Semantic Segmentation for Medical Ultrasound Imaging

by

Florin Andrei

A Capstone Project Paper Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

In

Data Science

University of Wisconsin – La Crosse

La Crosse, Wisconsin

December 2022

Abstract

Breast cancer is the type of cancer with the highest prevalence nowadays, around the world. According to the World Health Organization, millions of cases are diagnosed each year, and in the year 2020 it has caused approximately 685,000 deaths worldwide. Forecasts indicate that the number of cases and the number of deaths may increase for the foreseeable future.

Within a larger joint project by the University of Wisconsin - La Cross, and the Mayo Clinic Health Systems, this study has developed deep learning techniques that could be used to improve the diagnostic process for breast cancer. There are two main ways that deep learning technology could be applied: by providing real-time visual cues to radiologists while exploring lesions using ultrasound, and by creating predictions that become inputs for other models, in so-called ensemble methods, where multiple models work together to predict characteristics of the lesion.

Based on a review of the literature on medical imaging and image segmentation techniques, several semantic segmentation models were created and trained on breast ultrasound imaging datasets. The research found that the best models performed at a level that would be expected from state-of-the-art image segmentation techniques, adjusted for the additional challenges raised by ultrasound imaging.

Further work is needed within the parent project to integrate the models trained within this study with software usable by radiologists, and to explore the performance of ensemble methods where multiple models, including models trained in this study, work together to make predictions on ultrasound images that would ultimately lead to better diagnostic results and better patient outcomes.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

According to the World Health Organization (2021), there are millions of women diagnosed with breast cancer each year, and hundreds of thousands of deaths occur worldwide due to this disease. It is the world's most prevalent cancer. It occurs in every country of the world in women of any age. Beginning in the 1980s, it has been shown that improvements in survival occur with early detection programs.

Ultrasound imaging is a widely used technique in the process of finding and diagnosing breast cancer. While it is not routinely used as a screening test, it can be useful to distinguish between fluid-filled masses such as cysts - which are unlikely to be cancer - and solid masses which may prove to be malignant upon further investigation (American Cancer Society, 2022).

Statement of the Problem

When looking at the way the results of ultrasound imaging are interpreted, and the outcomes of biopsy follow-ups, the range of outcomes is very wide. For example, when looking at biopsies performed by the Mayo Clinic as follow-ups to ultrasound imaging, the positive rate is in the 31-51% range. This is not optimal from a patient management perspective.

What is needed is an automated system that could look at lesions of indeterminate status, and help make a more accurate assessment of their nature - benign or malignant. That would improve patient outcomes and reduce costs. Artificial intelligence has been shown to reduce false positive findings in the interpretation of breast ultrasound images - (see Shen et al., 2021).

Theoretical Framework

A widely used technique for the analysis of medical images is image segmentation. It divides the image into multiple regions that have similar properties. In other words, it distinguishes between background (parts of the image that are not of interest), and target (parts of the image that are of interest). Various machine learning and deep learning methods are used for this purpose - see Jahwar and Abdulazeez (2022).

Traditionally, some examples of segmentation techniques are: thresholding methods (simply based on the absolute value of each pixel), region-based segmentation (grouping neighboring pixels together based on similar values), and edge-based segmentation (finding boundaries between regions).

Deep learning, based on artificial neural networks, is a modern approach with great potential in image segmentation and classification. Convolutional Neural Networks (CNN) are often used for classification, where the output is a single class label.

For actual image segmentation, a class label needs to be assigned to each pixel. This would clarify not only the nature of the target being examined (benign lesion, malignant lesion), but also would isolate its shape from the background. Generating segmentation masks, and assigning different classes (pixel values) to each kind of segmented object, is known as semantic segmentation.

Statement of the Purpose

This study has produced deep learning models which create image segmentation masks from breast ultrasound images, and also classify the images based on the nature of the lesions (benign or malignant).

Identifying the shape of the lesion was one of the main goals. Given an ultrasound image, the models isolate the lesion from the background, and paint a "mask" over the lesion, indicating the region of interest in the image.

Identifying the nature of the lesion was another goal. If the images have enough information to distinguish benign lesions from malignant ones, the models create masks with different values for each type (benign vs malignant).

The goal was not to train several models and then pick a single winner - but to train several models such that each model performs as well as possible on its own on segmentation tasks. Different models have different strengths, and multiple models could become parts of an

ensemble method where each model contributes in a specific way - but the ensemble methods are outside the scope of this study.

Research Questions

Image segmentation is a deep learning topic that is actively researched. Using typical datasets (e.g. city and street images) which are commonly available, are large, and are made of high-quality images and labels, the performance of typical segmentation models can be very high. But medical ultrasound images, being uncommon, remain relatively less explored.

This study has explored the performance of deep learning models for semantic segmentation on breast ultrasound image datasets. Specifically, the intersection-over–union (IoU) metric has been used to evaluate the performance of the models on the validation dataset. Also, the consistency of the classification on a partially labeled dataset has been evaluated. Methodology details are described in Chapter 3.

One of the model architectures used in this project (SegFormer), while considered a state-of-the-art model for image segmentation in 2021 when used on generic image datasets, was not known to perform well on the particular task of segmenting breast ultrasound images. Its performance was evaluated during this project, and compared to a model (U-Net) known to perform well in this domain.

Significance of the Study

This study was done as part of an ongoing project called Tumor Detection from BUS (Breast Ultrasound) Images, at the University of Wisconsin - LaCrosse in collaboration with the Mayo Clinic, represented by Dr. Jeff Baggett at UWL, and Dr. Rich Ellis at Mayo. The BUS project is a feasibility study aiming to develop state-of-the-art breast ultrasound lesion interpretation support software for radiologists. It has three main objectives:

1. Build and train machine learning / deep learning models for classifying breast ultrasound lesions into the correct BI-RADS assessment categories.

2. Build a second algorithm that mimics systems used by experts, relying on characteristics of the lesions such as shape, orientation, margins, etc.
3. Combine the models developed in the first two objectives into a single system that outperforms each of the previous models.

The models fine-tuned in this study create lesion mask predictions which, as part of the second objective, can be used by other models to make further predictions about lesions, based on characteristics such as shape. The mask predictions could be used to offer hints to radiologists while they are scanning tissue for potentially malignant lesions. The models fined-tuned during this study also classify lesions, when possible, into broad categories such as benign or malignant, which may provide further hints to the radiologists regarding the nature of the lesions they are exploring.

Definitions of Terms

BI-RADS: A classification system that provides an approximate risk of malignancy to a lesion from essentially zero to greater than 95%. The categorization and final assessment decreased ambiguity in recommendations. BI-RADS was built to be fluid and change with the adaptation of new techniques and research. Such changes that have occurred are the inclusion of lexicons for ultrasound in 2003 and MRI in 2006. The latest edition is BI-RADS 5 (2013) and included six classifications for lesions - Magny et al. (2022).

CNN: Convolutional Neural Network. A type of artificial neural network consisting of multiple layers - an input layer, several convolution layers, and an output. This architecture is one of the main drivers of progress in recent years in artificial intelligence.

Convolution layer: a layer in an artificial neural network that transforms the input into an output based on repeated applications (in a sliding-window fashion) of a filter to the input. Each application of the filter generates a fragment of output - Brownlee (2020).

Chapter 2: Literature Review

Using computer algorithms to help diagnose breast cancer from ultrasound and other types of imaging is not a new idea. Chen and Hsiao in 2008 looked at machine learning techniques such as neural networks and support vector machines (SVMs) as potential ways to automate parts of the diagnostic process. They've compared textural analysis with morphology analysis (the analysis of the shape of the lesion), and noted that, while most studies try to classify tumors in two major classes - benign vs. malignant - breast cancer is heterogeneous, with significant overlap between classes.

More recently, deep learning has become the topic of research in the field of image analysis. Convolutional neural networks (CNNs) have been shown (Szegedy et al., 2015) to perform well as image classifiers. This literature review focuses on CNNs as a general approach, and also on particular implementations of the CNN architecture, along with newer architectures such as transformers.

Convolutional Neural Networks

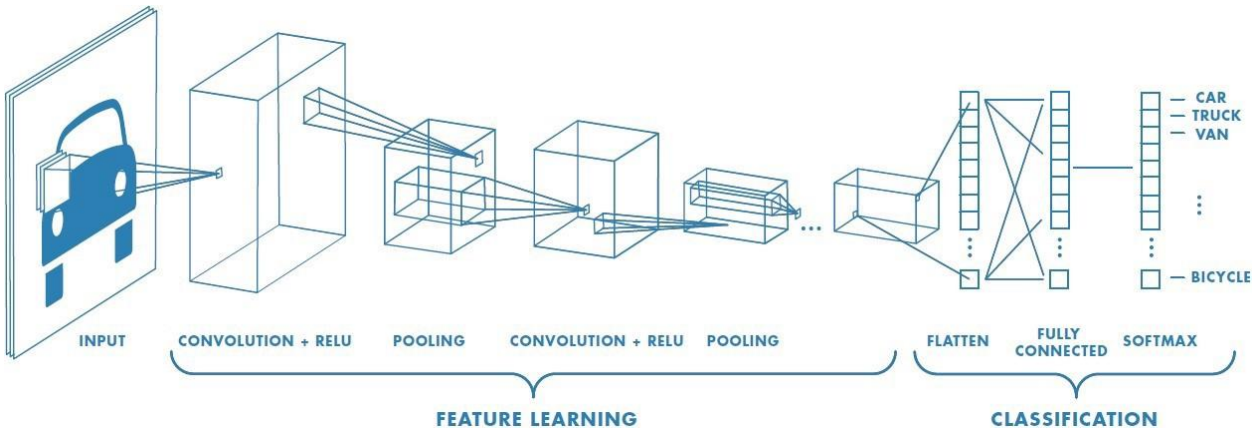
LeCun et al. (1989) proposed convolutional neural networks as a way to perform automated recognition of hand-written zip code digits. They've shown this architecture can generalize from particular examples on complex tasks, while also keeping the number of free parameters close to a reasonable minimum without affecting performance. This network was the prototype of what was later known as LeNet. This seminal paper was followed nearly a decade later (LeCun et al., 1998) by LeNet5, an improved network architecture, shown to outperform all handwritten digit recognition techniques at the time. Practical applications were also proposed, such as systems for recognizing handwritten characters online, and models that could read millions of checks per day.

Fundamentally, a CNN is a neural network consisting of a cascade of layers of neurons (see Figure 1 below), each layer performing functions such as convolution, pooling, softmax, etc. Convolution is the central idea here - it is, in essence, a way of summarizing the relevant

data provided by the previous layers, and pass on to the next layers only the most relevant features - in the process, and together with pooling, it gets rid of noise and less relevant features (Goodfellow et al., 2016).

Figure 1

Convolutional Neural Networks (Saha, 2018)



Generally speaking, a CNN tends to perform convolution and pooling several times on the initial data. In a sense, the data is "distilled down" this way to its essential features. Once the data has been summarized, the final layers in the CNN may perform other tasks such as classification.

Due to the way they summarize data, CNNs tend to perform well on highly multidimensional data such as computer vision datasets. That type of capability is needed for medical imaging, including tasks such as the detection of organs and body parts, cell detection, and image segmentation (Shen et al., 2017).

