Ben-Gurion University of the Negev

Faculty of Engineering Science

School of Electrical and Computer Engineering

Dept. of Electrical and Computer Engineering

Fourth Year Engineering Project

Final Report

Microscopy Imaging Analysis via Deep Learning

Project number:	**p-2024-125**	
Students	**Nadav Levy**	**316409531**
(name & ID):	**Eran deutsch**	**209191063**
Supervisors:	**Tammy Riklin Raviv**	
Sponsors:		
Submitting date:	**18.8.24**	

Table Of Contents

Research project -Microscopy Imaging Analysis via Deep Learning

Student Names:Eran Deutsch, Nadav Levy.
Email: eran1898@gmail.com, nadklayn@gmail.com
Adviser's name: Tammy Riklin Raviv.

1. Project completion summary

1.1 Background

The project addresses the problem of Segmenting Microscopy Cell images in order to improve and enhance biological research capabilities.

1.2 Purpose

Improve the ability of Machine learning models to segment and detect cells in microscopy images.

1.3 Objectives:

To train a Machine Learning Model to segment cells out of a biological microscopy image.

1.4 Innovation:

Creating a U-net capable of classifying the image into 3 different classes using input preprocessing image processing methods.

1.5 Proposed Method:

Our method involves building a U-net [-] architecture model and augmenting the data given to us using many different types of augmentations [] to create a more diverse dataset. also optimizing the model and the hyperparameters in order to achieve the best results of the specific data set.

1.6 Expected Results:

The expectation is that our newly trained model will improve upon the old models currently existing and be able to segment and track cells more accurately.

1.7 Keywords:
instance segmentation, machine learning, U-net, image processing, edge detection,CNN

2. Project Completion Summary (Hebrew)

רקע:

הפרויקט מתייחס לבעיית הפרדת תאים מרקע בתאים ביולוגיים על מנת לשפר ולייעל תהליכי מחקר ביולוגיים.

מטרה:

המטרה בפרויקט היא לשפר את יכולתם של מודלים לומדים להפריד ולספור תאים בהינתן תמונות מיקרוסקופיות של תאים ביולוגיים.

יעדים:

היעד שלנו בפרויקט הוא לאמן מודל שיכול לשפר את התוצאות הקיימות בהפרדת וספירת תאים בתמונות.

חידוש

יצירת רשת מסוג "U-net" שיכולה לאפיין בין שלושה סוגי תוויות לכל פיקסל בתמונה, ושימוש בטכניקות עיבוד מידע מקדים הקשורות לעיבוד תמונה.

השיטה המוצעת:

השיטה שלנו היא בניית רשת מארכיטקטורת "U-net" ושימוש בהרבה סוגי אוגמנטציות שונות על מנת להרחיב את המידע שלנו. בנוסף מטרתינו היא להגיע לתוצאות מיטביות על המידע הנתון על ידי שיפור הפרמטרים הלא נלמדים של המודל.

תוצאות צפויות:

הצפי שלנו היא שהמודל שלנו יעקוף את התוצאות המצויות כרגע בתחום הסגמנטציה על סוג הדאטה הזה.

מילות מפתח:

סגמנטציה פרט, למידת מכונה, ארכיטקטורת "Unet", עיבוד תמונה, זיהוי קצוות, רשתות קונבולוציה.

3. - Project Goals

3.1 - Main Goal

The objective of our project is to enhance the performance of existing models in the field of instance segmentation of cells. Our goal is to develop a model that can more accurately identify, and quantify cells within an image automatically, thereby eliminating the necessity for manual intervention.

3.2 - Measure

Our measure of success is the SEG measure as proposed by [1] which is used for the Evaluation of the Cell Segmentation Benchmark [2]. The SEG measure is a variant of the IoU measure for multiple instances of the same object in an image.

3.3 - Secondary Goal

Our second goal is to win in the Cell Segmentation Benchmark [2] and get a better result than the current leaders.

3.4 - Measure

Our measure of success is the results of the Cell Segmentation Benchmark [2].

3.5 Side Goal :

An additional objective of our project is to analyze various hyper-parameters and data augmentation techniques to identify those that yield the best performance.

3.6 Measure:

Our measure of success is to develop a model that outperforms those that have not undergone hyper-parameter optimization.

4. Introduction

One of the significant challenges in the field of biology is the analysis of medical images. The main difficulty arises from the substantial human effort needed to interpret these images, as well as the complexity involved in distinguishing and analyzing them.

Automating cell Tracking and segmentation helps reduce errors and speeds up data analysis in biological research.
This automation is especially important in cancer research, as it allows for faster and more accurate analysis of cell behavior, leading to earlier detection and more personalized treatment.

Semantic segmentation is a computer vision process that classifies each pixel in an image into predefined categories, dividing the image into regions corresponding to different objects or features.

Automated systems not only reduce the workload for researchers but also ensure that analysis results are more reliable and consistent. This, in turn, enables more precise conclusions, supports larger-scale experiments, and improves medical decision-making processes.

The goal of this project is to perform segmentation tasks effectively, integrate innovations, and conduct analyses that will support future research and provide insights into various structures and datasets. The project is evaluated using a precision metric called SEG, which focuses on the accuracy around cells presented in the images and imposes significant penalties when unnecessary cell mergers are predicted.

The project focuses on 2D image Dataset of Cells movement. previous projects achieved 83.2% accuracy with the SEG metric, using UNET architecture, combining creativity augmentation, and more.

To overcome this achievement, we decided to take advantage of recent studies and add some innovations of our own. The keypoint of our enhancements was delivering more information to the chosen model to allow it to learn better, punish the classes differently and try to separate the objects in the image as precisely as possible.

5. - Final Technical Specifications

5.1 - The Data

The dataset utilized in our work is the Fluo-N2DH-SIM+ dataset [4], which comprises simulated nuclei of HL60 cells stained with Hoechst, generated using MitoGen, a component of Cytopacq. This dataset includes three videos of simulated cells: two of these come with manually created golden ground truth segmentation masks, which we employed for training and validation, while the third is reserved for testing the model's performance.

