PROJECT-REPORT:

"Semantic Segmentation of Lung and Colon Cancer Histopathological Images Using U-Net"

Prepared by:

Team Phoenix (Santosh Bogati)

Submitted to:

Dr. Yutao He

Table of Content

1 Introduction

Cancer diagnosis often relies on analyzing tissue samples under a microscope. This project aimed to use deep learning, specifically the U-Net architecture, to automatically identify and segment cancerous regions in histopathological images. This automation can significantly aid doctors in diagnosing and planning treatments.

2 Objective

The primary objective of this project was to develop a semantic segmentation model using the U-Net architecture to accurately segment lung and colon cancer histopathological images. This model helps in the precise localization of cancerous regions, which is critical for diagnosis and treatment planning.

3 Background and Rationale

Accurate cancer diagnosis is like finding a needle in a haystack. Traditional methods are time-consuming and often vary between different doctors. Deep learning, particularly the U-Net model, acts like a skilled artist who can consistently and precisely paint cancerous regions on medical images, improving diagnostic accuracy and speeding up the process.

Why U-Net?

U-Net, introduced by Ronneberger et al. (2015), is particularly well-suited for biomedical image segmentation because of its ability to learn from relatively few images and produce high-quality segmentations. It uses a combination of convolutional and up sampling layers, with skip connections that help retain spatial information, essential for precise localization of cancerous regions.

4 Methodology

4.1 Device Details:

The device used for this model training had following specs:

- ✓ GPU: UHD 620
- ✓ Microprocessor: Intel® Core™ i5-8250U (1.6 GHz base frequency, up to 3.4 GHz with Intel® Turbo Boost Technology, 6 MB cache, 4 cores)
- ✓ Memory: 8 GB DDR4-2933 SDRAM
- ✓ Device: Envy HP13

4.2 Dataset

We used the Lung and Colon Cancer Histopathological Images dataset from Kaggle. This dataset originally contained 25,000 images across five classes (5,000 images/classes) mentioned below. However, because of the shortage in computational capability (620 GPU, i5/8th Gen), the dataset used for this project was reduced to 750 images across five classes, with 75 images per class:

- ✓ Lung benign tissue
- ✓ Lung adenocarcinoma
- ✓ Lung squamous cell carcinoma
- ✓ Colon adenocarcinoma
- ✓ Colon benign tissue

To enhance the dataset, we applied data augmentation techniques, increasing the diversity and robustness of the model.

4.3 Preprocessing

The images were resized to 128x128 pixels and normalized to have pixel values in the range [0, 1]. We also simulated binary masks by converting images to grayscale.

4.4 Data Splitting

The data was split into training, validation, and testing sets with proportions 70%, 15%, and 15%, respectively. This ensured the model learned effectively and was evaluated accurately.

4.5 Model Implementation

4.5.1 U-Net Architecture

The U-Net model was implemented using TensorFlow and Keras. It consisted of an encoder to capture context, a bottleneck, and a decoder to enable precise localization. Skip connections between corresponding layers of the encoder and decoder helped retain spatial information.

4.5.2 Training

The model was trained using the Adam optimizer with a learning rate of 0.0001, a loss function of binary cross entropy, for 30 (initially tried with 50 epochs but had to be reduced) epochs and a batch size of 32.

5 Results

Because of the limited dataset that our PC could process, the trained model was underfitting. Consider the following metrics:

- ✓ Validation Loss: 59%
- ✓ Validation Accuracy: 9%

With such validation accuracy and loss, the other performance metrics – as one can imagine – such as F1, Precision, and Recall were nightmares. Following let us have some glimpse of what it means to have an underperforming trained model:

