“Is This Cell Alive?” U-Net Segmentation for LiveCell Imaging

İsmail Gürler
03779855
Management & Technology
TU Munich
ismail.guerler@tum.de

***Abstract*— The field of live cell imaging and the complex challenge of automating cell segmentation using deep learning techniques is the main subject of this project. Live cell imaging has revolutionized the field of cell biology by enabling real-time observation of dynamic cellular processes. However, manual analysis of these images is labor-intensive and prone to subjectivity. This project was set up to implement and evaluate the U-Net architecture as a solution for automatic segmentation of live cell images.**

Keywords—Live cell imaging, Cell segmentation, Deep learning, U-Net architecture, Image analysis, Convolutional neural network, Python programming, Image segmentation, Data preprocessing, Data augmentation

I. Introduction

The field of cell biology has undergone a major transformation with the advent of live cell imaging techniques that enable scientists to unravel the dynamic processes occurring within living cells. This technology has facilitated groundbreaking insights into cellular behavior, interactions, and responses to stimuli. The study of cell movement and cell behavior is of fundamental importance in areas such as cell embryology, wound healing, host defense mechanisms, and mechanisms of tumor cell metastasis and invasion. [1] However, as the volume and complexity of live cell imaging data increases, manual analysis of these images becomes increasingly laborious, time-consuming, and subjective. [2]

The challenge of automating the segmentation of live cell images holds the promise of streamlining research workflows, improving accuracy, and enabling the extraction of quantitative information from large datasets. [3] Traditional image processing methods often struggle to adapt to the diverse cell morphologies, varying lighting conditions, and complex spatial relationships in live cell images. [4] This creates a space for Machine Learning (ML) and Deep Learning (DL) techniques that can capture complex patterns and hierarchies in data, making them well suited for complex segmentation tasks.

Python serves as a powerful tool for implementing ML and DL models with its versatile libraries and frameworks. [5] Python's integration with built-in libraries such as TensorFlow, PyTorch, and scikit-learn enables the creation of advanced models for image segmentation. The focus of this project is to exploit the potential of Python and DL techniques to address the challenges of automated livecell segmentation, aiming to contribute to the cell biology research field.

II. Methods

A. U-Net Architecture

The core methodology employed in this project is the U-Net architecture, a deep learning framework specifically designed for image segmentation tasks. [6] The U-Net architecture has an encoder-decoder structure that utilizes convolutional neural networks (CNNs) to learn hierarchical features from input images. Connections between related layers help capture higher-level information, facilitate the preservation of spatial information, and enable precise segmentation. [7]

The U-Net architecture has proven successful in various biomedical image segmentation tasks, including cell and nucleus segmentation is shown in Figure 1. [8] In my implementation, I utilized the PyTorch library to create and train the U-Net model.

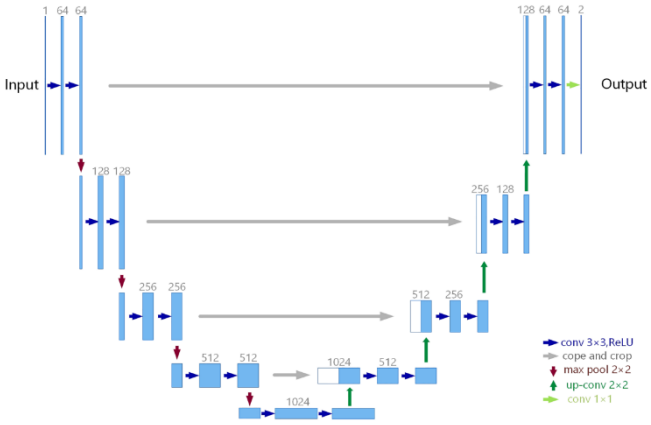

Figure 1 - U-Net Architecture [16]

B. Data Preprocessing and Augmentation

The dataset utilized in this project comprises live cell images and corresponding segmentation masks. The images are acquired from the LiveCell dataset provided by our lecturer. Prior to model training, the images and masks underwent preprocessing steps including resizing to a consistent resolution of 256x256 pixels and normalization to a range of [0, 1].

To mitigate overfitting and increase the diversity of the dataset, data augmentation techniques were applied. These techniques included random rotations, flips, zooms, random brightness and contrast. This augmented dataset was then used for model training.

