Breast Ultrasound Image Segmentation Analysis: Emma Duffy, Hammad Hassan, Yuna Kim

A. Introduction

As the "most commonly diagnosed cancer among U.S. women", breast cancer affects approximately 1 in 8 women in the U.S [1]. Therefore, the motivation to explore image segmentation in breast ultrasound images is an important and applicable problem that requires diligent and creative solutions.

Our data focuses on the ultrasound images of 600 female patients, encompassing 780 total ultrasound image provided from a Kaggle notebook [3]. Image segmentation is highly pertinent to breast cancer in ultrasound imaging because it allows for tumors to be segmented out of ultrasound imaging to assist in the classification, benign or cancerous, of those spots found.

Breast image segmentation is highly significant because the early detection of breast masses can significantly improve medical prognosis and is important in identifying abnormalities that may be found within an ultrasound image. Specifically, using new algorithmic technology to do image segmentation is crucial because it can quickly analyze large volumes of medical images, allowing healthcare professionals to focus their attention on interpreting results and making treatment decisions. Segmentation algorithms are also much more accurate and reliable than manual segmentation methods, reducing the chance of oversight or misinterpretation and aiding in time efficiency. Lastly, establishing standardized segmentation methods can ensure consistency in the interpretation of ultrasound images and more reliable diagnoses.

Our project uses a U-Net model to conduct the image segmentation, relying on a variety of baseline hyperparameters in order to collect information on the models' performance. Through our work, we attempt to alter the hyperparameters from their default types to calculate accuracy, AUC, and loss values that improve from baseline. We experiment in using different optimization techniques such as Adam, SGD, Adam Decay, and SGD Decay. We also utilize different loss techniques such as KLD, DICE, and Binary Cross Entropy. By creating various combinations of these techniques, we sought to find the outcome with high accuracy and minimal loss.

By finding a solution that seeks to improve the accuracy and precision of breast ultrasound image segmentation, better care can be given to patients who are identified as being diagnosed with breast cancer, in a feasible amount of time to continue living life as usual with minimal disruptions.

B. Related Work

Image segmentation and region detection have evolved significantly since their foundation in the 1960s. Early methods relied on simple algorithms, primarily focusing on thresholding pixel values to delineate distinct regions. However, these approaches were limited by their lack of sophistication, particularly when applied to complex images.

In contemporary research, advanced models such as U-Net and DeepLab have emerged as powerful tools for image segmentation, leveraging convolutional neural networks (CNNs) to achieve remarkable accuracy. While U-Net is widely recognized for its effectiveness in this domain, DeepLab stands out as another prominent option, particularly renowned for its application in semantic segmentation.

In the realm of medical imaging, DeepLab has proven invaluable, especially in reducing noise inherent in biomedical images. For instance, in mammography, precise segmentation of the breast region is crucial for accurate diagnosis of breast cancer. Utilizing DeepLab, researchers have achieved superior results compared to traditional methods and even other CNN architectures like U-Net [4]. The enhanced performance of DeepLab in extracting the breast region from mammograms underscores its potential in advancing computer-aided detection systems for breast cancer.

C. Method

C.1. U-Net Model

This is a type of CNN that is effective when the output is expected to be a high resolution segmentation corresponding to the input size of an image. Its architecture is characterized by a U-shaped structure consisting of a contraction and an expansion path. The contraction path is used for capturing context in an image. It has repeated applications of two 3x3 convolutions, each of which are followed by a rectified linear unit (ReLU) and a 2x2 max pooling operation. The Expansion path of the U-Net consists of synergistically combining the feature and spatial information through a large sequence of nonsampling combining those with the high-resolution features from the initial contraction path. It begins by nonsampling of the feature map and then follows that by a 2x2 convolution that divides the number of feature channels into two and then concatenates them with corresponding feature map from the contracting path and the two 3x3 convolutions, each of which are followed by a ReLU. Finally at the last layer, a 1x1 convolution is used to map each of the 64-component feature vectors to the desired number of classes.

