet - 100 epochs	2	81%

The above table shows the results of a binary classification between one biomaterial with different densities, 20ug and 50ug. Model 6, combining KMeans, PCA, and VGG16, achieves an impressive accuracy of 98%. This result is slightly surpassed by Model 7, which uses ResNet50 instead of VGG16 and achieves 97.5% accuracy. Both models demonstrate the effectiveness of using PCA with these deep learning models for binary classification tasks. LeNet models, represented by Models

8 and 9, show lower performance, with accuracies of 83% and 81%, respectively. This lower performance relative to VGG16 and ResNet50 suggests that LeNet, being a simpler model, is less capable of capturing the complex features required for high accuracy in these tasks.

Table 21: All classification results

Model_Name	Description	Model_Combination	Classes	Accuracy
Model_1	A549; 3T3	KMeans	2	12%
Model_1_1		KMeans+PCA+t-SNE		66%
Model_1_2		AgglClust+PCA+t-SNE		66%
Model_1_3		DBSCAN+PCA+t-SNE		50%
Model_1_4		SOM+PCA+t-SNE		90%
Model_2	D2; D5; D2_20; D2_50	KMeans + VGG16	4	22%
Model_3		KMeans+PCA+VGG16		67%
Model_4		**KMeans+PCA+ResNet50**		**70%**
Model_5		KMeans+PCA+ResNet101		30.8%
Model_6	20ug; 50ug	**KMeans+PCA+VGG16**	2	**98%**
Model_7		KMeans+PCA+ResNet50		97.5%
Model_8		LeNet - 10 epochs		83%
Model_9		LeNet - 100 epochs		81%
Model_10	D2; D2_5	KMeans+PCA+VGG16	2	50%
Model_10_1		KMeans+PCA+VGG16+tSNE		50%
Model_11	D2; D2_20	KMeans+PCA+VGG16	2	10.8%
Model_11_1		KMeans+PCA+VGG16		82 %

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusions

In the segmentation phase, 28 different UNet models were trained and tested for determining cell confluence, experimenting with varying hyperparameters such as loss functions, batch sizes, and epochs. Among the models tested, Model 23 and Model 28 emerged as the top performers. Model 23 employed a combined dice-binary cross-entropy loss function and showed impressive train and validation accuracies of 99.6% and 97.0%, respectively. Model 28, which was trained for 200 epochs with the same combined loss function, demonstrated a similar performance, with train and validation accuracies of 99.3% and 96.6%, respectively. Stacked Unet with two layers and U2Net also yielded good results.

The preprocessing techniques did not yield good results. Surprisingly they made the models worse. That can be said after testing the models on images with ground truths.

The analysis of cell confluence provided essential insights into the effects of various biomaterial concentrations on cell health. Two UNet models were used to evaluate cell confluence across different biomaterial densities, ranging from low to high concentrations.

Both models exhibited remarkable consistency in their predictions of cell confluence, with a difference within 2-3%. This consistency supports the reliability of the models in assessing cell growth and the impact of biomaterials.

For all three datasets, the initial confluence on Day 1, averaging around 45%, served as a baseline, with a significant increase to approximately 80% by Day 2. This rapid growth in the absence of biomaterials reflects healthy cells. At low concentrations (5 ug), both PLL250 and PAR30 were found to be non-toxic, with confluence ratios above the 70% threshold. This suggests that low-density biomaterials do not slow cell growth process, and may even support cell growth, as indicated by the slight increase in confluence on Dataset_2 measurements. Contrarily, at medium (20 ug) and high concentrations (50 ug for PAR30 and 500 ug for PLL250), there was a marked reduction in cell confluence. The ratios for medium density dropped below the 70% toxicity threshold, signaling significant inhibitory effects on cell growth. At the highest densities, the confluence ratios decreased to around 0.35, indicating severe toxicity, likely leading to extensive cell death.

The classification phase involved analyzing various models on datasets with different cell images with different biomaterial densities. Principal Component Analysis (PCA) and t-SNE consistently enhanced the classification accuracy across models. Models combining KMeans clustering with PCA and/or t-SNE, such as those utilizing VGG16 and ResNet50 architectures, achieved significantly higher accuracy compared to models without. This indicates that these techniques effectively reduce dimensionality, facilitating better feature extraction and classification performance.

The analysis revealed that increased model complexity does not always lead to improved performance. For example, the more complex ResNet101 model performed worse than the simpler ResNet50, highlighting issues of overfitting and the need for balanced model complexity.

Binary classification tasks generally yielded higher accuracies compared to multi-class tasks. Models distinguishing between two distinct cell conditions achieved accuracies nearing 98%, while multi-class models struggled. The highest accuracy in multi-class models (dividing in 4 classes) was 70%. This suggests that binary classification is more straightforward, likely due to the clearer distinction between two conditions compared to multiple, potentially overlapping categories.