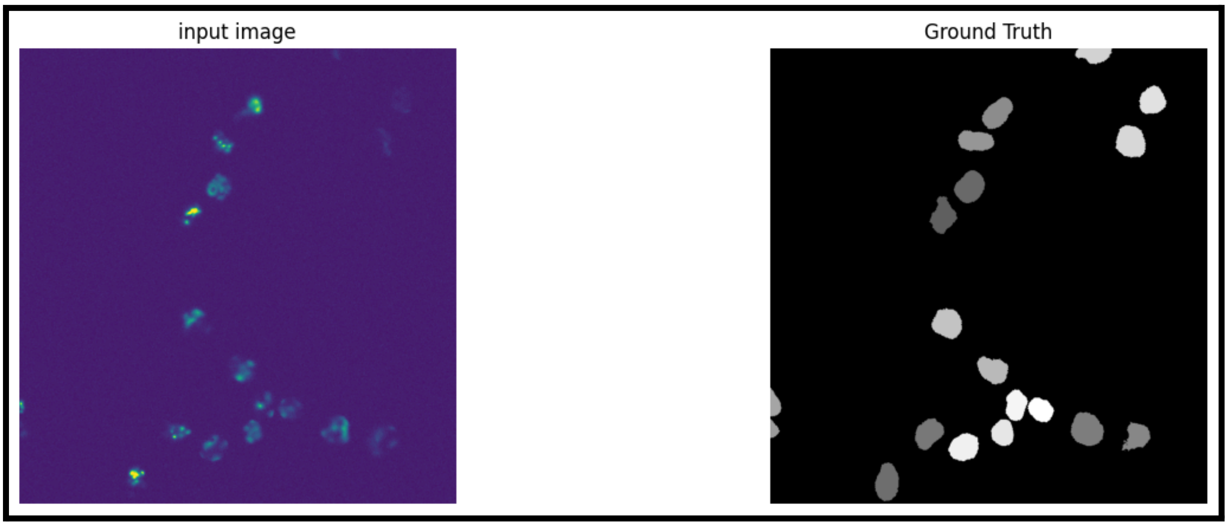

Fig.1
Input cell image (left) and the ground truth reference image (right).

5.2 - Model Technical Specifications

5.2.1 - Input Pre-Processing

In this section, we sought to enhance the information provided to our model by augmenting and enriching the dataset through advanced data preprocessing techniques prior to model input. This strategy was particularly valuable as it introduced a novel approach that demonstrated improved performance in our project and has the potential to yield superior results in other segmentation-related tasks.

5.2.2 - Augmentation

This approach allows us to increase the number of available images by applying geometric transformations, thereby helping the model to better generalize to unseen data.

5.2.3 - Canny and Invers

In this instance, we sought to enhance the model's input by incorporating both the image inverse and its Canny edge detection results, which were fed into the model together with the original image.

5.2.4 - Edges Output Mask

Here, we utilized the provided ground truth data alongside an edge detection algorithm to accurately identify the cell boundaries.

5.3 - U-Net Architecture

5.3.1 - U-net Selection

We selected the U-Net architecture [3] for image instance segmentation due to its ability to preserve spatial context and combine high-resolution features with contextual information, essential for accurately segmenting complex biological structures like cells.

5.3.2 - 3 Classes Classifier

In our architecture, we utilize a classifier with three output classes: cells, background, and cell edges. This setup is designed to enhance the model's capability to distinguish and segment closely spaced cells individually. Additionally, we apply different weights to each class to address the imbalance, giving higher importance to the cell edges, which are less represented but critical for accurate segmentation.

5.4 - SEG Measure

To evaluate our model and its results, we employed the SEG measure that calculates the Intersection over Union (IoU) for each cell by comparing the model's outputs with the golden reference. This measure then averages the IoU scores to produce a final score ranging from 0 to 1 [1].