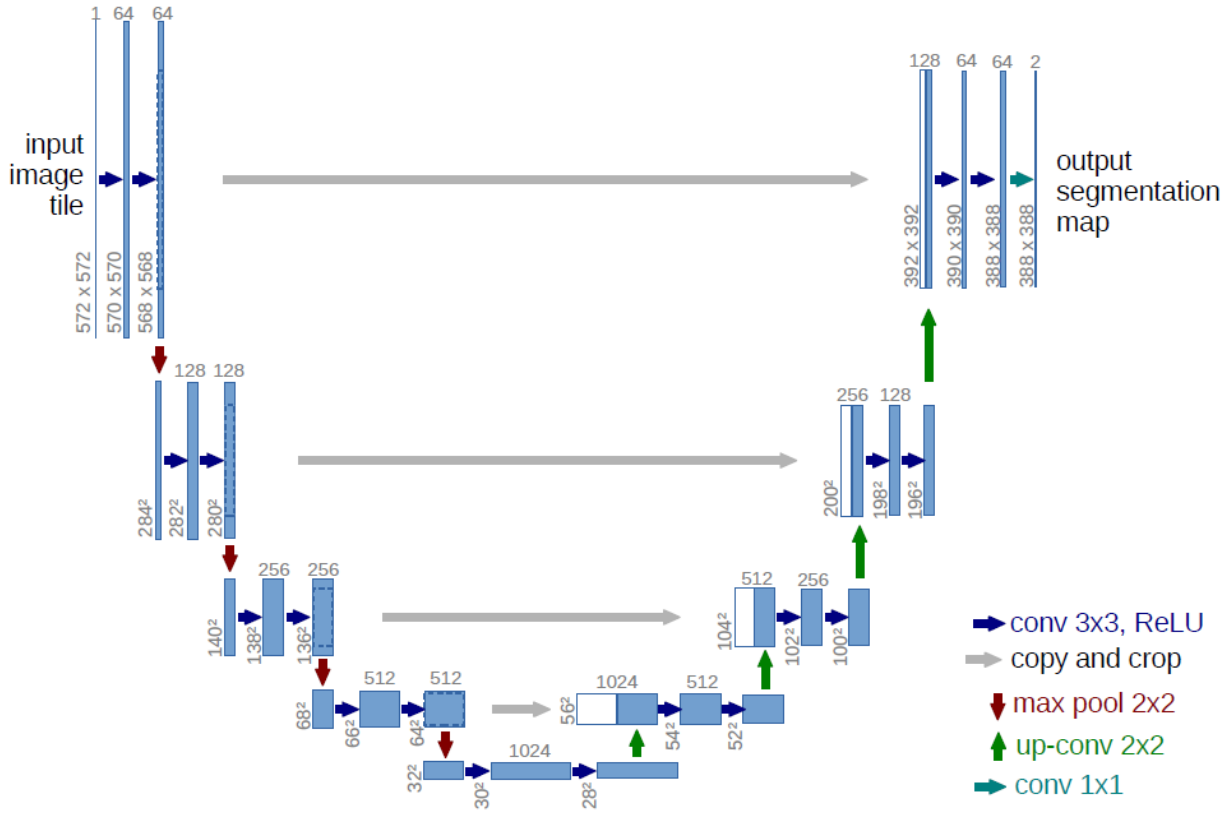
U-Net

Specifically related to image segmentation in a biomedical context, U-Net is a refinement of the CNN concept (Ronneberger et al., 2015). Image segmentation is complex, it involves making a summary of the initial data (just like CNNs in general do), a classification of individual pixels, and then the construction of masks that correspond to the various objects recognized in the image. The initial data is reduced in size; after that, a relatively complex output, which itself

is an image, is reconstructed from it. This summary-followed-by-reconstruction sequence is reflected in the U-Net architecture (see Figure 2), where the later layers in the network have direct access to some of the early image features, in addition to the distilled data generated at the bottom of the network diagram.

Figure 2

U-Net (Ronneberger et al., 2015)



U-Net has been shown to perform well for segmentation tasks on cell images. Closer to the main topic of this project, variations and expansions of the original U-Net concept have been studied in the context of segmentation of breast ultrasound images (Guo et al., 2021). For these reasons, U-Net is one of the deep learning architectures explored in this project.

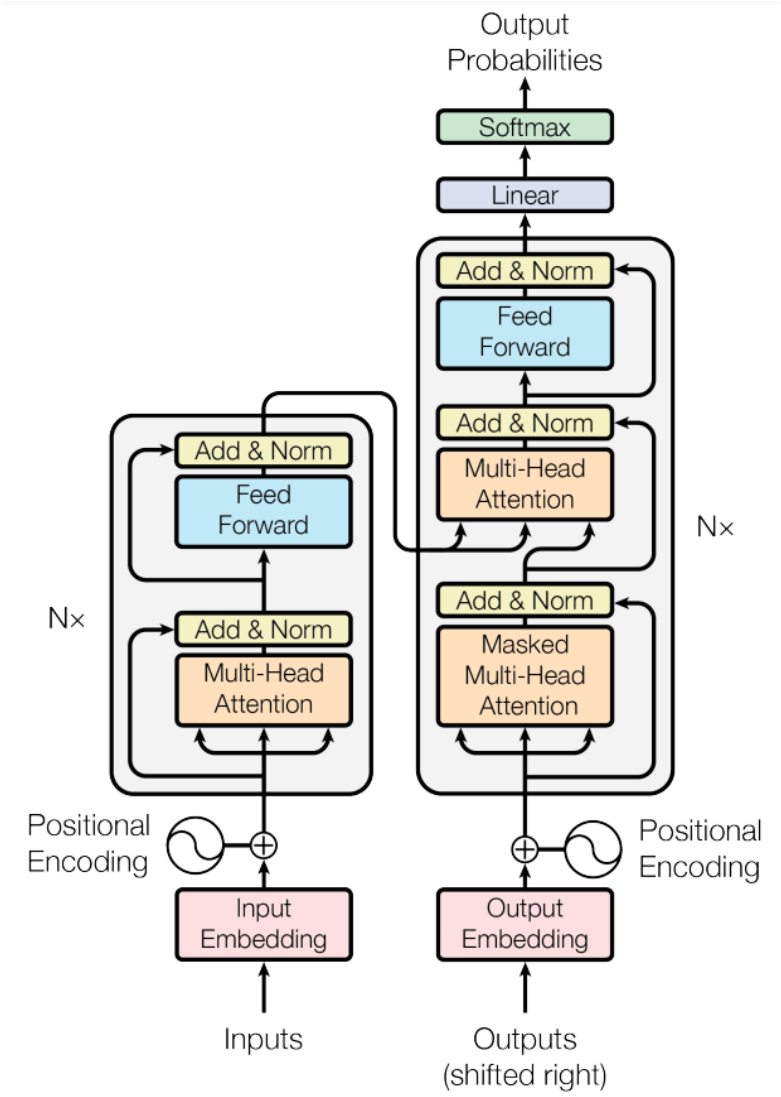
Transformers

Traditional neural networks typically deal with a single data point at a time. In order to handle sequences of data, the long short-term memory (LSTM) architecture was proposed by Hochreiter and Schmidhuber in 1997. LSTMs are able to consider data points in the context of the sequence where they appear. They are therefore well suited for tasks such as speech recognition and machine translation. They have the equivalent of a memory where data is processed in successive steps.

LSTMs use recursion, which is computationally expensive. They also process data sequentially (one element at a time). To avoid these disadvantages, an evolution of the LSTM concept, called transformers was introduced in the seminal paper "Attention Is All You Need" (Vaswani et al., 2017). Figure 3 illustrates a diagram of the central idea behind the transformer architecture.

Figure 3

Transformer (Vaswani et al., 2017)



Transformers are able to consider the various elements of the data in the context where they occur - the entire input to the network is processed at once. The parallel nature of the process allows for easier parallel training, which reduces the training time and allows for the development of very large networks. While transformers were initially created for natural language processing (NLP) tasks, vision transformers (ViT) introduced this architecture to image processing tasks as well (Dosovitskiy et al., 2021). Language models and vision models based

on the transformer architecture can be trained for a variety of tasks, such as image generation based on a text prompt.

Recent research has suggested (Matsoukas et al., 2021) that transformers in general, and ViTs in particular, may well outperform CNNs for medical imaging tasks, especially when pretrained on large image datasets such as ImageNet. A bonus is the attention / saliency maps that ViTs may provide for free, which aid with the interpretability of the model.

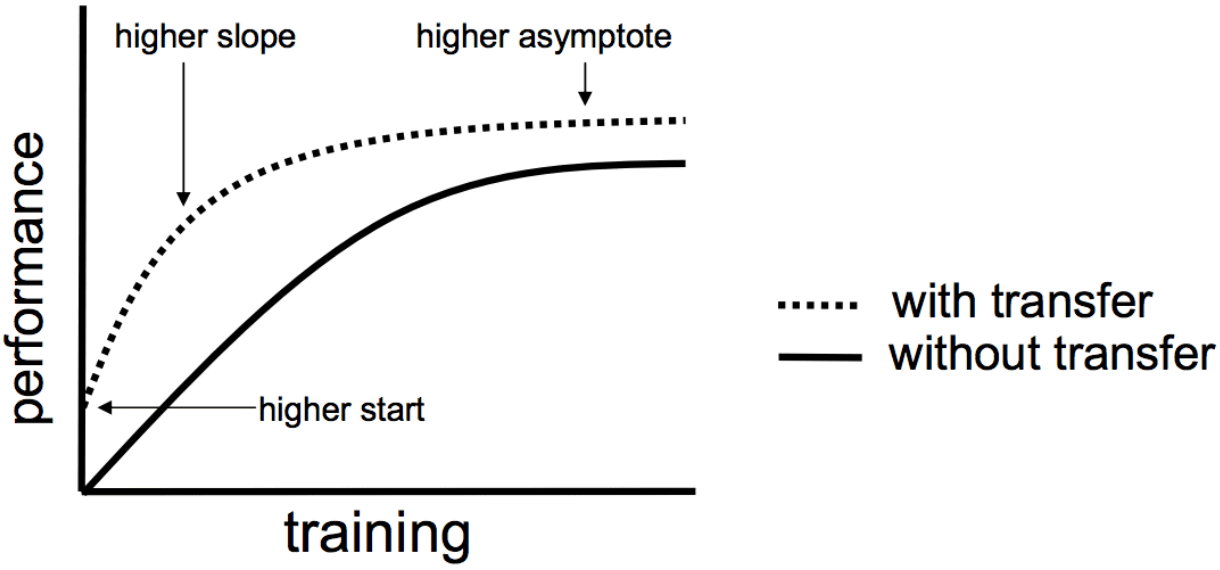
A recent evolution of the transformer concept, called SegFormer, has been shown (Xie et al., 2021) to deliver state-of-the-art performance metrics (mIoU - mean intersection-over-union) on semantic segmentation datasets such as ADE20K, Cityscapes and COCO. For these reasons, SegFormer is another deep learning architecture explored in this project.

Transfer Learning

The idea of transfer learning for neural networks was proposed for the first time decades ago (Bozinovski, 1976). In essence, a model is initially trained on one dataset, after which training continues for a somewhat different task using a different dataset. For example, the model could initially be trained on a dataset containing images of cars, and then training continues on a dataset containing images of trucks. This way, the model can be trained more quickly on the second dataset, and it will perform better (see Figure 4). Some of the training on the first dataset has been "transferred" to the second dataset, and the model performance benefits from it.

Figure 4

Transfer learning (Brownlee, 2019)



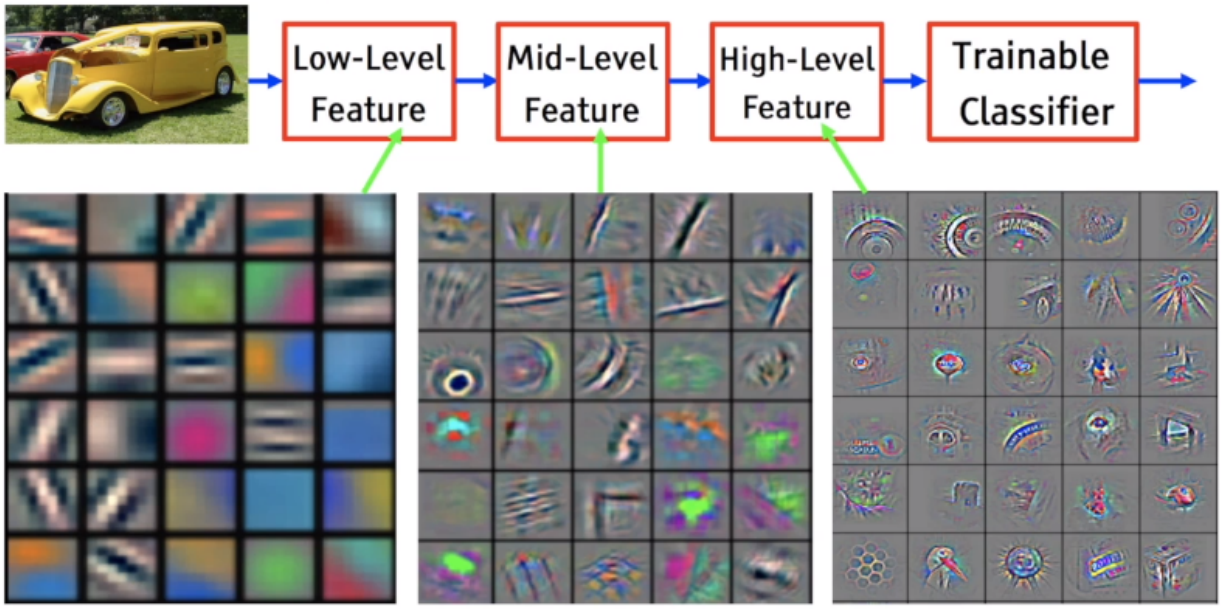
The intuition behind the process is this: if the datasets are not completely dissimilar, some of the features of the objects in the first dataset remain partially stored in the model, and are built upon while continuing training on the second dataset.