C.2. Key Characteristics

As the model's architecture is symmetric, it allows for an efficient information transmission. As the data passes the initial contracting path, it is forced to lose spatial reso-

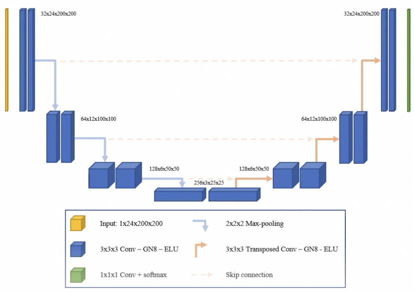

Figure 1. U-Net Architecture [2]

lution but regains extensive contextual depth. Followingly, during the extensive path, it uses transposed convolutions to regain spatial dimensions. It also employs a feature reuse system where these connections are able to bypass the network's layers and bring feature maps from the contracting path directly into the expansive path. U-Net is also very efficient with data as it can learn effective feature representations from a relatively small number of training images. In our case, we employed 780 ultrasound images derived from 600 female patients.

C.3. Upsampling

We employed Upsampling and Conv2DTranspose as our two different upsampling upsampling. They are known as two common methods for upsampling in neural networks, particularly in the context of tasks like image segmentation. Upsampling2D is a simple method that duplicates rows and columns of the input tensor. It doesn't have any trainable parameters and doesn't perform any learning during the upsampling process. It's fast and easy to implement, but it may lead to checkerboard artifacts in the output images, especially when used repeatedly in deep architectures. Conv2DTranspose, also known as ”deconvolution” or ”fractionally-strided convolution,” involves learning the upsampling process. It uses transposed convolutional layers to increase the spatial resolution of the input feature maps. Unlike Upsampling2D, Conv2DTranspose has trainable parameters, allowing the network to learn the upsampling process and potentially capture more complex patterns in the data. It can mitigate issues like checkerboard artifacts through learned upsampling.

C.4. Optimizers and Learning Rates

We tried 2 optimizers, which are stochastic gradient descent (SGD) and Adam optimizer, and 2 types of learning rates, which are fixed learning rate and exponentially descending learning rate. Therefore, in total, we tried 4 different types of optimization. SGD is a classical optimization algorithm widely used in machine learning and deep learning. It updates model parameters in the direction opposite to the gradient of the loss function with respect to those parameters. It involves taking small steps towards the minimum of the loss function in each iteration. SGD has a parameter called the learning rate, which determines the size of the steps taken during optimization. In standard SGD, this learning rate remains constant throughout training. Adam (Adaptive Moment Estimation) is a popular optimization algorithm that adapts the learning rate for each parameter during training. It combines ideas from both Momentum optimization and RMSProp. Adam maintains adaptive learning rates for each parameter based on estimates of first and second moments of the gradients. It's generally regarded as more robust and faster than vanilla SGD because it automatically adjusts learning rates and typically converges faster.

C.5. Loss Functions

We tried Kullback-Leibler Divergence (KLD), Dice Loss, and Binary Cross Entropy (BCE) loss functions. KLD is a measure of how one probability distribution diverges from a second, expected probability distribution. In the context of neural networks and probabilistic models, KLD is often used as a regularization term to encourage the learned distribution to be close to a predefined distribution, typically a prior distribution or a target distribution. It measures the information lost when the predicted distribution is used to approximate the target distribution. Dice Loss, also known as the Sørensen-Dice coefficient or F1 score loss, is commonly used in tasks like image segmentation. It measures the similarity between two samples and ranges from 0 to 1, where 0 indicates no overlap between the predicted and target segmentations, and 1 indicates perfect overlap. Dice Loss is defined as the complement of the Dice coefficient, which is calculated as twice the intersection of the predicted and target masks divided by the sum of their sizes. BCE Loss, also known as log loss or logistic loss, is a standard loss function used for binary classification tasks. It measures the dissimilarity between two probability distributions, typically the predicted probabilities and the true labels. BCE Loss penalizes incorrect predictions logarithmically, with a higher penalty for larger deviations from the true labels.

C.6. Data Processing

We tried to resize, normalize, standardize, and augment the data and try to improve the performance. Let's deeply talk about how each method works. Resizing is related to adjusting the dimensions of input data, such as images, to a uniform size helps ensure consistent input for the model. It can reduce computational overhead and improve generalization by providing uniformity in input sizes during training. Normalizing is scaling the values of features to a common range, like between 0 and 1 or -1 and 1, helps stabilize training by bringing features to a similar scale. This pre-

Method	Loss	Accuracy	AUC
Upsampling2D	0.1877	0.9251	0.9728
Conv2DTranspose	0.1758	0.9290	0.9761

Table 1. Model: Upsampling Results.

vents certain features from dominating others during training and promotes faster convergence and better generalization. Standardizing is transforming features to have a mean of 0 and a standard deviation of 1 centers the data around zero and scales it to have a unit variance. It can accelerate optimization algorithms and make models more robust to outliers by creating a uniform optimization landscape. Finally, augmenting is generating new training samples by applying transformations like rotation, scaling, flipping, or cropping increases the diversity of training data. This enhances the model's ability to generalize to unseen samples and improves robustness to variations in input data, such as changes in lighting conditions or viewpoints.