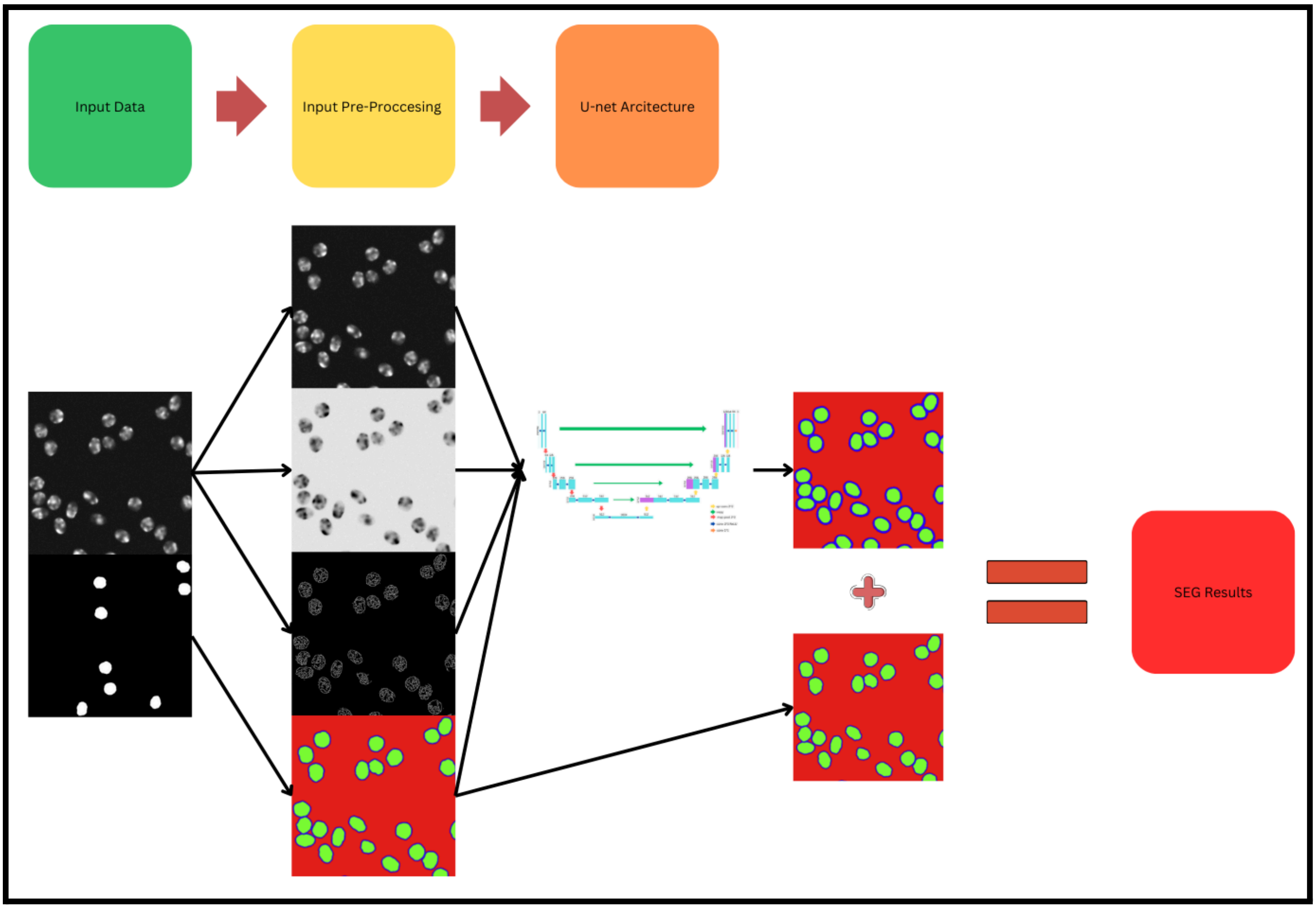

Fig.2.

Technical specification flow chart.

6. - The Approach Taken To The Solution And The Engineering Design:

6.1 - Input Preprocessing

6.1.1 - Input Enhancement

The purpose of input Enhancement is to implement reliable data to help the model learn the patterns and the mission itself. it's done by duplicating the input image into 3 images -
1 - Original image - The unaltered image as initially captured.
2 - Inverse image - Inverse all pixels values - An image where all pixel values have been inverted, highlighting contrast and emphasizing features that might otherwise be subtle in the original.
3 - Canny edge image - An image that focuses on the edges of the objects within the original input, generated using the Canny edge detection algorithm.

This strategy is critical as it allows the model to learn from a richer set of data representations, offering diverse perspectives on the same image. By presenting the model with these three variations, it gains the ability to discern and learn from different aspects of the image, thereby enhancing its predictive capabilities without requiring additional raw data.

We consider this enhancement to be both innovative and pivotal to the success of our model. Not only does it introduce a novel way to augment the dataset, but it also significantly improves the model's ability to generalize from the available data. We anticipate that this addition will have a substantial impact, contributing to more accurate and reliable predictions in our project and potentially benefiting other segmentation tasks in future research.

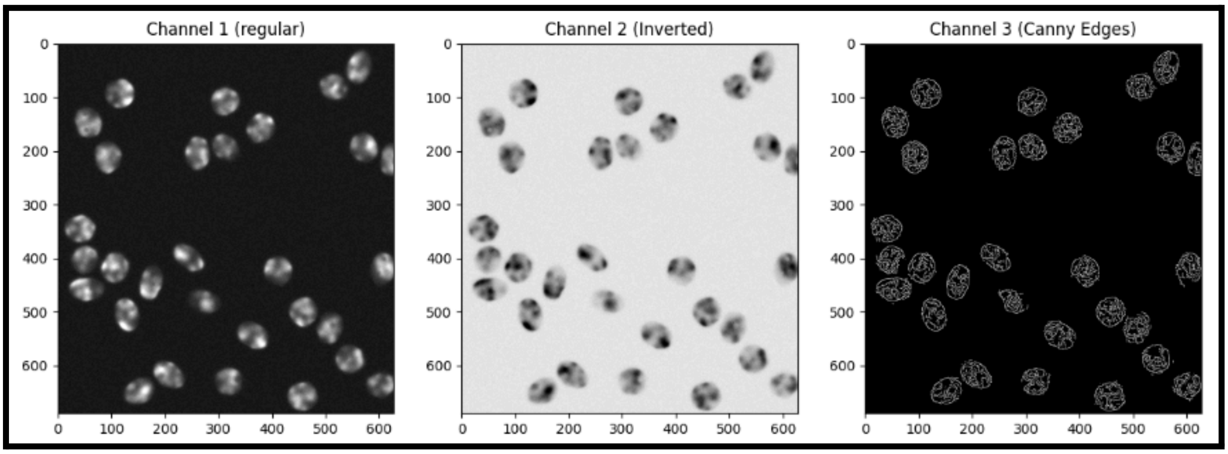

fig. 3.

The input converted into 3 images - the first (left) is the original, second (center) is the inverted image, the last (right) is the Canny Edge detection on the image

6.1.2 - Data Augmentation

To get a well-Trained model the dataset should be enormous, so the model could learn from all kinds of Cell sizes, types, amounts, shapes and more. when dealing with medical images the amount of dataset - images is limited. Moreover, blurring and distorting the image could challenge the model in training and lead to better results.
All augmentation was picked randomly, so the data would be the most diverse.
For that, we have implemented 3 different augmentations:

6.1.2.1 - Stretch and Crop

Linear transformation algorithm which stretches and constricts the input. To be consistent and to create data diversity we crop the stretched input to [256,256,3] shape.