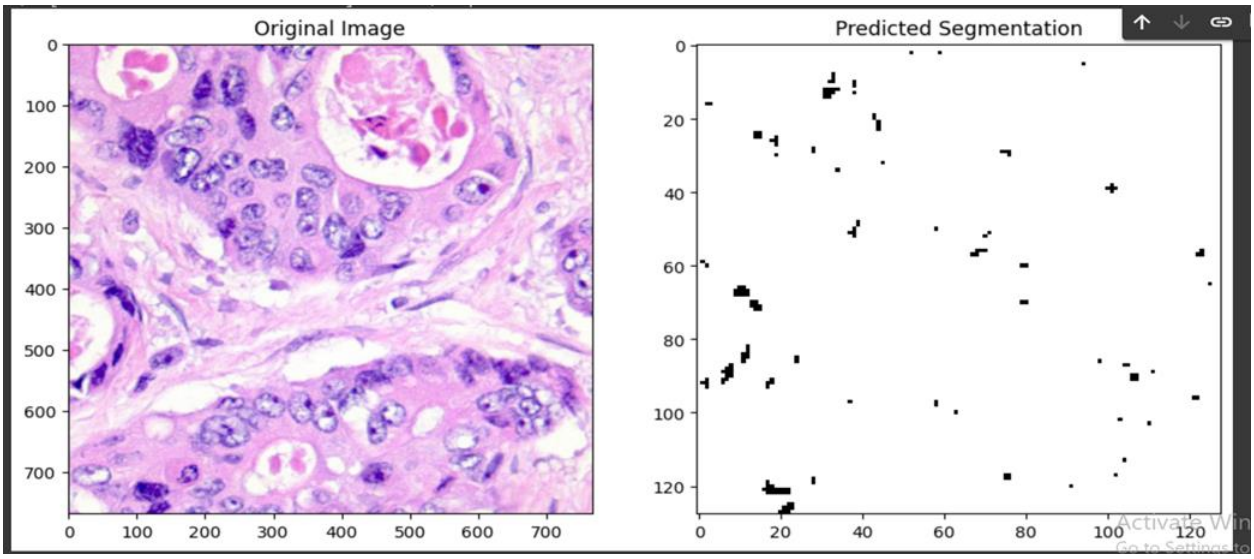

Figure 1:Model predicting an image when unseen data was fed

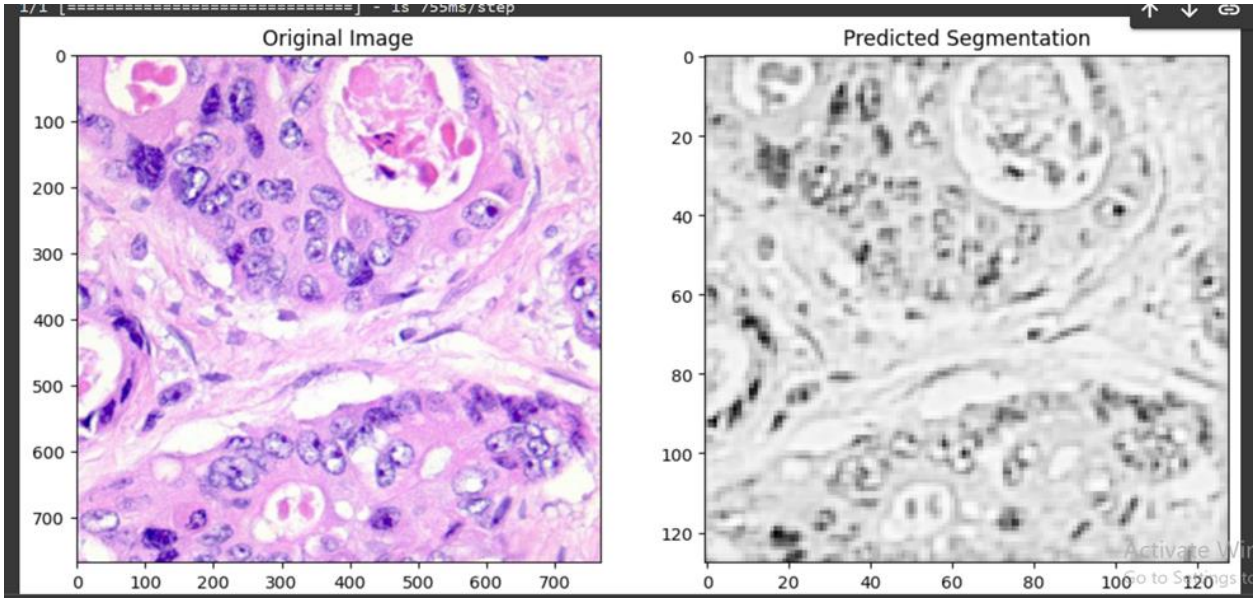

Figure 2:Model Predicting an image when training image was fed

Clearly, the Fig(1) and Fig (2) depict how weak the trained model was, attributed to low amount of data used in the training process.

6 Challenges:

The only challenge Team Phoenix faced during the project was computational resources limitations. Firstly, the project was tried in local computing system with 250 images, and then 150 images (VScode). The laptop was so heated and crashed many times. Then, the project was switched to Cloud Computing (Google Collab), which was successful to train the model at 75 images/5 classes.

7 Conclusion

Along this solo-project journey, and first time experience in training the model, it was well rewarding – despite the fact that the model failed to render the expected results. The model was itself GPU-exhaustive while training, and cloud computing was somewhat helpful. Therefore, the key takeaways were first having the right size of data, and then sufficient computing power in order to successfully train the model.

8 Acknowledgements

Special thanks to Professor Yutao He for his timely responses and invaluable support throughout this project. At the same time, I do not forget myself, for consistency along the way.

9 References

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. Medical Image Computing and Computer-Assisted Intervention (MICCAI).

Zhang, Z., Liu, Q., & Wang, Y. (2018). Road Extraction by Deep Residual U-Net. IEEE Geoscience and Remote Sensing Letters.