Specifically, in the case of computer vision models, a popular strategy for using transfer learning consists of training models initially on a very large, comprehensive dataset such as ImageNet, which contains many different classes of objects with many examples each. After that, the model is fine tuned on a more specific dataset for a more specific visual task. This has been shown to improve the performance of CNN models, when compared to models that do not benefit from the initial training on ImageNet (Huh et al., 2016).

Looking at how CNNs work reveals why this is possible in practice. The initial layers of the network learn basic features: straight lines, corners. The following layers build on top of that and learn simple shapes. The final layers are then able to understand more complex shapes. This hierarchical approach to data distillation and feature extraction is illustrated in Figure 5, which shows the types of features that each layer in a CNN may detect.

Figure 5

Features in CNN layers (Vignesh, 2020)



The basic features learned by the first layers are common to many objects. Most objects out there have something akin to an edge. If a model has already learned to recognize that, it will then be able to more quickly be trained to recognize other objects - which share some of their simple features with the objects used for the initial training.

In the field of medical imaging, specifically applied to breast ultrasound imaging where large datasets are rare and difficult to build (which makes training difficult), transfer learning has been shown to improve the performance of CNN models (Byra, 2021).

Data Augmentation

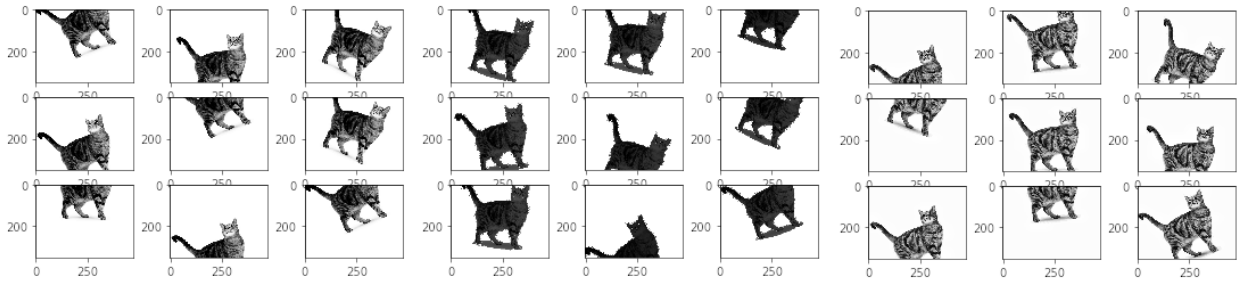
One of the difficulties in training vision models is the fact that for some specialized tasks, such as diagnosing breast cancer from ultrasound images, there is a lack of large datasets with diverse images. The datasets that exist tend to be small and repetitive.

CNNs in general, and especially when working on visual tasks, tend to not perform well if the training dataset is small and lacks diversity. To solve this problem, various data augmentation techniques can be used.

Some techniques involve purely geometric transformations of the images: rotation by various angles, flipping the image upside-down or left-right, affine transformations, cropping, random erasing. Other techniques may take inspiration from digital photography and adjust the image brightness, contrast, saturation, or may change the color contents of the image (Shorten & Khoshgoftaar, 2019). Figure 6 below illustrates examples of a few basic types of image augmentation.

Figure 6

Image Data Augmentation (Lau, 2017)



Typically, data augmentation is applied during model training, in a random fashion, using predetermined frequencies and amplitudes for each transformation. The parameters of augmentation may themselves become subject to hyperparameter optimization, in order to find the combination that delivers the best performance overall.

For medical imaging tasks, data augmentation techniques such as flips and gaussian filters have been shown to deliver validation accuracies of 84% and 88% (Hussain et al., 2017). This project uses data augmentation as an important strategy to improve the performance of the vision models.

Summary

Deep learning models such as U-Net and transformers have been shown to perform well on image classification and segmentation tasks in general, and also when applied to medical imaging. Training such models from scratch on small datasets may not be feasible, or may only produce marginal performance levels. It is imperative, given the scarcity of ultrasound imaging

data, to start with models pre-trained on large image datasets, such as ImageNet, and use transfer learning to continue fine-tuning the models on ultrasound data.

Additionally, to make sure the training process makes the most of the existing, rather scarce data, image augmentation techniques were extensively used throughout the training process. This has ensured that the project has used powerful, state-of-the-art deep learning architectures that benefit from pre-training on large datasets, which were then fine-tuned thoroughly on the existing ultrasound image datasets. The next chapter describes this process in detail.

Chapter 3: Methodology

This project aimed to build machine learning models for semantic segmentation on ultrasound images with good performance. For this task to succeed, several components of the project needed attention: the datasets used to train the models, the way data (images) was processed for training, the model architectures and parameters, the training process, and the performance evaluation. These topics are explored in detail in this chapter.

Model Architectures

Two model architectures were selected for this project: U-Net and SegFormer. They are described in some detail in Chapter 2.

U-Net was one of the first architectures known to provide good performance for segmentation tasks, especially in the field of biology and medicine. It was proposed by Ronneberger et al. in 2015, and was considered a state-of-the-art segmentation architecture for some years after its release. It was chosen for this project as a safe, known-good model.

SegFormer is a much newer model, proposed by Xie et al. in 2021, as a variation of the generic transformer architecture. Transformers in general are known to perform well on segmentation tasks, and Matsoukas et al. (2021) have shown transformers to perform well on medical imaging datasets. SegFormer was shown by its authors to deliver state-of-the-art performance for image segmentation on generic image datasets. But whether SegFormer in particular would perform well for image segmentation on ultrasound imaging datasets was not known. For this project, SegFormer was chosen as a potential alternative to U-Net, or as a potential model complementary to U-Net.

Datasets

The datasets used for this project are sets of breast ultrasound images, along with their labels. One of the main challenges when training vision models is the large amounts of data needed for training - the number of images used for training needs to be a large value, ideally thousands of images or more. For common tasks such as the segmentation of images of city

streets, the amount of labeled data available is usually large enough, but fully labeled medical ultrasound images are scarce. This was seen as a potential obstacle at the beginning of the project.

Several ultrasound imaging datasets were collected by the parent project (the BUS project - see Chapter 1) before the segmentation work had started. Four of these datasets (Mayo, BUS Dataset B, BUSIS, Dataset BUSI with GT) are fully labeled - they have lesion masks and the lesions are labeled with classes. One of them (BUV Dataset) is much larger, but is only partially labeled - it has class labels, but no masks, so it cannot be used for segmentation training. See Table 1 for details.

Table 1

Ultrasound imaging datasets

Name	**Has Classes**	**Has Masks**	**# of Images**	**Fully Labeled**
Mayo	yes	yes	289	yes
BUS Dataset B	yes	yes	163	yes
BUSIS	yes	yes	562	yes
Dataset BUSI with GT	yes	yes	780	yes
BUV Dataset	yes	no	25272	no

The original intent, at the start of the BUS project, was to pick one of the four fully labeled datasets and train the models using that dataset alone. The small sizes of the fully labeled datasets raised concerns about the performance of models trained on insufficient data. In fact, early attempts within the BUS project, prior to this study, indicated that training a model on just one dataset may not provide satisfactory segmentation results.

Therefore, a different approach was devised for this project: the dataloaders used by the models were written in such a way as to concatenate all four fully labeled datasets (Mayo, BUS

Dataset B, BUSIS, Dataset BUSI with GT) and present them to the models as a single, uniform, fully labeled dataset, used for training and validation, containing a total of 1794 images. This number of images was shown to be sufficient for good performance in segmentation tasks – see Chapter 4 for results.

One challenge was the fact that the datasets were heterogeneous. The image sizes on any dimension vary between approximately 200 and 2000 pixels. The image formats are diverse (PNG, BMP). The number of bits per pixel varies between 8 and 24 (or as low as 1 bit per pixel for some mask frames). A library of Python functions was written for this project which transforms the data into a uniform format: all images are represented in standard sizes (224x224 for U-Net, 512x512 for SegFormer), at 8 bits per pixel, before the models ingest them. From the point of view of the model, it is trained on a single, unified dataset consisting of 1794 images with uniform parameters.

The training / validation split of the unified 1794 image dataset, for both models, was 80 / 20. To ensure that all classes and all datasets were represented fairly in both training and validation sets, the split was stratified by a tuple of both dataset and class. In other words, any combination of dataset and image class was represented proportionally in both training and validation. The classes were not perfectly balanced, but the overall imbalance was deemed small enough to not require class balancing techniques.

One dataset (Dataset BUSI with GT) had a series of images without either benign or malignant lesions; the mask frames were empty and the images showed normal tissue. These were labeled "normal" and provided a baseline of normalcy for the models to learn - essentially showing the models what an image looks like if there is no lesion in it, which is a situation that can definitely occur in practice. Table 2 shows the composition of each dataset based on classes - how many images belong to each class.

Table 2

Image counts for classes in the original datasets

Name	# of benign labels	# of malignant labels	# of normal labels
Mayo	150	139	0
BUS Dataset B	109	54	0
BUSIS	306	256	0
Dataset BUSI with GT	437	210	133

Finally, the BUV Dataset, made of more than 25,000 images, which only had class labels, but no masks, could not be used to train models for segmentation tasks. Instead, it was used to assess the consistency of the predictions of the models. In other words, for images labeled as benign, the ratio between the number of predicted benign pixels, and the total number of predicted pixels, indicates how consistently the model makes predictions in accord to the benign image class. A similar definition applies to predictions on images in the malignant class.

The BUV Dataset was also used to sample model predictions to provide an indication for the models' behavior in the real world. A full video was created with the predictions generated on the whole BUV Dataset, combined into a single video file. The video provides a synthetic overview of the behavior of the models when operating in real time, making predictions on images taken from real patients by a radiologist. More details and actual results are included in Chapter 4.

Image Augmentations

Even with the relatively large number of images available in the single, unified dataset, there was still the concern that the models may overfit before they achieved the desired level of performance. A common technique to improve performance and avoid overfitting the vision models is image augmentations. As explained in more detail in the previous chapter,

augmentations are a series of geometric and pixel value transforms applied to images and masks during training, with the goal of "diversifying" the images while retaining the information contained in the original dataset.

Since two different models were trained and compared for this project, a desirable goal is to compare them fairly - i.e. train both models using the exact same data and augmentations. Due to time constraints, augmentations were implemented in slightly different ways for each model, but using similar lists of transforms, and similar limits for the random variations of each, and similar probabilities. Also, the models exhibited different responses to augmentation, with SegFormer appearing to tolerate higher augmentation limits - this is another reason for the different augmentation parameters used in training.

Additionally, in order to accelerate the coding pace, different libraries were used to perform image augmentations for each model. U-Net was created and trained using the fast.ai framework, which also provided the augmentation methods. For SegFormer, the popular Albumentations library was used instead.

For both models, images were normalized before being used for training - the central pixel values and the spread of the pixel values were adjusted to prevent very large variations from having a negative impact on performance. This technique is very commonly used with vision models.

In other words, similar augmentation techniques were used for both models, but code implementations used for each were different, and some of the parameters for each transform were different. Table 3 provides an overview of the augmentations used in training, and the transform limits (maximum values) enforced for each model.

Table 3

Augmentation techniques and maximum random values

	U-Net		SegFormer	
	amplitude	**probability**	**amplitude**	**probability**
Brightness	0.2	0.75	0.5	0.75
Contrast	0.2	0.75	0.5	0.75
Horizontal Flip	-	0.5	-	0.5
Rotate	30	0.75	30	0.75
Scale	0.1	0.75	0.2	0.75
Translate	-	-	0.2	0.75
Shear	-	-	30	0.75

The augmentation techniques used in this project are widely used in most image segmentation tasks. But for breast ultrasound imaging in particular, there are several aspects to keep in mind. Horizontal flip can be used without problems, and it was used in this project, since it only deals with left-right laterality, and lesions could be scanned in any direction. But vertical flip (another common augmentation technique) would not make sense for breast ultrasound imaging, since lesions appear in the ultrasound scan always in the same vertical orientation. It would be counter-productive to train the model for images that it will never see.