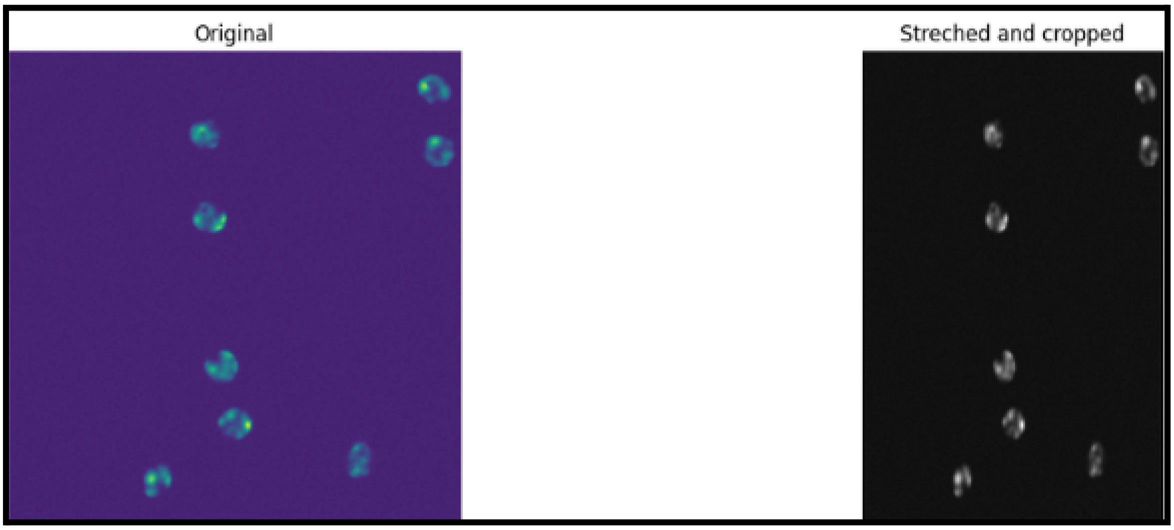

fig. 4.
The original image (left) and the stretched image (right)

The second we did was cropping the image.
To increase the data we chose a random index which pointed to the top left of the image and then cropped it to size 256,256.

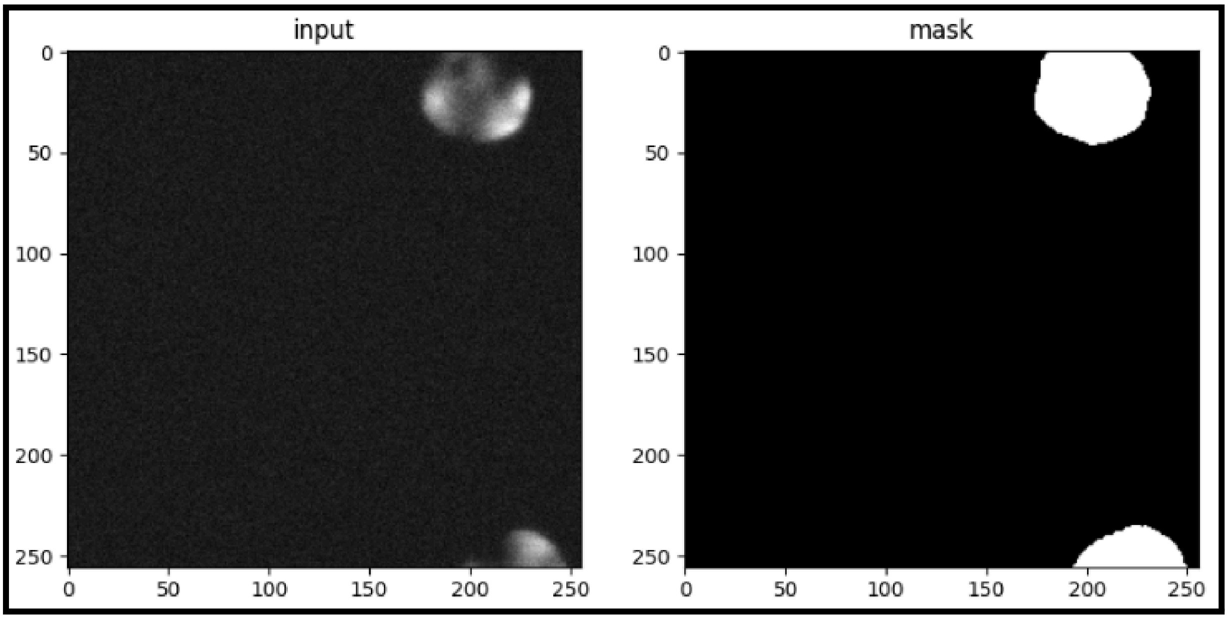

Fig. 5.
The cropped input (left) and the cropped mask (right).

6.1.2.2 - Vertical Flip and Horizontal Flip

This method increases the diversity of the data and creates "new" images by Flipping \ mirroring or both.

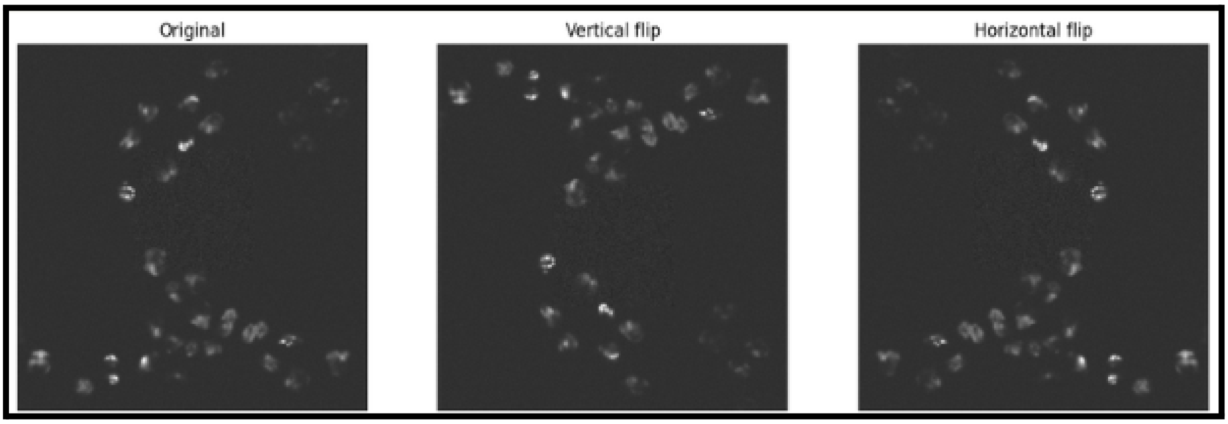

Fig. 6.
A given image (left), after vertical flip (center) and after horizontal flip (right).

6.1.2.3 - Adding Gaussian Noise

This method is designed to challenge the model during learning, thereby improving its performance. The noise is added to the first two layers (original and inverse) but not to the Canny layer. To determine the appropriate variance for learning, we attempted to analyze the difference in values between a cell pixel and a background pixel. We discovered that there is no consistent pattern and that the difference varies per image. Therefore, we took the average of the differences, which is sigma = 0.005.