D. Experiments

D.1. Model: Upsampling

For our two upsampling experiments, loss value, accuracy, and AUC score were all improved. There are several reasons that Conv2DTranspose can perform better on breast cancer segmentation. - Learned Upsampling: Conv2DTranspose learns the upsampling process during training, allowing the network to adapt to the specific characteristics of the data. In tasks like breast cancer segmentation, where capturing fine details is crucial, learned upsampling can be more effective than simple duplication of pixels. - Feature Learning: Conv2DTranspose layers have trainable parameters, allowing the network to learn complex features and patterns in the data during both downsampling and upsampling stages. This ability to capture intricate details in the data can be beneficial for tasks like segmenting breast cancer regions from mammography images, where subtle features can be indicative of the presence of cancerous tissue. - Mitigation of Checkerboard Artifacts: Checkerboard artifacts are common in neural networks that use simple upsampling methods like nearest-neighbor or bilinear interpolation. These artifacts can degrade segmentation performance by introducing unwanted patterns in the output. Conv2DTranspose, with learned upsampling, can mitigate these artifacts, resulting in smoother and more accurate segmentations.

Therefore, Conv2DTranspose, by capturing more detailed features during upsampling, can potentially reduce the discrepancy between the predicted segmentation mask and the ground truth mask, leading to lower reconstruction loss and hence better segmentation performance compared to simpler upsampling methods like Upsampling2D.

Method	Loss	Accuracy	AUC
SGD+fixed LR	0.1877	0.9251	0.9728
Adam+fixed LR	0.3068	0.9083	0.9052
SGD+Decaying LR	0.2970	0.9083	0.9285
Adam+Decaying LR	0.2973	0.9082	0.9278

Table 2. Optimization Results.

D.2. Optimization

Our purpose was to analyze whether changes to the optimization method significantly affect the model's performance. Unfortunately however this did not seem to be the case as observed below. However, based on these results we can draw following outcomes:

- Robustness to Optimization Method: Deep learning models for tasks like breast cancer segmentation often deal with large amounts of data and complex architectures. These models are often robust to the choice of optimization method due to their capacity to learn from the data. While different optimizers may converge at different rates or reach slightly different optima, the overall performance difference might not be substantial, especially for well-tuned models. - Task Complexity and Dataset Characteristics: Breast cancer segmentation typically involves processing medical imaging data, which often contains consistent patterns and structures. The task's inherent characteristics might make the optimization landscape relatively smooth and allow various optimizers to converge to similar solutions. Additionally, the dataset size and quality can influence the sensitivity of the optimization process to the choice of optimizer. - Learning Rate Adaptation: While the learning rate is a critical hyperparameter, modern optimizers like Adam adaptively adjust the learning rate during training based on the gradient magnitudes and previous gradients' moments. This adaptiveness helps in navigating the optimization landscape efficiently, making the choice of a specific learning rate less crucial.

D.3. Loss

Through the loss function experiment, we find out that KLD performs better than the other loss functions in terms of accuracy. This was a very encouraging outcome and believe it can be explained by the following reasons:

- Probability Distribution Alignment: In breast cancer segmentation, it's common to use probabilistic models, such as variational autoencoders (VAEs) or generative adversarial networks (GANs), to model the uncertainty in the segmentation task. KLD loss encourages the learned distribution to match a predefined distribution, such as a prior distribution or a target distribution derived from the

Method	Loss	Accuracy	AUC
KLD	0.1489	0.9429	0.9850
DICE	0.0916	0.9083	0.9070
BCE	0.2269	0.9083	0.9624

Table 3. Loss Results.