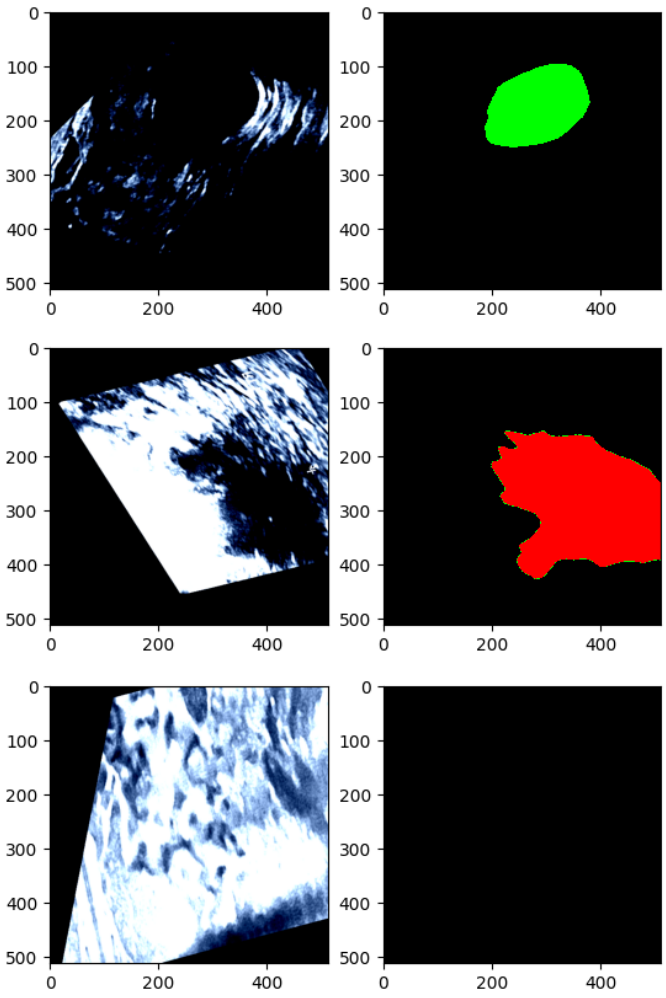
For the same reason, the maximum rotation angles used in this project are quite moderate (30 degrees), compared to generic image segmentation tasks. Since the vertical axis of any lesion in a scan image tends to stay the same, it would not make sense to rotate the images in training too much.

Brightness and contrast in particular are desirable when training a model on ultrasound images, since these images typically show a wide range of pixel values. It is beneficial for the performance of a model to perform training on images ranging from very dark to very bright.

Figure 7 below shows a few samples of images used to train the SegFormer. The images are augmented and normalized. The normalization explains the dark, high contrast aspect of the images. The masks for each image are color-coded by class: green for benign, red for malignant. One image is in the "normal" class (no lesion) and the mask is empty. Please note that the consistent 90 degree rotation to the left is an artifact of the visualization code which was not corrected due to time constraints; the images were not actually rotated that way in training.

Figure 7

Image augmentation samples from SegFormer training



Training the Models

As mentioned in Chapter 2, building models from scratch and training them exclusively on a single small image dataset may not provide good performance. Models need large amounts of imaging data to learn basic image features such as edges, corners, etc. Commonly, models are built and pre-trained on very large public image datasets. This provides a solid foundation for fine-tuning them on the actual datasets used in the study, using transfer learning. For this project, we've started with pre-trained models, and fine-tuned them on the ultrasound imaging dataset.

U-Net, with a ResNet34 backbone, was imported from torchvision.models.resnet, via the models.resnet34 method in the fast.ai libraries. This is similar to the architecture proposed by He et al., 2015, pre-trained on ImageNet.

SegFormer, in the MiT-b3 model variant, was imported from the HuggingFace repository via the SegformerForSemanticSegmentation.from_pretrained() method in the transformers library (also part of HuggingFace). The base model, as provided by HuggingFace, is pre-trained on ImageNet. The starting point for the transfer learning code was the method described by Cornille and Rogge, 2022.

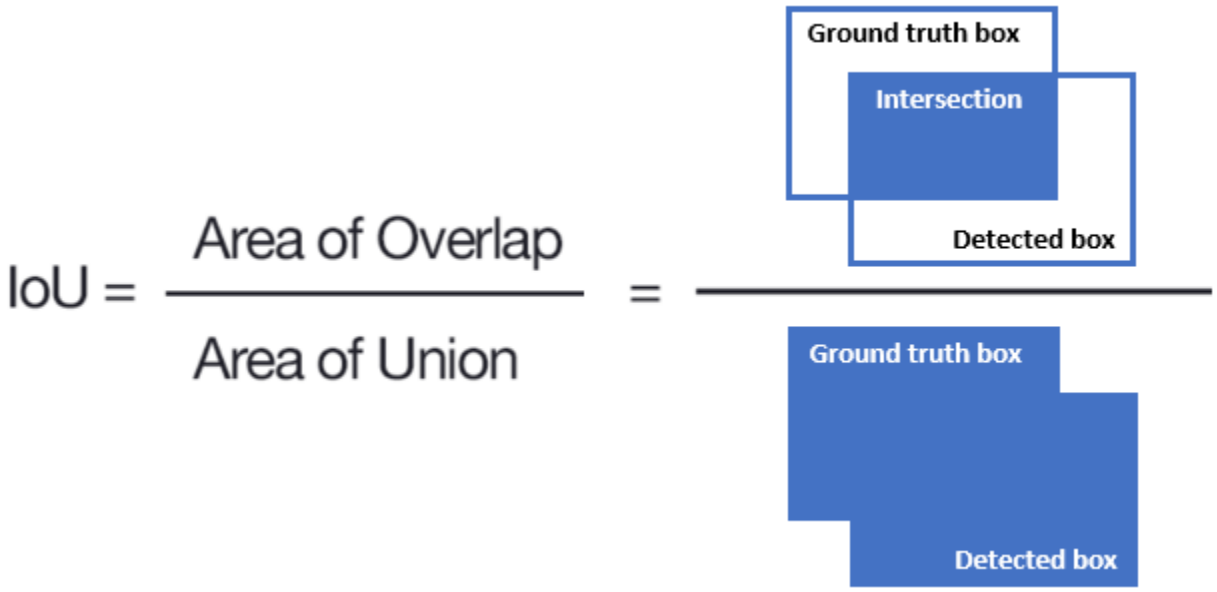
While U-Net was trained using the plain Adam optimizer, SegFormer notably used a variant called AdamW, proposed by Loshchilov and Hutter in 2017, which decouples the weight decay from the optimization steps taken with regard to the loss function.

Intersection-over-Union

To judge the performance of each model, the metric that was used for segmentation is intersection-over-union (IoU), also known as the Jaccard index, which is the standard performance metric for segmentation tasks. It shows how good the overlap is between a ground truth mask and a predicted (a.k.a. "detected") mask - the more the two masks overlap in the image, and the less they do not overlap, the better the fit between prediction and ground truth. IoU is the ratio between the area of intersection, and the area of union, of two sets of pixels. Figure 8 provides a visual intuition of the concept.

Figure 8

Intersection-over-union (Baeldung, 2022)



In terms of actual code implementation, HuggingFace provides a library called evaluate, which includes an implementation of multiclass (semantic) IoU, which was used to evaluate SegFormer. For U-Net, the fast.ai ecosystem does provide an IoU calculator function, but it operates only in single-class mode. For this project, we've modified the fast.ai function to work for semantic segmentation (multiclass), and the modified version is included in the project's code repository.

Selecting Hyperparameters for Optimization

A deep learning model has many parameters. It is common for such models to have millions or even billions of parameters. But not all parameters are equally important. Some parameters are, in a sense, of overarching importance, since they describe the model globally. Changing one of these parameters affects the whole model. They are known as hyperparameters. An example is the number of layers in a convolutional neural network: a network with 10 layers has a value of 10 for that hyperparameter.

Finding the best values for these parameters is a crucial part of model training and optimization, and it is known as hyperparameter optimization, which essentially involves training the model with different hyperparameter values while measuring performance, and is described in detail in the next section in this chapter. But first, the right hyperparameters had to be selected for optimization.

Depending on the model architecture (U-Net, SegFormer) and the specific task they are trained for (segmentation), some hyperparameters are expected to have a large impact on performance, while others do not typically influence performance in an appreciable way. It is important to optimize only the most influential parameters, or else the optimization may take a very long time. For the other hyperparameters, reasonable default values could be set and used throughout the search. As shown in Chapter 4, the search process itself may reveal that even parameters thought to be influential may not actually have a substantial impact on performance for the given task and dataset. Choosing the right hyperparameters for optimization is based on previous experience with deep learning models, and also on initial sampling of performance metrics (manual trials). Table 4 shows the parameters chosen for the optimization process for each model in this project. Each hyperparameter is described in detail after the table.

Table 4

Hyperparameters used in the optimization process

U-Net	**SegFormer**
base learning rate	base learning rate
learning rate multiplier	warmup ratio
freeze epochs	weight decay

The learning rate represents the amount of change applied to the model weights after each training batch. At the end of the batch, the model's error is estimated, and the model weights are changed such that the error is reduced, and the process is then repeated for each

batch, always looking to minimize the error. In technical terms, this process involves gradient descent (searching for the minimum of some function, such as the error) and backpropagation (changing the model's weights based on the error at the output layer), and it is the core of standard training algorithms for deep learning models.

The amount of change for the model weights needs to be carefully calibrated since not enough change would prevent the model from learning at all, while with too much change the training process would become chaotic and may never succeed in minimizing the error. The learning rate controls the amount of change. It is generally considered the most important hyperparameter, regardless of the model architecture. The base learning rate is either the initial or the highest value of the learning rate while training.

The learning rate may vary during training. For U-Net, it decreases continuously from an initial high value. The end value of the learning rate is obtained by dividing the base learning rate by the learning rate multiplier. In other words, the learning rate multiplier for U-Net is the ratio between the initial and the final learning rates.

SegFormer may not always start at the highest learning rate. A "warmup" time could be defined, during which the learning rate increases quickly from zero to the base learning rate. After this brief period, the learning rate decreases gradually to zero. The ratio between the duration of the warmup period, and the total training duration, is the warmup ratio.

With U-Net, a commonly used technique consists of freezing (preventing change in) all layers except the last, then performing model training for one or a few epochs (so only the last layer will be trained during these epochs), then unfreezing the model for the rest of the training. The "freeze epochs" hyperparameter controls the duration of this procedure.

Weight decay is a common regularization technique used to control overfitting the model while training. It essentially penalizes model weights with very large values. The hyperparameter called weight decay controls the penalty applied to large weights, and it was selected for the optimization loop for the SegFormer model.

In total, only three hyperparameters were selected for optimization for each model. This may seem to be a small number. But training a neural network is a complex process. Since each training round may have many epochs, and the models are relatively large, each training round could take a long time. For this project, fully training one model on an RTX 3090 GPU took close to 1 hour on average, which is a very substantial duration if training needs to be repeated many times during optimization.

This has important consequences when choosing which hyperparameters to optimize. With very short training times (seconds), most if not all hyperparameters could be included in the optimization process. This is not feasible with vision models which have long training durations, where the total optimization time would become prohibitive if a large number of parameters are optimized.