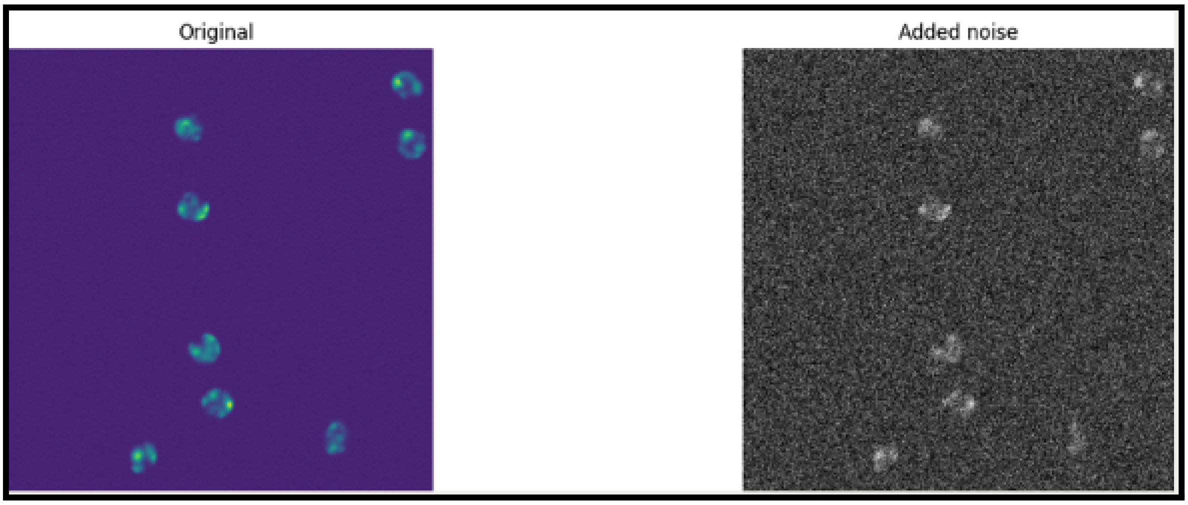

fig. 7.
The original input (left) and the noisy input (right)

6.2 - U-net Architecture [3]

We chose the U-Net architecture for the image instance segmentation task because it effectively preserves spatial context while integrating high-resolution features from the encoder with contextual information from the decoder. This architecture is particularly well-suited for accurately segmenting complex biological structures, such as cells, where the images may contain noise or overlapping elements. The U-Net's design allows it to capture both fine details and broader contextual cues, which are crucial for distinguishing between closely situated or similarly appearing structures in challenging biological imagery.

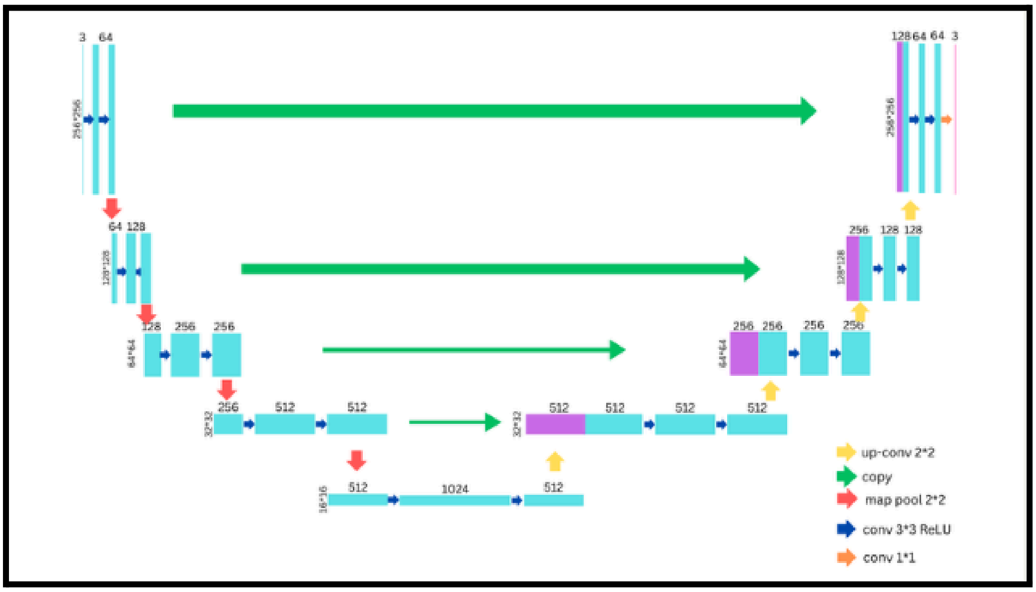

Fig. 8.

U-net architecture - combining skip connection, convolutional layers, deconvolutional layers pooling and more.

6.3 - 3 Classes Classifier.

We added a 3-rd class which determines the edge of cells.

The addition of an extra class was implemented to enhance segmentation and improve the identification of individual cells within the image. Introducing an edge-type class can help us better distinguish and create a more effective boundary between closely situated cells.

To create the third class, we utilized an image processing algorithm called Canny edge detection[]. This algorithm is widely recognized for its ability to detect a wide range of edges in images by applying a multi-stage process, which includes noise reduction, gradient calculation, non-maximum suppression, and edge tracing by hysteresis.

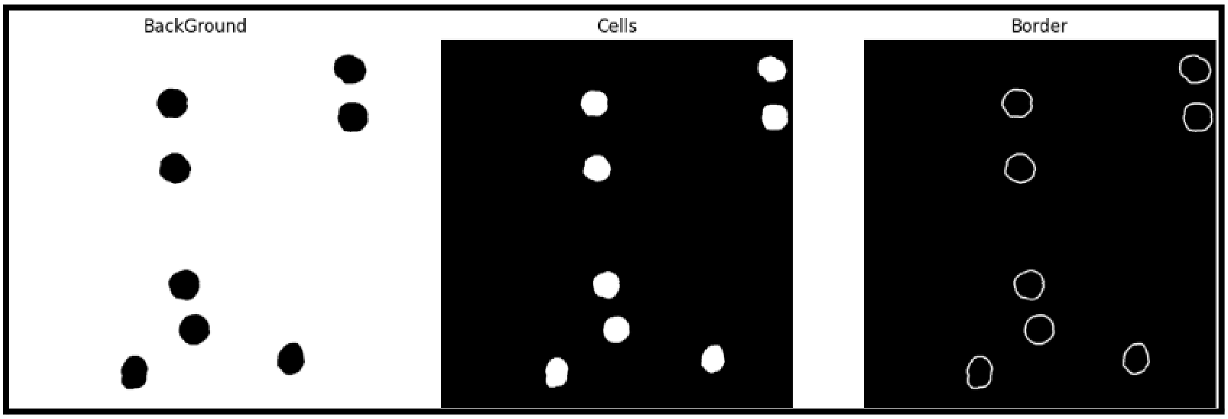

Fig. 9.