C. Model Implementation and Training

The U-Net model was implemented using PyTorch, a popular deep learning framework. 9 The binary cross-entropy loss function was chosen as the optimization criterion, as it is well-suited for binary segmentation tasks. I utilized Adam optimizer [9] for weight updates.

To prevent overfitting and assess model performance on each epoch step, the dataset was split into training and validation sets, with 15% of the data reserved for validation. During training, I employed a learning rate scheduler with step

decay to adaptively adjust the learning rate as training progressed.

D. Evaluation Metrics

To assess the effectiveness of the U-Net model, I employed several evaluation metrics. The Intersection over Union (IoU), also known as the Jaccard index [10], measures the overlap between predicted and ground truth segments is shown in Figure 2. Additionally, we computed pixel-wise accuracy and loss on both the training and validation sets.

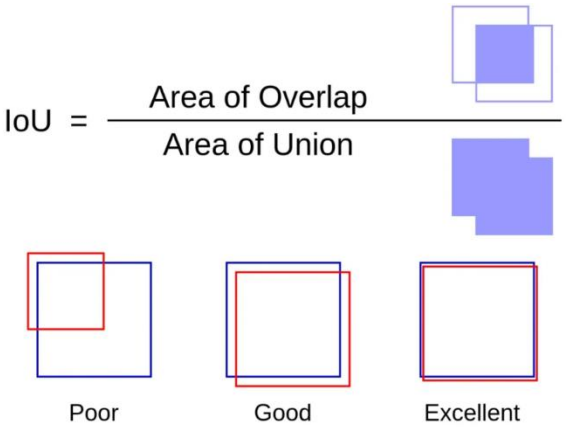

Figure 2 - Diagram represent Intersection over Union (IoU) [11]

III. Data Description

A. Dataset Source

The dataset utilized for this project is derived from the LiveCell dataset which is provided by Mr. Kouroudis, Ioannis lecturer to be used in this project. The LiveCell dataset is a collection of live cell microscopy images, capturing dynamic cellular processes over time. The dataset includes a diverse range of cell types and imaging conditions, making it suitable for training and evaluating deep learning models for cell segmentation.

B. Dataset Composition

The LiveCell dataset comprises a total of 3 train images, each accompanied by a corresponding ground truth segmentation mask with 100 test images. Figure 3 shows the 3 training images and their labels. Each image in the dataset captures a unique cellular scene, characterized by various cell shapes, sizes, and arrangements. The dataset encompasses a broad spectrum of cellular morphologies and appearances, enabling the model to learn to segment cells with diverse characteristics.

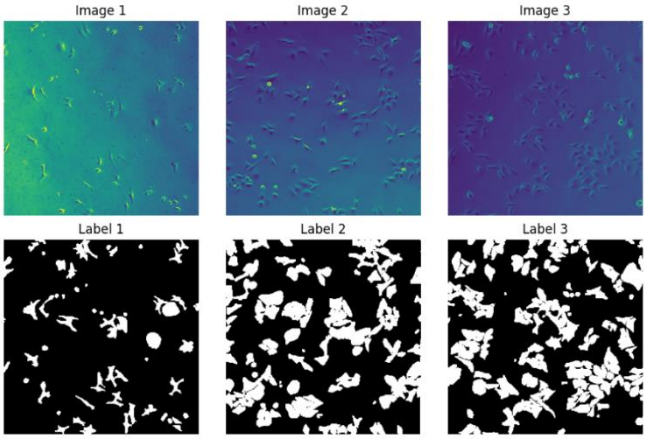

Figure 4 - Display of Images and Labels

C. Image Properties

The images in the LiveCell dataset are captured using fluorescence microscopy techniques, with each image containing distinct cellular structures, such as nuclei and cytoplasm. [12] The images are grayscale and have a resolution of 1024x1024 pixels. The pixel values are in the range [8114, 65535], with higher values indicating regions of higher fluorescence intensity.

D. Ground Truth Segmentation Masks

Accompanying each image in the dataset is a corresponding ground truth segmentation mask. These masks are binary images that outline the boundaries of individual cells within the image. The masks are manually annotated by domain experts and serve as the reference for evaluating the segmentation accuracy of the trained U-Net model.

E. Data Preprocessing

Prior to model training, the dataset underwent preprocessing steps to ensure uniformity and facilitate convergence during training. The images and segmentation masks were resized to a consistent resolution of 256x256 pixels. Furthermore, pixel intensity normalization was applied to the images to bring their values within the [0, 1] range.