Hyperparameter Optimization

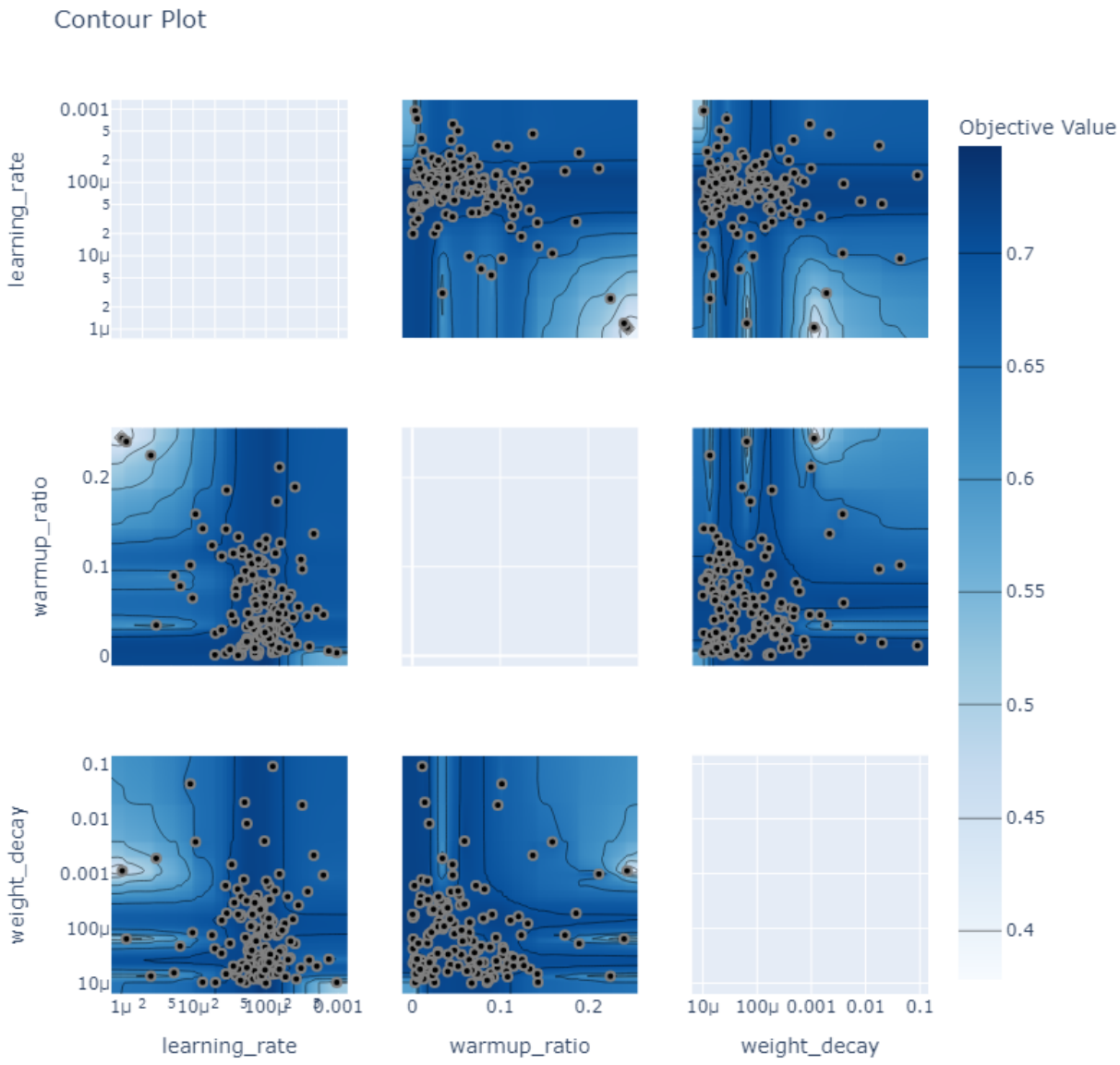
Finding the best values for the model hyperparameters, with such long durations for training, can become a costly process if it's done by trial and error. This analysis cannot be done manually. To systematize the process of finding the best hyperparameter values, this project used the Optuna library (Akiba et al., 2019), which is widely used in deep learning projects for hyperparameter optimization. Of the several samplers available with Optuna, this project used TPESampler (Tree-structured Parzen Estimator) which is the default Optuna sampler for single-objective studies. (Bergstra et al., 2011)

Optuna works by sampling the hyperparameter space for combinations of values, evaluating the performance (IoU in this project) for each combination, and inferring further points of interest in that space based on previous performance values. Given enough trials, it tends to converge towards combinations of hyperparameters that offer good performance. The Optuna documentation suggests a minimum of 100 trials for best results, or up to 1000 trials. This project has cleared the 100 trials bar by a margin of about 50% for each model (see Chapter 4 for details).

As an example, Figure 9 shows an overview of the entire Optuna search for the SegFormer model. The nine diagrams (only six actually used) are projections of the hyperparameter space on different dimensions. Each dot represents a particular trial (a model training session) that explored a particular combination of hyperparameters, and provided an IoU value to the optimization algorithm. Actual results will be shown in Chapter 4, this figure is intended to provide just an overview of the process.

Figure 9

Optuna study, optimizing SegFormer



Single Class Segmentation

One of the secondary goals of the project was to produce a model that could perform simple mask predictions that only consider the shapes of the lesions while ignoring the class

(benign or malignant). These predictions could be used by other projects, within an ensemble method, as input to other models which could predict different aspects of the lesions.

One approach to solving this problem would be to train a completely different model from scratch, with a separate hyperparameter optimization loop. Due to time constraints, this method was discarded. Instead, using the best hyperparameters found previously for the SegFormer, a new SegFormer model was trained on a version of the 1794 image dataset that only had one class - both benign and malignant lesions were labeled the same, using a generic class called "lesion". In other words, the dataset used for training this model was the same dataset used for the main models, but the different lesion classes were reduced to a single class. Only one training session was performed, and the model performance metrics were extracted from this session.

This model was called the single-class SegFormer. Think of it as a model looking for lesions regardless of their nature (benign or malignant). The model is called "single class" because the only "class" it is looking for is the lesion itself. Its performance metrics are shown in the next chapter.

Summary

Two model architectures were selected for this project: a safe model known to work well for segmenting breast ultrasound images (U-Net), and a transformer model (SegFormer) known to work very well on generic image datasets, but untested for the particular task of segmenting breast ultrasound images.

The project used a variety of datasets to train and evaluate the models. To maximize training efficiency, several fully labeled datasets were combined into a single, relatively large fully labeled dataset. Standard image augmentations were applied to the training dataset, to reduce overfitting. The augmentation techniques, while not novel, were adapted to the characteristics of breast ultrasound images.

The main metric used to assess the performance of models was intersection-over-union, or IoU. This is very commonly used with segmentation models. Where existing IoU functions did not provide the features needed by this project, they were modified to provide the necessary features, and were included in the project's code library.

Training the models involved the standard gradient descent technique with backpropagation. To maximize model performance, several hyperparameters were chosen for each model, to be included in a hyperparameter search, using the Optuna library. Many training sessions were performed this way, with Optuna looking for the best combination of hyperparameters that yields the best IoU. Details and final results of the IoU performance numbers are shown in Chapter 4.

Prediction consistency for both U-Net and SegFormer was evaluated on a single large dataset (BUV Dataset) containing incompletely labeled images - Chapter 4 shows the results of this analysis. A video file was generated as an artifact, containing predictions on the BUV Dataset, mimicking the behavior of the models performing predictions in real time.

A separate SegFormer model was trained once on a variation of the training dataset that discarded class differences (benign vs. malignant). This was called the single-class SegFormer; while not of immediate use for this project, it could be used to provide predictions for other models created within the parent BUS project.

The outcomes of this methodology are shown in the next chapter.

Chapter 4: Findings and Results

As mentioned in the previous chapter, this project aimed to build segmentation models with good performance on breast ultrasound images. Since IoU (intersection-over-union) is one of the main measures of performance for segmentation models, that was the main metric tracked during hyperparameter optimization. It is important that the models correctly localize lesions (in plain terms: "here is an area of interest"), and IoU provides a measure of this ability.

A secondary goal was the ability of the models to correctly classify malignant lesions as malignant, and benign lesions as benign, and be consistent in this classification. Simplifying the terms somewhat, this is like maximizing true positives and true negatives. Essentially, malignant lesions must be correctly identified as malignant - this means reducing the false negative predictions, and preferring models that have the equivalent of a high recall. Since these models perform pixel-level predictions, not case-level, the analogy with classic terms such as precision and recall is not perfect.

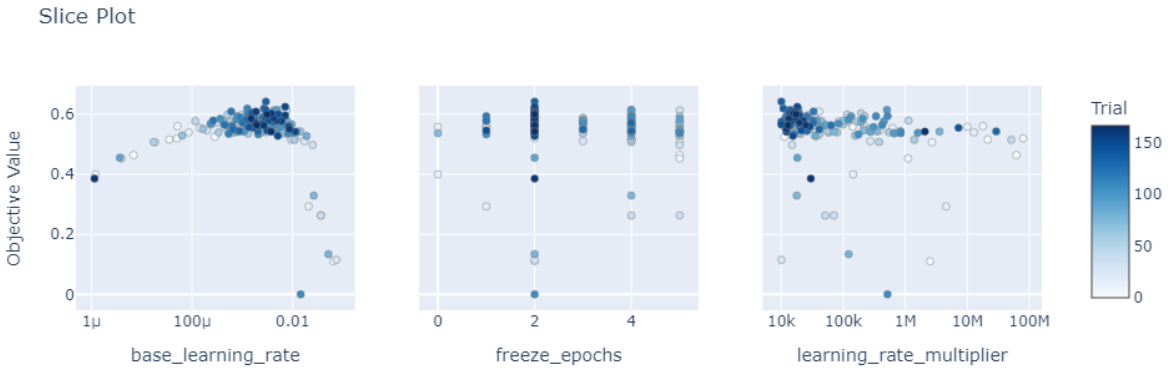
Best Hyperparameters

Since hyperparameters are so important for the model's performance, and finding their best values is a complex process, by far the largest amount of compute time for this project was spent on hyperparameter search - the methodology used for this search is described in some detail in Chapter 3.

Figure 10 shows the Optuna slice plot for U-Net, with the U-Net hyperparameters and the IoU values used for each combination. The Y axis is IoU on the validation dataset; the X axis for each panel is a different hyperparameter. The color of each dot represents the trial number (darker dots are later trials). Each trial is a complete training session, producing a fully trained model with different parameters; a total of 166 trials were performed for U-Net, with Optuna generating various combinations of hyperparameters for each.

Figure 10

U-Net IoU, dependency on hyperparameters



It is clear that the objective value (IoU) varies significantly with the learning rate and its correlation with the learning rate is strong, while the correlation of IoU is much weaker with the other two hyperparameters. This is confirmed in Figure 11, which is the Optuna estimate for the importance of each hyperparameter for U-Net. Essentially, the fixed default value of the learning rate multiplier would have worked well enough, and the time needed for hyperparameter optimization could have been substantially reduced by eliminating the learning rate multiplier from the search.

Figure 11

U-Net Hyperparameter Importance

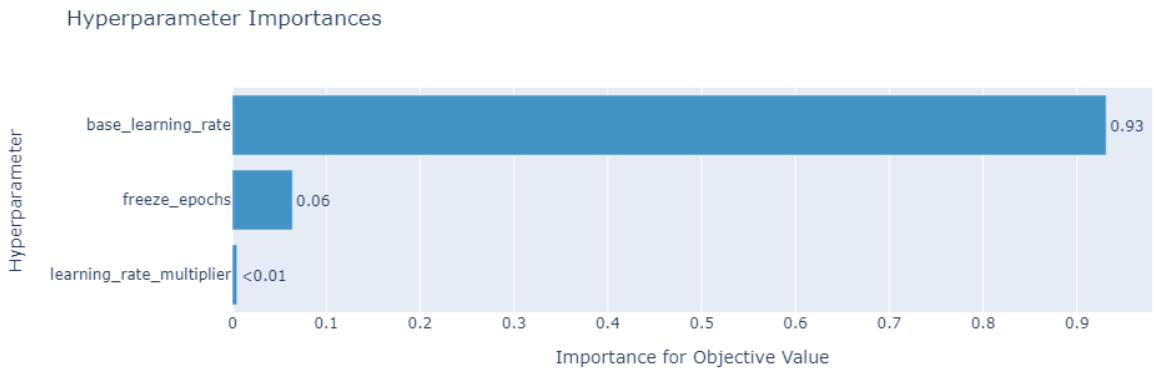
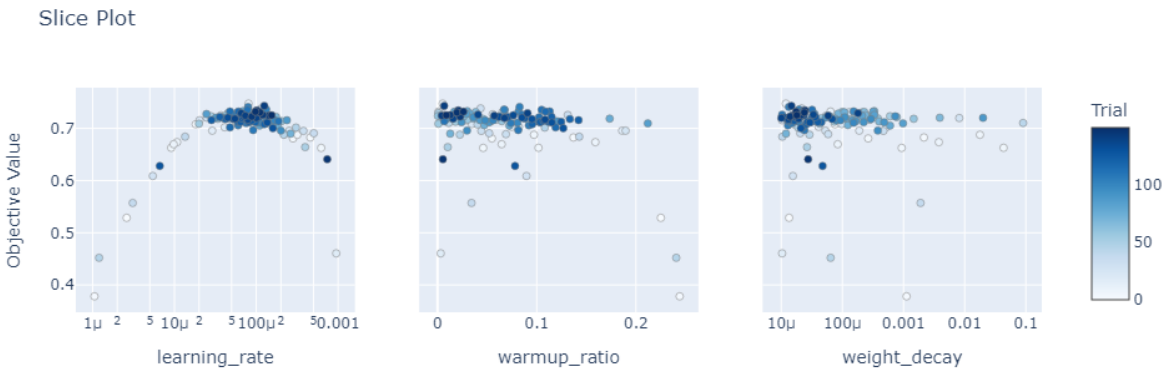


Figure 12 shows the Optuna slice plot for the SegFormer. A total of 149 trials were performed with the SegFormer.