Background (left), Cells (center) and Edges of the cells (right)

The combination of the three image channels from the three different classes results in a composite image where the background is represented in red, the cells in green, and the cell edges in blue.

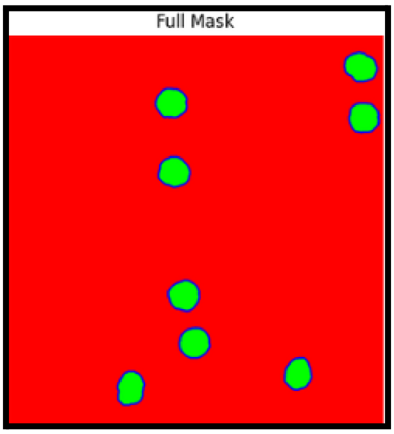

Fig. 10.

The full mask (reference)

We estimate that the added class will help reduce cell merging and improve SEG results.

6.4 - Weighted Cross-Entropy Loss

In classification tasks, underrepresented classes can lead to biased model performance, where the model favors more frequent classes. To mitigate this, we propose modifying the loss function by using a Weighted Cross-Entropy Loss. This approach assigns higher weights to underrepresented classes, encouraging the model to focus on them.

$$\boxed{\ell(x, y) = L = \{l_1, \ldots, l_N\}^\top, \quad l_n = -\sum_{c=1}^{C} w_c \log \frac{\exp(x_{n,c})}{\sum_{i=1}^{C} \exp(x_{n,i})} y_{n,c}}$$

Fig. 11.

Weighted Cross Entropy Loss equation, where w_c represents the weight for class c.

This method enhances the model's ability to classify underrepresented classes accurately.

7. - Final Acceptance Tests for the Product

7.1 - Evaluating Measures

In order to test our model performance we used two main measures. the Accuracy measure and the SEG measure. Both of them were used by us for different aspects of the training and development of our final model.

7.1.1 - Accuracy Measure

This measure evaluates how accurately our algorithm classifies each pixel in the image. Accuracy is determined by comparing the predicted class of each pixel (e.g., cell, background, or cell edge) with the actual class from the ground truth. Specifically, it is calculated as the ratio of correctly classified pixels to the total number of pixels in the image. This measure provides an overall assessment of how well the model is performing across the entire image. High accuracy indicates that the model is correctly identifying the majority of pixels, but it's important to consider that accuracy alone may not fully capture the model's effectiveness, especially in cases with class imbalances or when small but critical regions (like cell edges) are involved.

7.1.2 - SEG metric

To address these challenges, we also employed the SEG [1] measure to evaluate our model. The SEG measure is calculated using the Intersection over Union (IoU) for each individual cell in the image, with the IoU included in the calculation only if the detected cell covers at least half of the reference cell. To compute this metric, we first aggregated all the pixels for each connected component and compared them to the corresponding components in the ground truth. This process yielded an IoU score for each cell, which was then averaged across the entire image and subsequently across the entire validation set.

This metric imposes a significant penalty when cells are incorrectly merged, as the IoU will be substantially lower when one of the cells is doubled (due to merging). This characteristic provides an incentive for including the third, edge class, as it helps prevent the model from combining adjacent cells, thereby leading to more accurate segmentation.

When calculating the SEG measure, we treated the cell edge class as background, excluding it from the IoU calculation, as the primary purpose of the edge class is to aid in distinguishing between adjacent cells. This approach ensures that the SEG measure more accurately reflects the model's effectiveness in detecting and segmenting cells, while placing less emphasis on background classification errors.

7.2 - Results

7.2.1 - Hyperparametes analysis

One of the main goals in our project was to analyze the model and its parameters and try to achieve better results. We analyzed different parameters of the model and got some results on the best parameters to choose for our network.

learning rate results :

We found that as we decrease the learning rate our model gets less noisy and more robust.we found that lr = 0.0002 was the threshold for a good learning curve.

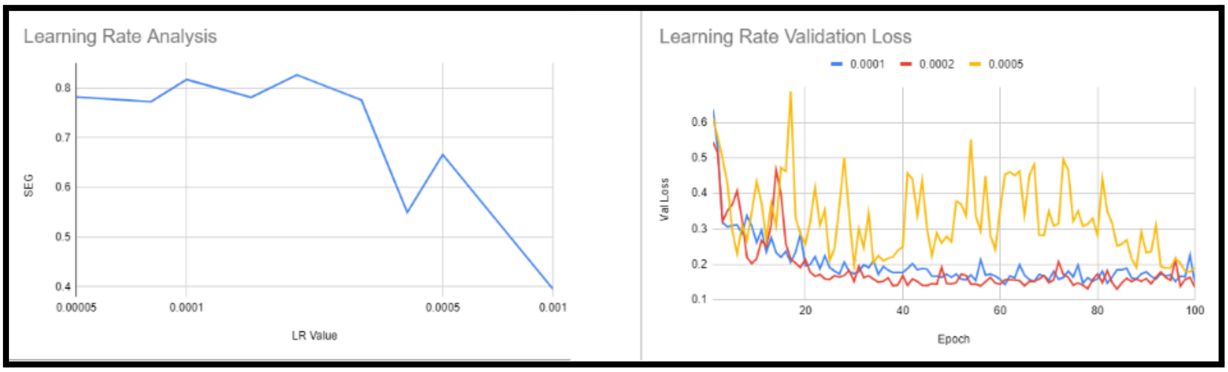

Fig.12.

SEG via Learning rate (left) , Val Loss via epoch (right)

Another hyperparameter that we tested was the relation of batch size in comparison with training image size. We found that the best performance was with batch size = 8 and training input of 128*128 image size. The small image size was probably successful due to more information in relation to background in the smaller images (as in our algorithm we make sure that at least a single cell is in the image).