F. Data Augmentation

To enhance the model's ability to handle variations in cell morphology and appearance, data augmentation techniques were employed during training. These techniques included random rotations, horizontal and vertical flips, and random zooms. By introducing these variations, the model becomes more robust and capable of accurately segmenting cells in a variety of scenarios. In addition, with the use of boosting methods, the number of training images increased from 3 to 1773, resulting in a significant increase in the number of samples. Figure 4 shows the first 3 augmented images and labels.

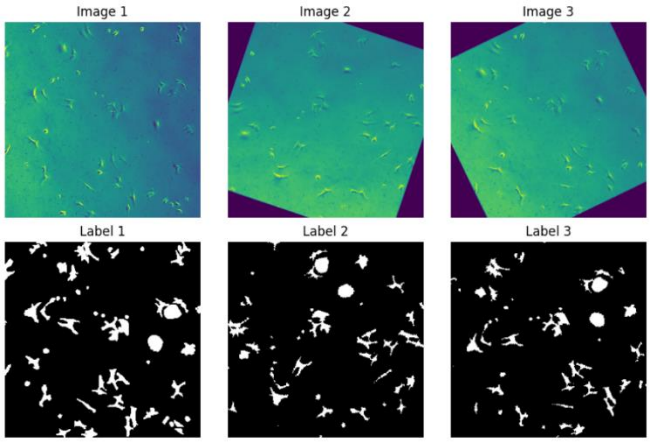

Figure 3 - Display of First 3 Augmented Images and Labels

IV. Building the Model

A. Model Architecture

The cornerstone of my approach lies in the utilization of the U-Net architecture, a deep learning model specifically designed for semantic segmentation tasks. The U-Net architecture's distinctive feature is its U-shaped design, which incorporates both downsampling and upsampling pathways to effectively capture both global context and local details in the input images. [13]

The U-Net model comprises an encoder and a decoder. The encoder, often referred to as the contracting path, consists of convolutional and pooling layers that progressively reduce the spatial dimensions of the input image while extracting higher-level features. [14] On the other hand, the decoder, or expansive path, involves transposed convolutions and concatenation operations to recover the spatial resolution and produce the final segmentation map.

The U-Net model employed in the project consists of several key components, each contributing to its segmentation process:

- **Encoder:** The encoder path consists of multiple convolutional layers, each followed by batch normalization and rectified linear unit (ReLU) activation. This sequence effectively extracts features from the input image, reducing its spatial dimensions while increasing the number of feature channels. The encoding process facilitates the understanding of global context and significant features.
- **Middle Layers:** The model further processes features through the middle layers, comprising additional convolutional layers with ReLU activation. These layers serve to capture more intricate patterns and details within the encoded features.
- **Decoder:** The decoder path involves a series of up-sampling layers combined with skip connections from the encoder. This structure facilitates the reconstruction of spatial information lost during the encoding process. The decoder gradually increases spatial resolution while simultaneously merging high-level semantic information from the encoder, allowing for accurate segmentation.
- **Output Layer:** The final layer of the decoder employs a 1x1 convolutional layer to produce the segmentation mask. The output layer's single-channel configuration corresponds to the binary nature of my segmentation task.

B. *Challenges and Solutions*

During the model development process, several challenges were encountered that demanded careful consideration. One primary challenge was the relatively limited size of the dataset. Deep learning models, especially those with a substantial number of parameters, require a substantial amount of data to avoid overfitting. To mitigate this challenge, I employed data augmentation techniques, as mentioned in the previous section, to artificially expand the dataset and introduce variability.

Another challenge was the class imbalance within the dataset. In live cell microscopy images, the regions occupied by cells are often significantly smaller in comparison to the background. This imbalance can lead the model to prioritize background predictions, resulting in reduced segmentation accuracy. To address this, I utilized class-weighted loss functions that assign higher importance to the minority class (cell regions) during training.

C. *Training Process*

The U-Net model was implemented using the PyTorch deep learning framework. The model was trained using a training-validation split of 85-15. Cross-validation was not applied due to the limited size of the dataset.

The Adam optimizer [9] is employed with a learning rate of 0.001 for parameter updates. The reason for keeping the learning rate low is to try to prevent overfitting. The model's performance was evaluated using the Intersection over Union (IoU) a metric commonly used for segmentation tasks, which quantifies the overlap between predicted and ground truth segmentations.