Figure 12

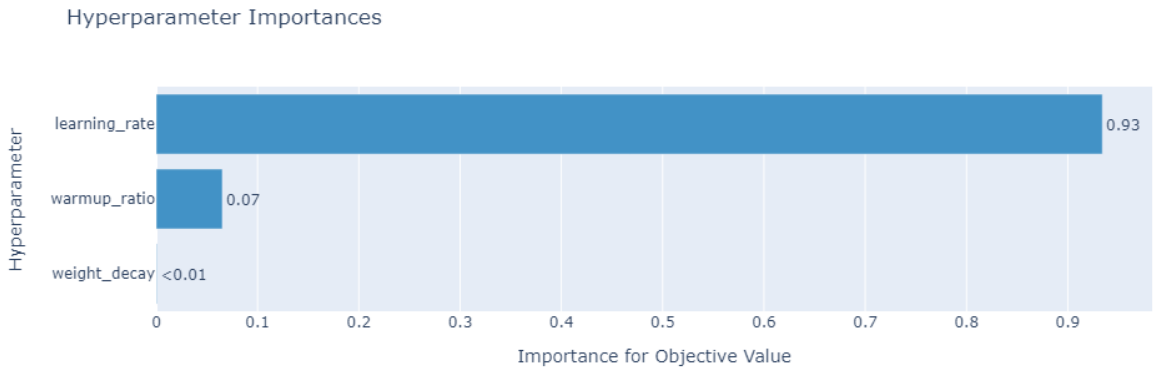
SegFormer IoU, dependency on hyperparameters



Again, the learning rate was determined to be the most important hyperparameter by far. To quantify that statement, the Optuna estimate for the hyperparameter importance for SegFormer is shown in Figure 13. Weight decay was estimated to be almost irrelevant, while the warmup ratio appears to have a mild effect on IoU performance.

Figure 13

SegFormer Hyperparameter Importance



These results are summarized in Table 5, which shows the best values found by the optimization process for the hyperparameters of each model. These values represent the hyperparameters used in training the best U-Net and the best SegFormer models. Given the datasets used in this project, and the model architectures chosen for training, these hyperparameter values are most likely to produce the best IoU performance.

Table 5

Best Hyperparameter Values

	U-Net	**SegFormer**
learning rate	2.95e-3	8.036e-5
freeze epochs	2	-
learning rate multiplier	10067	-
warm-up ratio	-	5.257e-3
weight decay	-	1.19e-5

All hyperparameter values are unsurprising, they look typical for segmentation models - but the possible range for each parameter is very wide, spanning orders of magnitude, hence the need to do an actual search to find the optimal values. It is not enough to guess - a systematic search must be conducted to confirm the intuition.

Intersection-over-Union Results

Table 6 shows the IoU performance results obtained when models were trained with the best hyperparameter values described above.

As explained previously in Chapter 3, one SegFormer model was trained with the best parameters from the main SegFormer but discarding all class labels (benign or malignant) and its IoU performance is also recorded below.

Table 6

Best IoU results

	U-Net	SegFormer	SegFormer single class
mean IoU	0.6419	0.7475	0.8901
benign IoU	-	0.7232	-
malignant IoU	-	0.7718	-

The best IoU produced by U-Net was 0.6419, or around 64.2%. This is the average IoU value over both classes (benign and malignant) - the training process for U-Net has not recorded separate IoU values for benign and malignant lesions, so they are missing from the table, and only the average over all classes was recorded.

64.2% seems to be a relatively low value for a segmentation model in general. Guo et al. (2021) have obtained 82.7% IoU from a U-Net model using a similar dataset, but in their case, the model was trained to perform only segmentation. In this study, the model performs segmentation while also attempting to assign classes (benign or malignant) to mask pixels, which unsurprisingly leads to lower overall performance metrics. Given all of the above, the performance of U-Net should be considered fair.

The best IoU produced by SegFormer was 0.7475, or around 74.7%, as an average over both classes (benign or malignant). From the model training logs, additional performance details were extracted. The IoU for the best model was 0.723 or around 72% for benign lesions, and 0.7718 or around 77% for malignant lesions.

An IoU score of around 75% averaged over all classes would be considered unremarkable for everyday semantic segmentation tasks, as seen with models trained on city street image datasets, which easily exceed an IoU performance of 80% (Xie et al., 2021). Given the nature of the data explored in this study, the size of the datasets, and the typically

problematic quality of ultrasound images, 75% IoU (with classification) should be considered good performance.

Both U-Net and SegFormer have achieved at least fair levels of performance in identifying the shape and the class of the lesions (see Chapter 1 for the project goals) which is what IoU indicates. SegFormer has outperformed U-Net when judged by the IoU metric. In a sense, this result is not surprising, since SegFormer contains the distilled results of 6 years of research after U-Net was released.

The single class SegFormer, free of the need to classify lesions as either benign or malignant, has obtained 0.8901 IoU, or around 89%. This is at the high end of the performance range of segmentation models trained on city street datasets - in other words, the model performs very well.

Classification Consistency on Partially Labeled Data

Fully labeled data for segmentation tasks, in specialized contexts such as medical imaging, is scarce and difficult to obtain. But partially labeled data, which may label images as simply benign or malignant without providing masks, is easier to obtain. The partial labels (class only, no masks) can still be used to characterize the performance of the model. In these cases, since the nature of the lesion for each frame is known (either benign or malignant), it is simply sufficient to notice whether predictions are consistent with the nature of the lesion. On benign frames, most predicted mask pixels ought to be benign; on malignant frames, most predicted pixels ought to be malignant. The number of correctly predicted pixels, divided by all predicted pixels, knowing the class of each frame, shows the consistency of the predictions with the known image classes.

When predicted masks are rendered as images, with the class of each pixel being color-coded (green = benign, red = malignant), a consistent model will generate predictions that will fluctuate less in terms of the predicted class / color. Higher consistency numbers indicate more stable predictions on any given video sequence. The best models will generate mask

frames with consistent colors (green for benign, red for malignant). Less consistent models will generate masks with colors that strongly fluctuate.

Table 7 shows the consistency numbers for U-Net and SegFormer, for benign as well as for malignant frames. The numbers were obtained using the best trained models, generating mask predictions on the partially labeled BUV Dataset (see Chapter 3).

Table 7

Video dataset prediction consistency, by class

	U-Net	**SegFormer**
benign frames consistency	0.3454	0.5818
malignant frames consistency	0.9050	0.8547

U-Net appears to have a bias towards generating malignant mask predictions. Its consistency is the best on the malignant frames in the video dataset - about 90.5%. If used in an ensemble method, with multiple models making different predictions, it seems like U-Net could provide good visual cues for malignancy, which is a desirable trait for such a model. Its consistency on benign frames is poor, only about 34.5%.

SegFormer is nearly as consistent on the malignant frames - around 85.5% consistency. It performs much better than U-Net on benign frames, where it shows a consistency of around 58.2%.

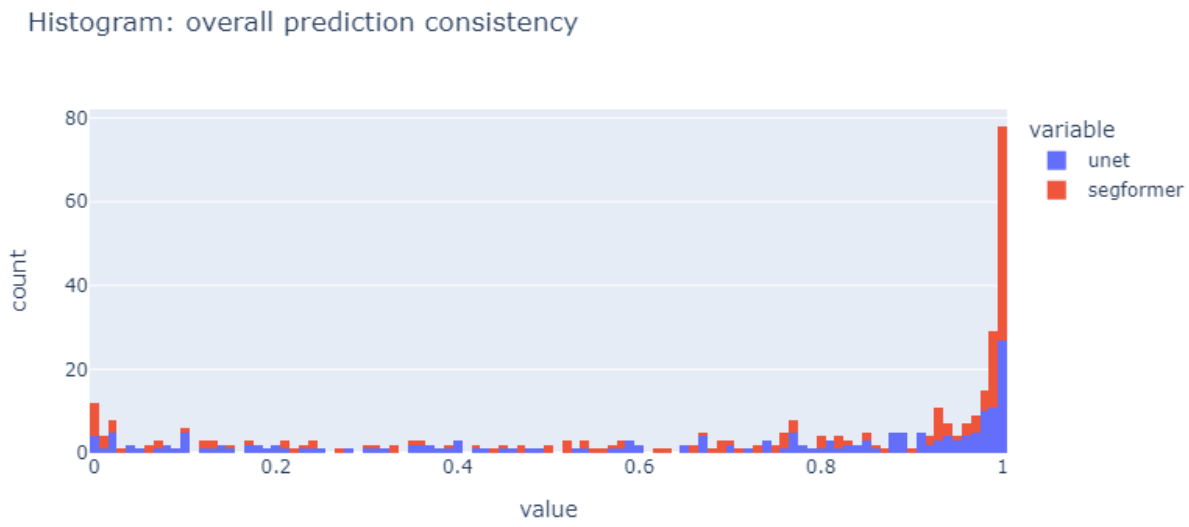
Both models show good consistency for the malignant frames. Their consistency is mediocre at best for benign frames. Given the importance of detecting malignancy in a lesion, the relatively low performance on benign frames is less important. The main fact is that both models show good consistency on malignant images, where their overall performance needs to be as high as possible.

The consistency numbers shown so far are aggregates over the whole BUV Dataset. But the BUV Dataset is made of 186 video sequences (images taken from 186 actual patients).

Prediction consistency can be calculated for each sequence, and the results can be shown as a histogram. Figure 14 shows the histogram of the prediction consistency for both models, for all video sequences (benign and malignant together).

Figure 14

Histogram of prediction consistency for both classes



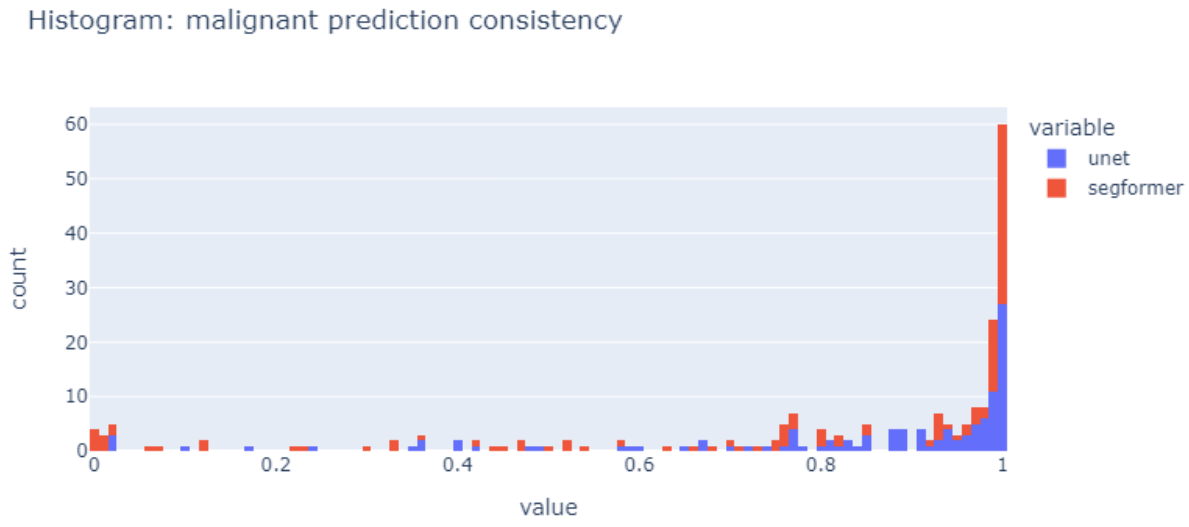
Figures 15 and 16 break down consistency histograms by the class of the lesions.