5.2 Use in Medicine

Significant practical benefits for doctors can be offered by the findings and developed models from this research. It is time saving for doctors and laboratory technicians to use automated cell confluence analysis tools, compared to manual counting and assessment. Automatization allows for more focus on diagnosis and treatment planning. The high consistency and accuracy in predicting cell confluence that are shown by these models, reduce variability and potential human error, ensuring reliable results. Toxic effects on cells can be quickly identified by evaluating cell growth at different biomaterial concentrations, which is crucial for early detection of adverse reactions in drug testing and treatment evaluations.

The appropriate biomaterials for treatments can be selected by doctors, by understanding the impact of various biomaterial concentrations on cell health. Safe and effective dosages for biomaterials and drugs can be determined by insights into toxicity thresholds, minimizing side effects and improving patient outcomes. Disease diagnosis based on cell morphology and conditions can be supported by accurate classification of different cell conditions, enhanced with PCA and t-SNE, contributing to more personalized and effective medical care.

Treatment progress can be tracked by monitoring cell confluence over time, allowing necessary adjustments to ensure optimal cell growth. New treatments can be assessed by comparing cell confluence and health before and after the treatment using these models. Personalized treatments tailored to individual patients based on their specific cellular conditions can be provided by doctors.

5.3 Future Work

One of the critical factors affecting the results of this study is the quality and consistency of manual annotations. It is highly recommended to re-annotate the images by involving different researchers. This will introduce diversity in the annotations and help reduce individual bias. With a diverse set of annotations, retrain the UNet models to investigate how differences in ground truth affect model performance. This will help understand the sensitivity of the models to annotation and identify which models perform better to such differences. In the end, it is suggested to conduct an analysis comparing annotations from multiple researchers and to assess the new UNet models. This can provide insights into the consistency and reliability of the annotations, and highlight areas where further training or standardization might be necessary.

One other suggestion for future work would be to try different cropping sizes, such as 512x512. This way the images will contain more information.

Given that preprocessing techniques did not yield the expected results in the current study, it is suggested to explore and experiment with various advanced preprocessing methods. Techniques such as image normalization, histogram equalization, Gaussian Filter, and Median Filter could be re-evaluated and potentially combined in different ways to improve the input quality for model training.

Retrain the UNet models with the newly annotated datasets and different preprocessing methods. This will allow for a direct comparison and identification of the most effective combination of annotation practices and preprocessing methods.

REFERENCES

[1] Gonzalez, R. C., & Woods, R. E. (2018). *Digital Image Processing*.

[2] Kaul, V., Enslin, S., & Gross, S. A. (2020). History of artificial intelligence in medicine. *Gastrointestinal Endoscopy, 92*(4), 807-812.

[3] Xing, F., Xie, Y., Su, H., Liu, F., & Yang, L. (2017). *Deep Learning in Microscopy Image Analysis: A Survey. IEEE Transactions on Neural Networks and Learning Systems, 1–19.* doi:10.1109/tnnls.2017.2766168

[4] Helgadottir, S., Midtvedt, B., Pineda, J., Sabirsh, A., Adiels, C. B., Romeo, S., Midtvedt, D., & Volpe, G. (2021). Extracting quantitative biological information from bright-field cell images using deep learning. *Biophysics Rev*, *2*(3), 031401. https://doi.org/10.1063/5.0044782

[5] Adnan, M. M., Rahim, M. S. M., Al-Jawaheri, K., Ali, M. H., Waheed, S. R., & Radie, A. H. (2020). A Survey and Analysis on Image Annotation. In 2020 3rd International Conference on Engineering Technology and its Applications (IICETA) (pp. 203-208). Najaf, Iraq. doi:10.1109/IICETA50496.2020.9318911.

[6] P. F. T. B. Olaf Ronneberger, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in MICCAI 2015: Medical Image Computing and Computer-Assisted Intervention, 2015

[7] Trizna, E. Y., Sinitca, A. M., Lyanova, A. I., et al. (2023). Brightfield vs Fluorescent Staining Dataset–A Test Bed Image Set for Machine Learning based Virtual Staining. Scientific Data, 10, 160.

[8] Polisi, X., Halili, A. N., Uka, A., & Ciulla, C. (2023, August). Two-Stage Unsupervised Classification of Cell Health. In *2023 International Conference on Computing, Electronics & Communications Engineering (iCCECE)* (pp. 145-149). IEEE.

[9] Polisi, X., Avdiu, D., Uka, A., Halili, A. N., Kollcaku, K., & Ciulla, C. (2023, August). Evaluation of Cell Segmentation Using Pruning and Quantization. In *2023 International Conference on Computing, Electronics & Communications Engineering (iCCECE)* (pp. 139-144). IEEE.