For the training process, we utilized Binary Cross-Entropy with Logits Loss. This loss function is well-suited for binary classification tasks such as image segmentation. It measures the difference between predicted logits and ground truth labels, helping the model to learn the correct boundaries of cell segmentations. Its compatibility with the sigmoid activation function is used in the final layer of the U-Net model. Figure 5 shows the plots of training and validation metrics, such as losses and Intersection over Union (IoU) over epochs.

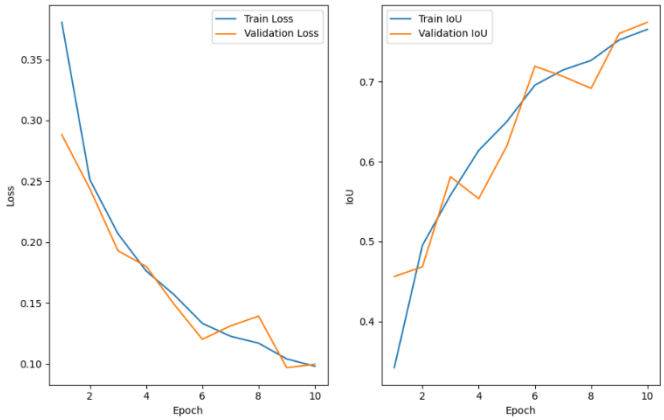

Figure 5 - Losses and IoUs over Epochs

D. *Regularization and Hyperparameters*

To prevent overfitting, several regularization techniques were incorporated into the model. Dropout layers were added to the decoder part of the U-Net to encourage feature diversification and enhance generalization. Batch normalization was also utilized to stabilize and expedite training. This technique normalizes the output of each layer, leading to smoother optimization and allowing for larger learning rates. [15] Learning rate scheduling was employed to dynamically adjust the learning rate during training. Additionally, learning rates, dropout rates, and batch sizes were among the hyperparameters adjusted to optimize model performance.

V. Conclusion

Efforts in the field of automated cell segmentation through live cell imaging and deep learning have yielded both insightful results and valuable insights. The U-Net architecture, harnessed for its skill in semantic segmentation tasks, proved to be a robust choice for this challenging task. Through this project, an important step has been taken in the automation of cell segmentation, mitigating the labor-intensive and subjective nature of manual analysis.

Upon evaluation, the trained U-Net model demonstrated impressive performance in segmenting live cell microscopy images. This performance is particularly promising considering the limited size of our dataset and the inherent complexity of cell morphology and dynamics. The labels produced by my model applied in the test images are shown in figure 6.

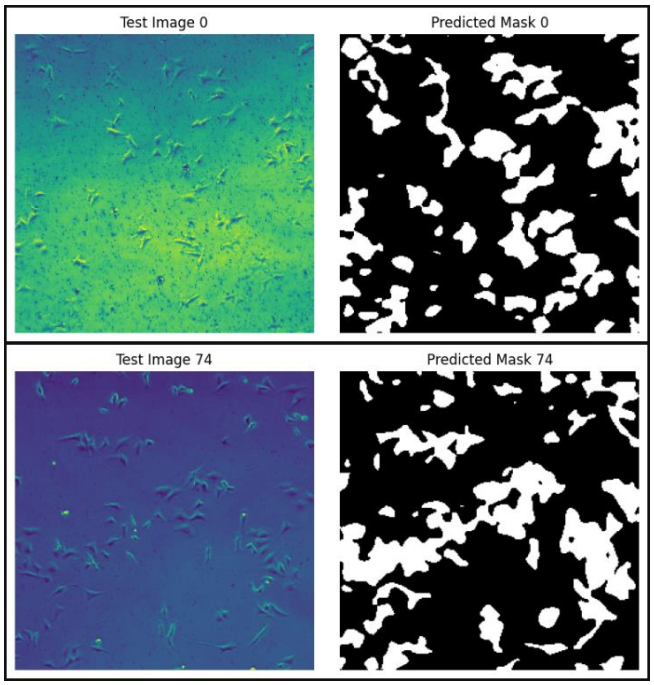

Figure 6 - Display of Random Test Images and Model Labels

However, it's essential to acknowledge the limitations and areas for improvement in my work. The relatively small dataset used for training and validation remains a limiting factor. Despite employing data augmentation techniques, the model's capacity to generalize to unseen data could be enhanced with a more extensive and diverse dataset. Further exploration of advanced techniques like focal loss or GAN-based augmentation could yield better results.