Method	Loss	Accuracy	AUC
No preprocess	0.1877	0.9251	0.9728
Resize+Norm+Std+Aug.	0.3070	0.9083	0.9059

Table 4. Data Results.

ground truth segmentations. This alignment helps in capturing the inherent uncertainty in the segmentation task, which is crucial in medical imaging where the boundaries between healthy and cancerous tissues can be ambiguous. - Regularization Effect: KLD loss acts as a regularization term, encouraging the learned distribution to be close to a predefined distribution while minimizing the information lost during the approximation process. This regularization effect helps prevent overfitting and encourages the model to learn meaningful representations of the data, which is beneficial in tasks like breast cancer segmentation where the dataset might be limited and prone to noise.

Therefore, to summarize, KLD loss performs well on the breast cancer segmentation task due to its ability to align learned distributions with predefined distributions and its regularization effect, which are crucial aspects in accurately segmenting cancerous regions from medical imaging data.

D.4. Data

Following our data processing experimentation, we did not observe any improvement in the model's performance. It is because the effectiveness of these techniques may vary depending on factors such as data quality, task specificity, and the risk of overfitting. For the case that we have, the data is already well-preprocessed and the task doesn't benefit significantly from additional preprocessing that these techniques might not lead to substantial improvement, and even made it worse in model performance.

E. Conclusions

This study aimed to demonstrate the effectiveness of a U-Net model in enhancing breast ultrasound image segmentation. Through experimenting with various hyperparameters and optimizing techniques including Adam, SGD, and advanced loss functions, we are able to demonstrate improvements in U-Net accuracy as well as efficiency. Using Conv2DTranspose over simpler unsampling methods proved effective in capturing intricate details. Additionally, this study highlights the effectiveness of KLD in enhancing model accuracy. While our data processing did not seem to improve performance, it does provide a valuable insight that task specific characteristics might limit the impact of these techniques. Combined, this study provided valuable insights into key techniques that may play a role in enhancing the precision of ultrasound image segmentation.

F. Contribution

- Introduction - Emma
- Related Work - Emma
- Methods - Hammad
- Experiments - Yuna
- Conclusions - Hammad

https://github.com/hassanhd0705/Ultrasound-Image-Segmentation/tree/main

References

[1] BreastCancer.org. Breast cancer facts and statistics. 2024. 1

[2] Johannes Schmidt. Creating and training a u-net model with pytorch for 2d & 3d semantic segmentation: Model building [2/4]. 2020. 2

[3] Arya Shah. Breast ultrasound images dataset. Kaggle Dataset, 2021. URL: https://www.kaggle.com/datasets/aryashah2k/breast-ultrasound-images-dataset/data. 1

[4] Zhao D. Zhou K, Li W. Deep learning-based breast region extraction of mammographic images combining pre-processing methods and semantic segmentation supported by deeplab v3. *Technol Health Care*, pages 173–190, 2022. 1