[10] 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.

[11] Caicedo, J. C., Goodman, A., Karhohs, K. W., Cimini, B. A., Ackerman, J., Haghighi, M., ... & Carpenter, A. E. (2019). Nucleus segmentation across imaging experiments: the 2018 Data Science Bowl. *Nature methods*, *16*(12), 1247-1253.

[12] Stringer, C., Wang, T., Michaelos, M., & Pachitariu, M. (2021). Cellpose: a generalist algorithm for cellular segmentation. *Nature methods*, *18*(1), 100-106.

[13] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., ... & Girshick, R. (2023). Segment anything. *arXiv preprint arXiv:2304.02643*.

[14] A Uka, X. Polisi, J. Barthes, A. N. Halili, F. Skuka and N. E. Vrana, "Effect of Preprocessing on Performance of Neural Networks for Microscopy Image Classification," *2020 International Conference on Computing, Electronics & Communications Engineering (iCCECE)*, Southend, UK, 2020, pp. 162-165, doi: 10.1109/iCCECE49321.2020.9231071.

[15] A. Uka, A. Tare, X. Polisi and I. Panci, "FASTER R-CNN for cell counting in low contrast microscopic images," *2020 International Conference on Computing, Networking, Telecommunications & Engineering Sciences Applications (CoNTESA)*, Tirana, Albania, 2020, pp. 64-69, doi: 10.1109/CoNTESA50436.2020.9302852.

[16] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. https://arxiv.org/abs/1505.04597

[17] 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)*. https://arxiv.org/abs/1606.06650

[18] Qin, X., Zhang, Z., Huang, C., Dehghan, M., Zaiane, O. R., & Jagersand, M. (2020). U^2-Net: Going deeper with nested U-structure for salient object detection. *Pattern Recognition*. https://arxiv.org/abs/2005.09007

[19] Ching, T., Himmelstein, D. S., Beaulieu-Jones, B. K., Kalinin, A. A., Do, B. T., Way, G. P., ... & Greene, C. S. (2018). Opportunities and obstacles for deep learning

in biology and medicine. Journal of the Royal Society Interface, 15(141), 20170387. https://doi.org/10.1098/rsif.2017.0387

[20] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324. https://doi.org/10.1109/5.726791

[21] Lu, H., Chen, J., & Fu, K. (2021). Improved LeNet architecture for cancerous and non-cancerous cell classification. IEEE Access, 9, 19500-19509. https://doi.org/10.1109/ACCESS.2021.3054321

[22] Shao, S., Wang, H., & Wei, Y. (2018). Application of deep convolutional neural network LeNet in the classification of cell images. Journal of Medical Imaging and Health Informatics, 8(6), 1242-1247. https://doi.org/10.1166/jmihi.2018.2444

[23] Wang, Y., Zhang, Y., & Li, X. (2022). Integrating deep learning and K-means clustering for cell type classification. Computers in Biology and Medicine, 143, 105288. https://doi.org/10.1016/j.compbiomed.2022.105288

[24] Wei, Q., Zhang, H., & Yang, Y. (2019). Classification of live and dead cells in fluorescence microscopy images using LeNet architecture. Bioinformatics, 35(14), 2445-2451. https://doi.org/10.1093/bioinformatics/btz094

[24] Li, Y., Zhang, L., Wang, X., & Liu, Q. (2022). K-means clustering for classifying cell development stages in time-lapse microscopy. IEEE Transactions on Biomedical Engineering, 69(1), 234-243. https://doi.org/10.1109/TBME.2021.3081002

[25] Xie, Y., Jiang, H., & Zhang, M. (2021). Hybrid CNN and K-means clustering for classification of cancerous and non-cancerous cells in microscopy images. Computers in Biology and Medicine, 134, 104514. https://doi.org/10.1016/j.compbiomed.2021.104514

[26] Yang, Z., Zhou, X., & Li, H. (2021). K-means clustering for segmentation and classification of single-cell images in complex tissues. Journal of Biomedical Informatics, 123, 103931. https://doi.org/10.1016/j.jbi.2021.103931

[27] Yao, Y., Zhao, X., & Chen, L. (2019). Unsupervised classification of cell types in histopathological images using K-means clustering. Bioinformatics, 35(14), 2365-2373. https://doi.org/10.1093/bioinformatics/btz118

[28] Zhang, H., Wei, Q., & Ma, L. (2020). K-means clustering for the classification of live and dead cells in fluorescence microscopy. IEEE Transactions on Medical Imaging, 39(5), 1238-1247. https://doi.org/10.1109/TMI.2019.2956259