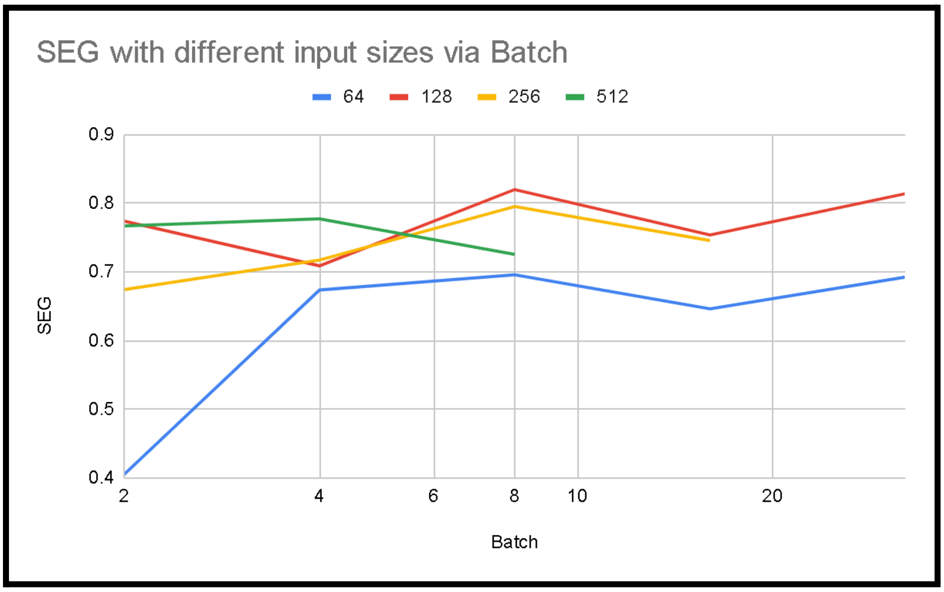

Fig. 13.

SEG via batch size for every input crop size.

7.2.2 - Model Results

The enhancements we made to the U-Net architecture yielded significant improvements in the automated segmentation of cells in microscopy images. Our modified U-Net model, which incorporated three output classes (cells, background, and cell edges), demonstrated a marked increase in its ability to accurately distinguish between individual cells, particularly in densely packed or overlapping scenarios.

The model's performance was evaluated using the SEG measure, which utilizes the Intersection over Union (IoU) for each detected cell. The inclusion of the edge class proved crucial in preventing the common issue of cell merging, where adjacent cells are incorrectly combined into a single entity. By assigning different weights to each class in the loss function, we emphasized the importance of correctly identifying cell boundaries, which are underrepresented in our data.

Furthermore, the utilization of three distinct input channels (original images, their inverses, and Canny edge-detected versions) provided the model with a more comprehensive set of features, thereby enhancing its ability to differentiate between closely situated cells. This multi-channel approach proved particularly advantageous in handling complex images, where conventional single-channel inputs might struggle to accurately distinguish between overlapping or adjacent structures. By incorporating these diverse representations, the model was able to better capture and interpret the subtle nuances within the data, ultimately leading to more precise segmentation outcomes.

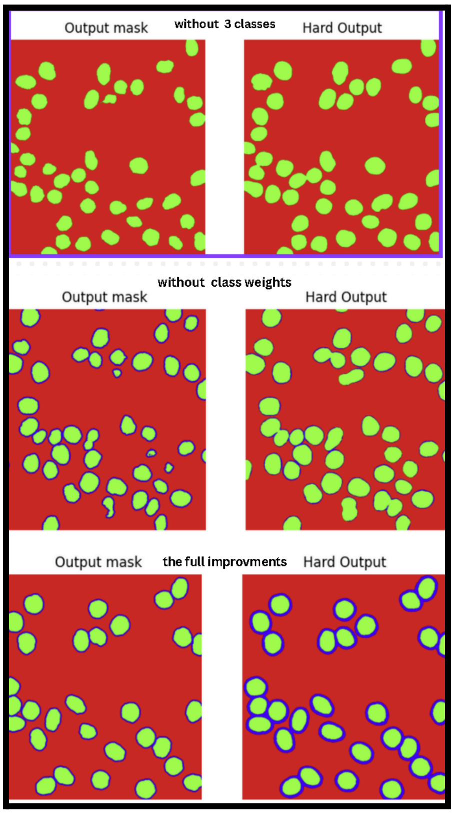

Fig. 14.

A visual comparison of segmentation outputs, highlighting the difference between 2-classes (top), 3-classes without weighted loss (center) and the full improvement (bottom).

Across the validation set, the model consistently achieved higher SEG scores, indicating that it was successful in accurately segmenting cells. The SEG measure was calculated by aggregating the IoU scores of each cell.