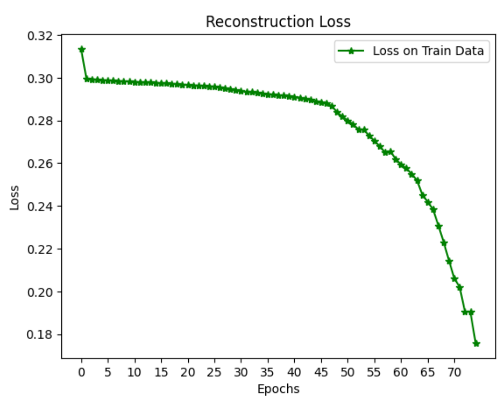

Figure 2. Conv2DTranspose Reconstruction Loss

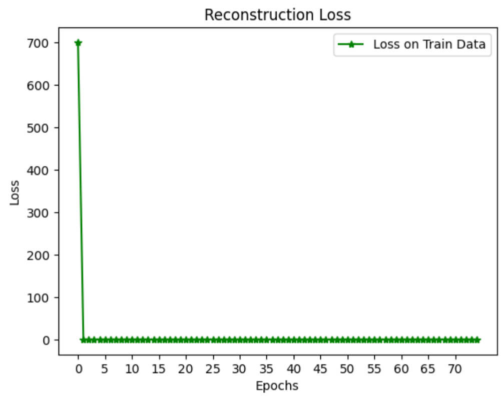

Figure 3. Adam+Fixed Learning Rate Reconstruction Loss

Figure 4. SGD+Fixed Learning Rate Reconstruction Loss

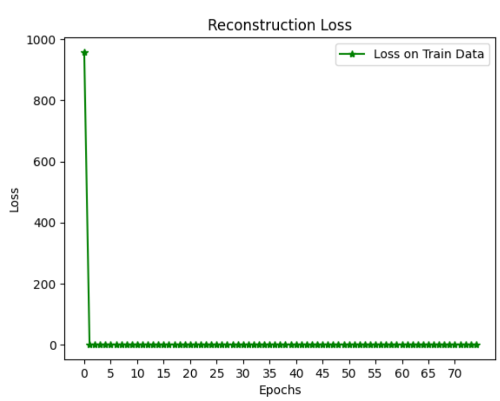

Figure 5. Adam + Exponentially Decaying Learning Rate Reconstruction Loss

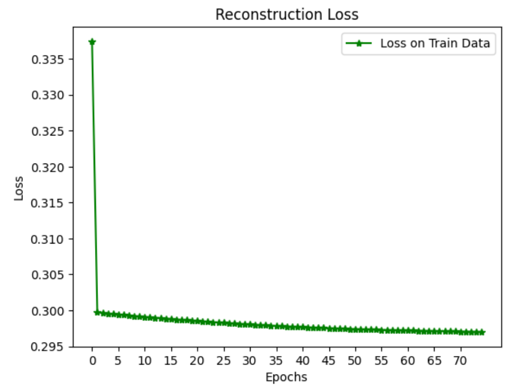

Figure 6. SGD + Exponentially Decaying Learning Rate Reconstruction Loss

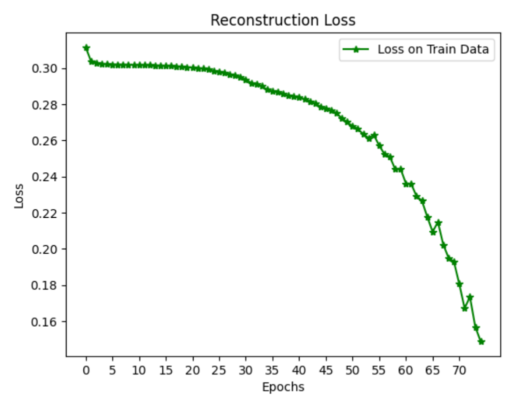

Figure 7. KLD Reconstruction Loss

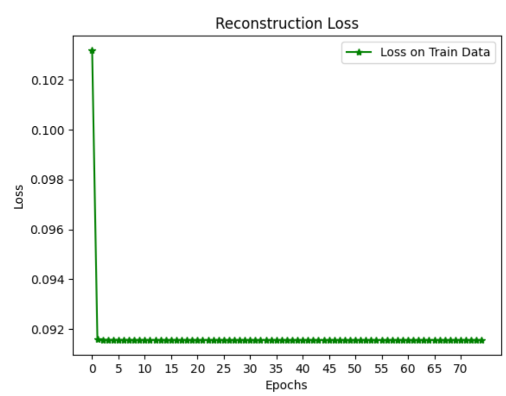

Figure 8. DICE Reconstruction Loss

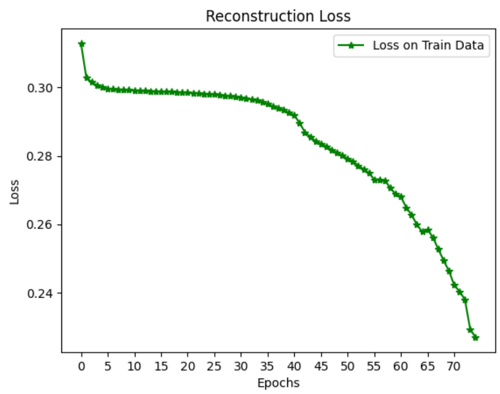

Figure 9. Binary Cross Entropy Reconstruction Loss

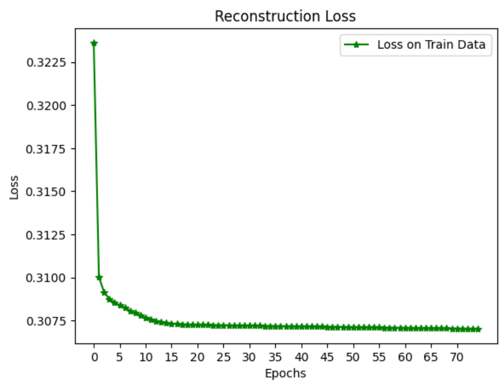

Figure 10. Resize+Norm+Std+Aug. Reconstruction Loss