Figure 15

Histogram of prediction consistency for benign lesions

Figure 16

Histogram of prediction consistency for malignant lesions



The histograms show that both models exhibit high (over 90%) consistency on a large fraction of the sequences, with the exception of U-Net on benign sequences. For both models,

there is a small fraction of sequences where their consistency is very poor (below 10%). Such large variations in consistency suggest that frame-by-frame predictions, where models consider each frame individually for each prediction, may be inherently limited. A model that uses whole sequences of frames at once (an idea that was considered during discussions within the parent BUS project) may perform better - but that was outside the scope of this project.

Actual Predictions on the BUV Dataset

When it comes to predictions performed on a video dataset, any synthetic metric is necessarily incomplete. The complexity of the behavior of the model is hard to describe completely based on just a few numbers. It is customary to show actual image predictions when describing the behavior of vision models. For this reason, a few frames with predictions on the BUV Dataset were included below in the next few images.

The "perf" metrics at the bottom of each frame are the aggregate consistency for that particular video sequence (not just for one frame, but for the whole sequence that the frame is part of). Each video sequence is identified by the "id" string in the top-left corner of the image frame. The name of the model is stated at the top of the prediction frames.

Figure 17 shows examples of very consistent predictions. The top half shows a benign frame with its predictions from U-Net and SegFormer, the bottom half shows a malignant frame with its predictions. Predicted masks are 100% green for the benign frame, and 100% red for the malignant frame, consistent with the nature of the lesions.

Figure 17

Highly Consistent Predictions on the BUV Dataset

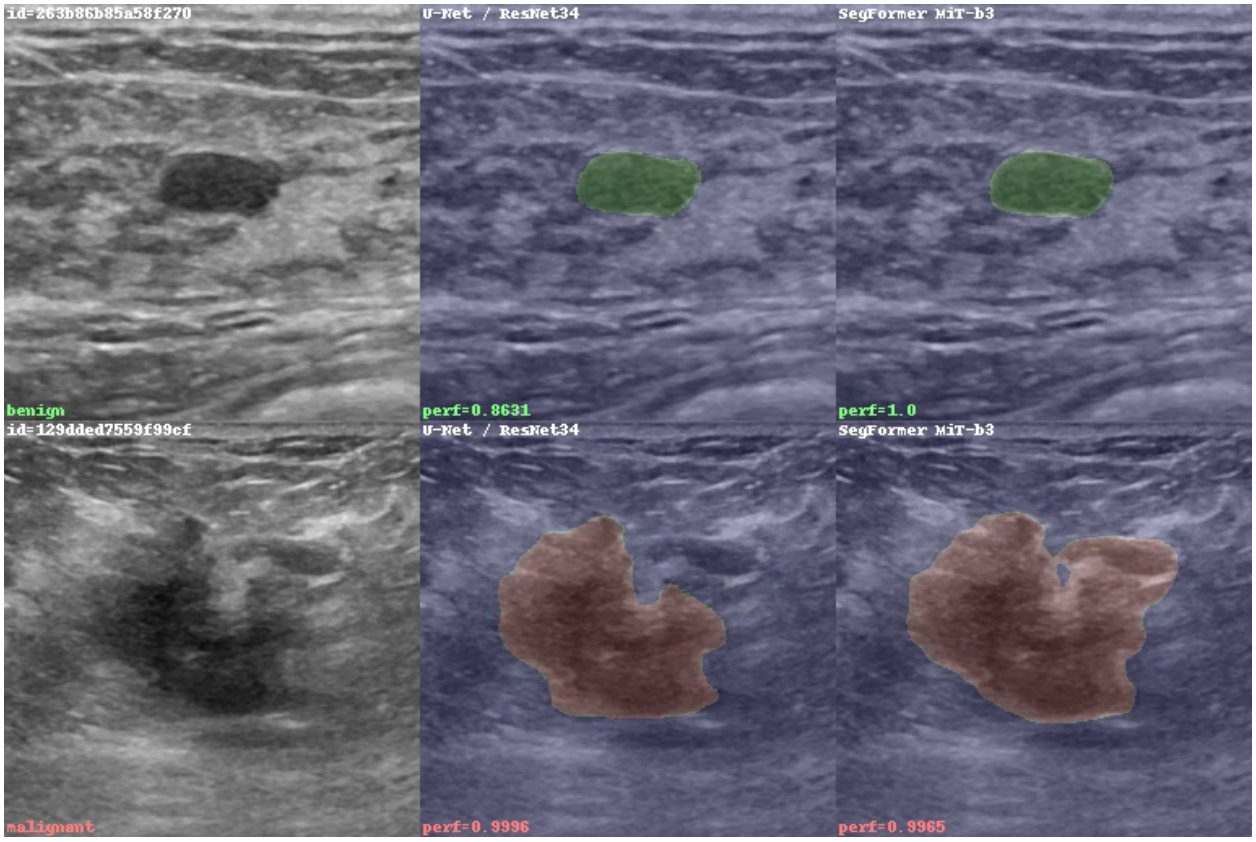


Figure 18 shows predictions exhibiting mediocre consistency. The top half shows the benign frame with predictions. The bottom half shows the malignant frame with predictions. Some or most predicted pixels are consistent with the nature of the lesion (benign or malignant), but significant fractions of the pixels are not consistent (have the wrong color).

Figure 18

Mixed Consistency Predictions on the BUV Dataset

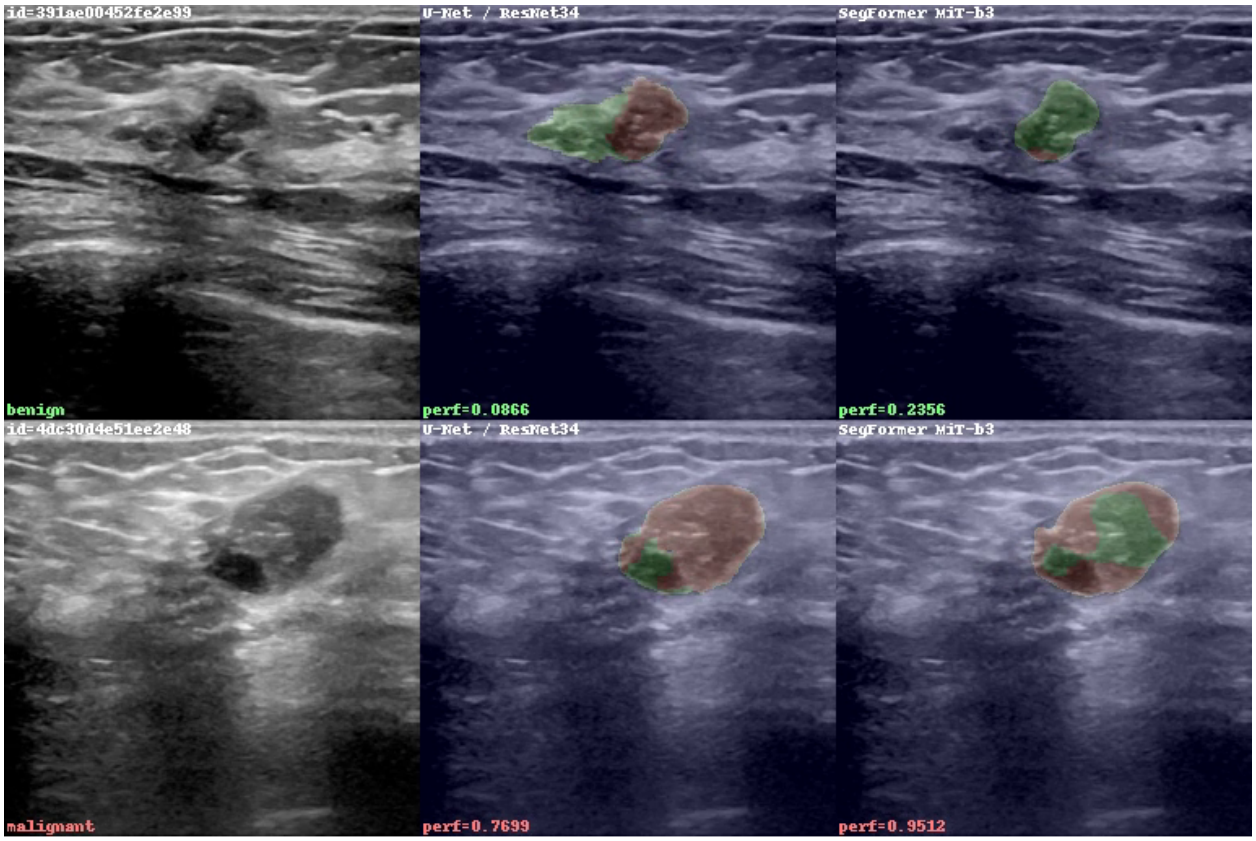
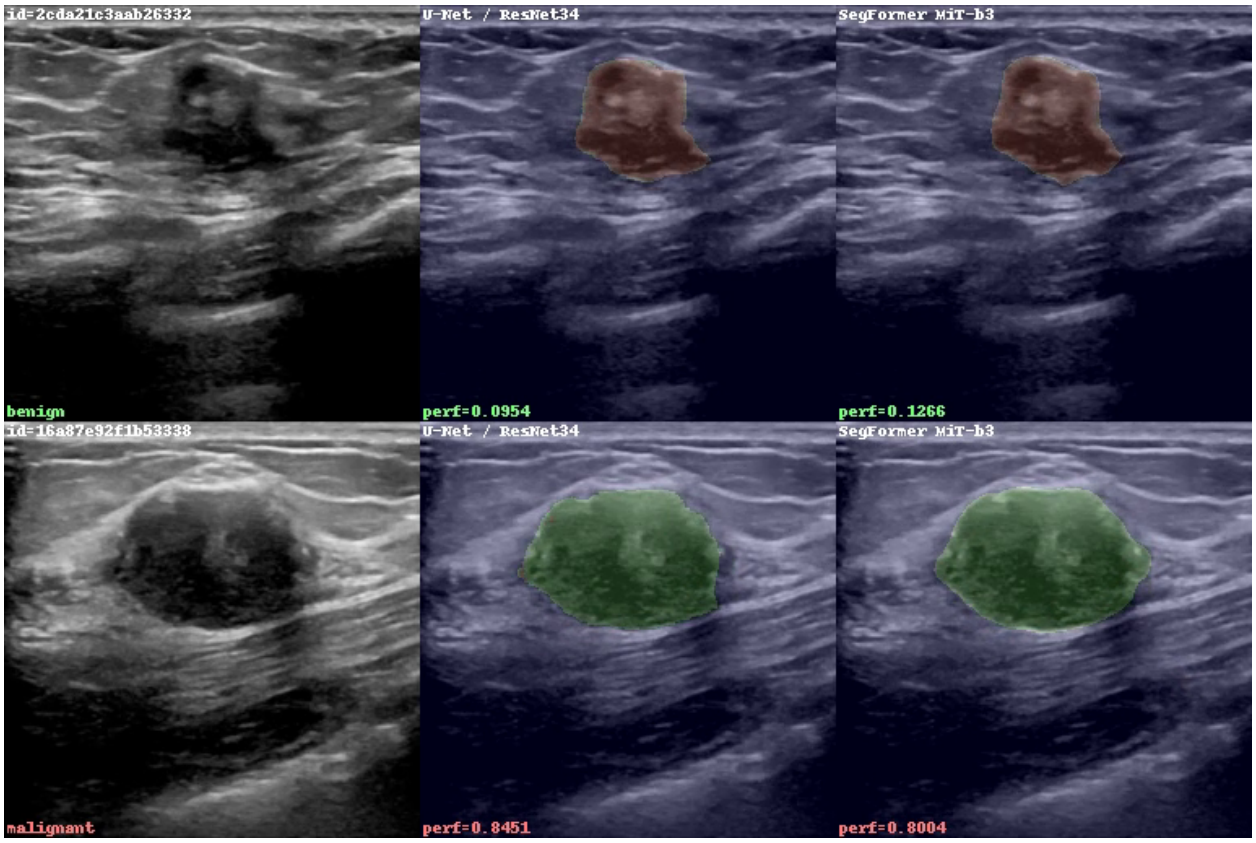


Figure 19 shows predictions that are completely inconsistent with the nature of the lesion. All predicted pixels are the wrong color.