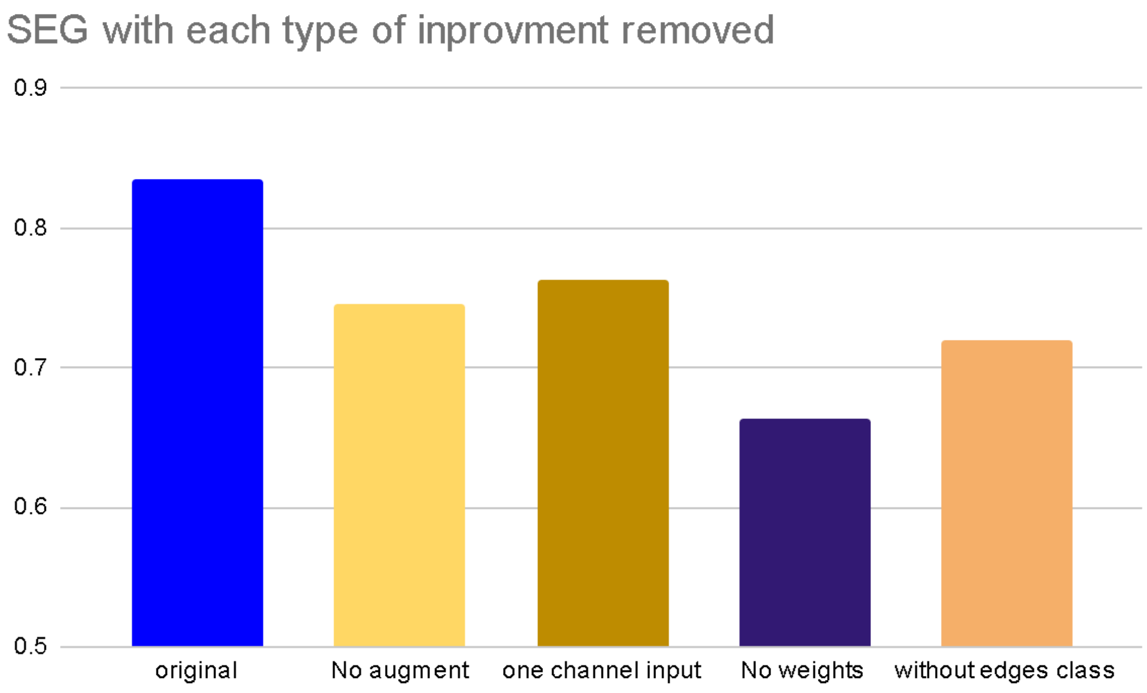

Fig. 15.

A graphical representation of the SEG scores across different test cases, showing the improvement in SEG with the modified U-Net model

8. - Challenges and Solutions

8.1.1 - Problem - Determining the Optimal Hyperparameters

To achieve the ideal model that produces the best possible results, it was necessary to identify the correct hyperparameters that would lead to optimal outcomes. The primary challenge was the interdependence between hyperparameters—some hyperparameters influence others and vice versa, making it extremely difficult to find the ideal configuration.

8.1.2 - Solution

A thorough analysis was conducted, based on images, relevant articles and studies, model learning curve graphs, and SEG results. This comprehensive approach allowed us to systematically identify the optimal hyperparameters, ensuring the best possible performance of the model.

8.2.1 - Problem - Merging of multiple cells

Due to the inherent complexities in the data, which include cell movements, deformations, splitting, merging, and other dynamic behaviors, it was challenging to create a sufficiently reliable separation that would allow the model to distinguish between two cells.

8.2.2 - Solution

To address this challenge, we integrated a third class specifically for cell boundaries. This addition significantly improved the model's ability to create distinct separations, enhancing its effectiveness in distinguishing between adjacent cells, even in cases where cells are closely positioned or partially merged.

8.3.1 Problem - Under representation of the 3-rd class (edge)

With the addition of the third class, the model's learning and identification of this class from the reference images were inaccurate. This inaccuracy was primarily due to the limited number of pixels representing the edge class, which were located precisely at the boundary between the background and the cell. As a result, these pixels were often overwhelmed and not adequately captured by the model.

8.3.2 - Solution

We addressed this issue by combining the use of Weighted Cross-Entropy Loss to create a stronger separation and clearer boundaries. This approach penalizes boundary misclassifications more heavily than those involving the background or cell regions.

9. - Conclusions and recommendations

9.1 - Conclusions

In this project, we successfully enhanced the U-Net architecture for image instance segmentation, particularly in the context of microscopy images. Our approach, which included the integration of a third class for cell boundaries and the application of Weighted Cross-Entropy Loss, resulted in significant improvements in accurately distinguishing between individual cells, even in densely packed or overlapping scenarios.

Advantages:

- Spatial Context: The model preserves spatial context, crucial for segmenting complex structures in medical images.
- Feature Set: Multi-channel input enhanced the model's ability to differentiate cells.
- Weighted Loss: Emphasizing underrepresented classes led to more precise segmentation.
- Extended Input: The use of extended input (original, inverted, and edge-detected images) helped the model learn and improve its performance significantly.

Disadvantages:

- Larger Objects: The model struggled with segmenting larger objects.
- Weight Analysis: We could not fully analyze the weight distribution in the loss function.
- Data Reliability: The initial dataset's reliability was insufficient, affecting result accuracy

9.2 - Recommendation For Future Research

We got a few insights for the future,

1. Conduct a thorough analysis of the weight distribution in the loss function to gain insights into optimizing the model's performance.
2. Dividing the output into 4 classes instead of 3 to get a class of edges that are more important (background,cell,edges between cell and background, edges between two or more cells).
3. Adding extra trainable models to evaluate more inputs to the model and give the model more data, for example number of cells in an image, edges, a rudimentary cell mask.

9. References

[1] -Real, Raimundo, and Juan M. Vargas. "The probabilistic basis of Jaccard's index of similarity." Systematic biology 45.3 (1996): 380-385

[2] - M. Maška, V. Ulman, D. Svoboda, P. Matula, P. Matula, C. Ederra, A. Urbiola, T. España, S. Venkatesan, D. M. W. Balak, P. Karas, T. Bolcková, M. Štreitová, C. Carthel, S. Coraluppi, N. Harder, K. Rohr, K. E. G. Magnusson, J. Jaldén, H. M. Blau, O. Dzyubachyk, P. Křížek, G. M. Hagen, D. Pastor-Escuredo, D. Jimenez-Carretero, M. J. Ledesma-Carbayo, A. Muñoz-Barrutia, E. Meijering, M. Kozubek, C. Ortiz-de-Solorzano. A benchmark for comparison of cell tracking algorithms. Bioinformatics, 2014.

[3] - Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," 2015.

[4] - Simulated nuclei of HL60 cells stained with Hoescht
Dr. V. Ulman and Dr. D. Svoboda. Centre for Biomedical Image Analysis (CBIA), Masaryk University, Brno, Czech Republic

[5] - A. Arbelle and T. Riklin Raviv, "Microscopy cell segmentation via convolutional lstm networks," pp. 10081012, 2019.

[6] - A. Arbelle and T. Riklin Raviv, "Microscopy cell segmentation via adversarial neural networks," IEEE ISBI, pp. 645–648, 2018.

[7] - S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in International conference on machine learning, 2015, pp. 448–456.

[8] - Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," 2017.

[9] - K. Musgrave, S. Belongie, and S.-N. Lim, "PyTorch metric learning," 2020.

10. Appendix

"Final_code.zip" - containing all the code files

"המלצת ציון לדוח מסכם"