In conclusion, this project underscores the potential of deep learning in revolutionizing the analysis of live cell imaging data. The U-Net model's capability to accurately segment cells open doors to accelerating research in cell biology, drug discovery, and disease understanding. The lessons learned from this project serve as steppingstones for future advancements in automated cell analysis.

ACKNOWLEDGMENT

I extend our sincere gratitude to Prof. Dr. Gagliardi, Alessio and all other lecturers of Python for Engineering Data Analysis course for providing us with invaluable guidance and resources throughout the duration of the course and project. Their expertise and support have been instrumental in shaping my understanding and approach to the challenges of live cell segmentation. Additionally, our appreciation goes out to the open-source community for developing and sharing the tools and libraries that empowered us to realize the goals of this project.

REFERENCES

[1] K. Wu, D. Gauthier and M. Levine, "Live cell image segmentation," *IEEE Transactions on Biomedical Engineering*, pp. 1 - 12, 1995.

[2] P. C. Q. J. S. H. L. L. A. H. H. Ahmed, "Artificial intelligence in radiology," *Nature Reviews Cancer*, p. 500–510, 2018.

[3] G. Du, X. Cao, J. Liang, X. Chen and Y. Zhan, "Medical Image Segmentation based on U-Net: A Review," *Imaging Science & Technology*, vol. 64, no. 2, pp. 1-12, 2020.

[4] T. A. S. H. R. G. M. R. &. R. J. Nketia, "Analysis of live cell images: Methods, tools and opportunities," *Methods*, vol. 115, pp. 65-79, 2017.

[5] J. VanderPlas, Python Data Science Handbook, 1005 Gravenstein Highway North, Sebastopol: O'Reilly Media, 2016.

[6] O. F. P. B. T. Ronneberger, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Medical Image Computing and Computer-Assisted Intervention*, pp. 234 - 241, 2015.

[7] Z. Gu, J. Cheng, H. Fu, K. Zhou, H. Hao, Y. Zhao, T. Zhang, S. Gao and J. Liu, "CE-Net: Context Encoder Network for 2D Medical Image Segmentation," *IEEE Transactions on Medical Imaging*, vol. 38, pp. 2281-2292, 2019.

[8] N. Ibtehaz and M. S. Rahman, "MultiResUNet : Rethinking the U-Net architecture for multimodal biomedical image segmentation," *Neural Networks*, vol. 121, pp. 74-87, 2020.

[9] D. P. Kingma and J. L. Ba, "Adam: A Method For Stochastic Optimization," *ICLR*, 2015.

[10] J. Yu, J. Xu, Y. Chen, W. Li, Q. Wang, B. Yoo and J.-J. Han, "Learning Generalized Intersection Over Union for Dense Pixelwise Prediction," *Proceedings of the 38th International Conference on Machine Learning*, pp. 12198-12207, 2021.

[11] N. Tomar, "What is Intersection over Union (IoU) in Object Detection?," Idiot Developer, 7 2 2023. [Online]. Available: https://idiotdeveloper.com/what-is-intersection-over-union-iou/. [Accessed 3 8 2023].

[12] M. &. S. C. Pachitariu, "Cellpose 2.0: how to train your own model," *Nature Methods*, vol. 19, p. 1634–1641, 2022.

[13] Y. L. Y. L. Z. Z. G. L. W. K. K. B. Chen, "TransAttUnet: Multi-level Attention-guided U-Net with Transformer for Medical Image Segmentation," *TRANSACTIONS ON INSTRUMENTATION & MEASUREMENT*, 2022.

[14] L. S. Y. F. L. Y. Z. X. Yin, "U-Net-Based Medical Image Segmentation," *Healthcare Engineering*, vol. 2022, 2022.

[15] T. D. I. A. M. A. S. Shibani, "How Does Batch Normalization Help Optimization?," *32nd Conference on Neural Information Processing Systems*, 2018.

[16] L. Q. Z. J. W. Y. Y. Y. F. S. Zhang H, "RatUNet: residual U-Net based on attention mechanism for image denoising," *PeerJ Computer Science*, 2022.