Figure 19

Inconsistent Predictions on the BUV Dataset



For completion, the link below shows all predictions on the video dataset, from both models, in video format. The video is a good simulation of the behavior of the models while performing real time predictions on actual patients in a hospital.

https://www.youtube.com/watch?v=en4aTGsbp3U

Summary

The U-Net and SegFormer models were evaluated based on their IoU performance on the validation part of the unified training dataset. Additional performance indicators (consistency) were extracted from predictions on the BUV dataset.

Using IoU as a metric, SegFormer outperforms U-Net. The IoU metrics on the validation dataset, 75% for SegFormer and 64% for U-Net, indicate that the transformer architecture is

better at segmenting breast ultrasound images than the convolutional model. While not surprising, this result was not known at the beginning of this study.

Using the best combination of hyperparameters found for the main SegFormer, a separate SegFormer model was trained once for single-class segmentation, and it delivered 89% single-class IoU on the validation dataset.

The prediction consistency on the video dataset varies significantly, depending on the video sequence, or even depending on each individual frame in a sequence. The consistency of both models is as high as 100% on many sequences, or as low as near zero for a few sequences. This may indicate that making predictions one frame at a time may be inherently limited - the models, in some cases, may completely flip the classification (benign vs malignant) from one frame to the next, even though the frames are not very different visually.

Both models could be used to provide visual cues to radiologists when exploring lesions via ultrasound. U-Net has a very slim consistency margin over SegFormer when it comes to malignant lesions, but its consistency is poor with benign lesions. If the absolute best consistency is needed for detecting malignancy, U-Net has a slight advantage over SegFormer.

SegFormer tends to deliver better performance overall, as intuition would suggest for a much newer model architecture. In terms of aggregate performance over all metrics, it should be preferred over U-Net.

Chapter 5: Discussion

The purpose of this study was to identify one or several model architectures for semantic segmentation, construct a methodology for training and optimizing the models, measure the performance of the models using different metrics on different datasets, and make recommendations for further use based on the measured strengths of each model.

Summary of Findings

Two architectures were used for the models trained in this study: U-Net and SegFormer. Both models were trained on a combined dataset, made of four smaller, fully labeled datasets, with a total of 1794 images.

A list of hyperparameters was identified for each model, to be used in model optimization. A systematic search was conducted for each model, using the Optuna library, to find the combination of hyperparameters that delivers the best performance as measured by IoU. The best hyperparameter values were used to train the final, best models.

The average IoU performance found for the best models is as follows: 64.2% for U-Net, and 74.7% for SegFormer.

A separate, single class SegFormer model was trained using the best hyperparameters, and its IoU performance was determined to be 89%.

Using a partially labeled dataset consisting of nearly 200 video sequences with a total of over 25,000 images, the prediction consistency was estimated for the best U-Net and SegFormer models. For U-Net, the consistency was 34.5% on benign frames, and 90.5% on malignant frames. For SegFormer, the consistency was 58.2% on benign frames, and 85.5% on malignant frames.

Prediction consistency was found to vary significantly for both models, from one video sequence to another, and even from one frame to another.

Discussion

Both models performed well on most tasks. In terms of actually localizing the lesion in the image frame, and predicting the nature of the lesion, which is what IoU indicates, the performance of U-Net is fair. This architecture was proposed in 2015, and it is known to deliver acceptable results for segmentation tasks.

For semantic (multi-class) segmentation, SegFormer exhibited good performance. When trained for single-class segmentation (shape only, no class), SegFormer delivered very good performance.

Both models would be at least adequate for the task of localizing the lesion and identifying its nature or class. SegFormer has a clear advantage over U-Net for this task, which is unsurprising - by some measures, SegFormer delivered state-of-the-art segmentation performance on everyday image datasets in 2021, when it was proposed, and it is likely still close to that range at the moment of this writing. If localizing the lesion is the only goal, ignoring the nature or class of the lesion, only SegFormer was evaluated and it has delivered very good performance.

In terms of prediction consistency, U-Net has shown some amount of bias towards predicting malignant lesions. Its consistency is very good on malignant frames, but it is quite poor on benign frames. SegFormer is nearly as consistent as U-Net on malignant frames, and it is much better than U-Net, although just adequate in absolute terms, for benign frames consistency.

To provide visual cues in real time to radiologists, both models would perform well overall. U-Net has some advantage on malignant frames over SegFormer, which is important, but its margin is very slim.

Next Steps

Much work could still be done to expand this study in the future. An obvious issue is the amount and the quality of training data - the image datasets. For everyday segmentation tasks,

e.g. images of city streets, in recent years we have witnessed an abundance of high quality, very large datasets. For these tasks, training segmentation models is easy, and delivers high performance models. For the specific task of segmenting breast ultrasound images, the available datasets are scarce, small in terms of number of images, and their quality is not always very good. Increasing the number and the size of the image datasets, and improving the quality of the labeling, would almost certainly improve the model performance quite significantly.

Vision models tend to be large. Training such a model requires significant compute resources, and takes a long time for each session. Accelerating the training and performance evaluation of the models would benefit the optimization process, resulting in models with potentially higher performance. For this study, U-Net has performed predictions in batch mode, where multiple images are presented to the model, which then performs predictions on all images at once. Batch predictions are very fast. Due to various constraints, SegFormer was used in frame-by-frame prediction mode, which is slow. Not being able to run batch predictions on all models has consumed much of the project's time budget with slow frame-by-frame prediction routines. Switching the SegFormer predictions to batch mode would accelerate performance evaluation significantly, and ultimately would lead to better models.

The U-Net architecture was proposed in 2015. Transformers as a concept were introduced in 2017, with the SegFormer paper being published in 2021 when it was considered a state-of-the-art segmentation model. SegFormer is a much more recent architecture than U-Net. Looking at the general direction of deep learning research, essentially all recent milestones were delivered by different kinds of transformers. As is often repeated in this field nowadays, "transformers are the future". The next iteration of this project, and related projects, ought to focus on the latest research in the field of transformers. U-Net is a known quantity, but it has been outperformed, overall, by transformers.

Measuring the consistency of the predictions from the main models has revealed that predictions fluctuate significantly from one video sequence to the next, or even from one frame

to the next within the same sequence. This suggests that performing frame-by-frame predictions may be inherently limited. During discussions within the parent BUS project, one idea that was mentioned was to potentially explore models that estimate multiple frames in a video sequence before making predictions. This is different from batch predictions in that all frames together contribute to the predictions made for each frame individually. Whole-sequence predictions could potentially improve all of the model's performance metrics, and might be worth exploring in the future.

Within the parent BUS project, the results of this study could be used in a number of different ways. Segmentation models running predictions in real time could be used to provide visual cues to radiologists as they are exploring a lesion with ultrasound. It is easy for a human operator to miss subtle aspects of ultrasound images, and any cue provided in this process by a model to the operator may help. Single-class models could be used to provide generic visual cues as well ("there may be a lesion here, regardless of its class"), or the predictions of single-class models could be used as input for other models to make further predictions about the lesion.

Conclusion

This study has explored a number of different techniques and models (some of them state-of-the-art not long ago) to extract useful information from raw ultrasound images. The overarching goal was to use deep learning technology to improve the diagnostic process, and improve patient outcomes. The results have shown that deep learning could be successfully used now to help identify lesions and classify them according to their nature. The models could be included in an app that could provide real-time visual cues to human operators, or could be included in ensemble methods, with the output from some models becoming input for other models, in order to increase overall performance. We look forward to seeing the improvements in patient outcomes that this technology could deliver in real life.

References

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A Next-generation Hyperparameter Optimization Framework. *arXiv.org*. Retrieved November 26, 2022, from https://arxiv.org/abs/1907.10902

American Cancer Society (2022), *Breast Ultrasound*. Retrieved October 15, 2022, from https://www.cancer.org/cancer/breast-cancer/screening-tests-and-early-detection/breast-ultrasound.html

Baeldung. (2022). Intersection over Union for Object Detection. Retrieved December 10, 2022 from https://www.baeldung.com/cs/object-detection-intersection-vs-union

Bergstra, J., Bardenet, R., Bengio, Y., & Kegl, B. (2011). Algorithms for hyper-parameter optimization. *NIPS'11*. https://dl.acm.org/doi/10.5555/2986459.2986743

Bozinovski, S. (1976/2019). Reminder of the First Paper on Transfer Learning in Neural Networks, 1976. *Informatica 44*: 291–302. https://doi.org/10.31449/inf.v44i3.2828

Brownlee, J. (2019). A Gentle Introduction to Transfer Learning for Deep Learning. *In: Machine Learning Mastery*. Available from: https://machinelearningmastery.com/transfer-learning-for-deep-learning/

Brownlee, J. (2020). How Do Convolutional Layers Work in Deep Learning Neural Networks? *In: Machine Learning Mastery*. Available from: https://machinelearningmastery.com/convolutional-layers-for-deep-learning-neural-networks/

Byra, M. (2021). Breast mass classification with transfer learning based on scaling of deep representations. *Biomedical Signal Processing and Control*, 69, 102828. https://doi.org/10.1016/j.bspc.2021.102828

Chen, D.-R., & Hsiao, Y.-H. (2008). Computer-aided diagnosis in breast ultrasound. *Journal of Medical Ultrasound, 16(1), 46–56*. https://doi.org/10.1016/s0929-6441(08)60005-3

Cornille, T., Rogge, N. (2022). Fine-Tune a Semantic Segmentation Model with a Custom

Dataset. *HuggingFace*. Retrieved November 26, 2022, from https://huggingface.co/blog/fine-tune-segformer

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv.org*. Retrieved October 29, 2022, from https://arxiv.org/abs/2010.11929

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. *MIT Press*. http://www.deeplearningbook.org

Guo, Y., Duan X., Wang C., & Guo H. (2021). Segmentation and recognition of breast ultrasound images based on an expanded U-Net. *PLoS ONE* 16(6): e0253202. https://doi.org/10.1371/journal.pone.0253202

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. *arXiv.org*. Retrieved November 26, 2022, from https://arxiv.org/abs/1512.03385

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation* 9(8):1735-1780. https://blog.xpgreat.com/file/lstm.pdf

Huh, M., Agrawal, P., & Efros, A. A. (2016). What makes ImageNet good for transfer learning? *arXiv.org*. Retrieved October 29, 2022, from https://doi.org/10.48550/arXiv.1608.08614

Hussain, Z., Gimenez, F., Yi, D., & Rubin, D. (2018). Differential Data Augmentation Techniques for Medical Imaging Classification Tasks. *Annual Symposium proceedings. AMIA Symposium, 2017*, 979–984. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5977656/

Jahwar, A. F., Abdulazeez, A. M. (2022). Segmentation and Classification for Breast Cancer Ultrasound Images Using Deep Learning Techniques: A Review. *CSPA 2022*. https://doi.org/10.1109/CSPA55076.2022.9781824

Lau, S. (2017). Image Augmentation for Deep Learning. *Medium*. Retrieved October 29, 2022 from https://towardsdatascience.com/image-augmentation-for-deep-learning-histogram-equali

zation-a71387f609b2

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation, 1(4), 541–551*. https://doi.org/10.1162/neco.1989.1.4.541

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

Loshchilov, I., Hutter, F. (2017). Decoupled Weight Decay Regularization. *arXiv.org*. Retrieved November 26, 2022, from https://arxiv.org/abs/1711.05101

Magny, S.J., Shikhman, R., & Keppke, A.L. (2022). Breast Imaging Reporting and Data System. *In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing*. Available from: https://www.ncbi.nlm.nih.gov/books/NBK459169/

Matsoukas, C., Haslum, J. F., Söderberg, M., & Smith, K. (2021). Is it Time to Replace CNNs with Transformers for Medical Images? *arXiv.org*. Retrieved October 29, 2022, from https://arxiv.org/abs/2108.09038

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv.org*. Retrieved October 15, 2022 from https://doi.org/10.48550/arXiv.1505.04597

Saha, S. (2018). A Comprehensive Guide to Convolutional Neural Networks - the ELI5 way. *Medium*. Retrieved October 29, 2022, from https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

Shen, D., Wu, G., & Suk, H.-I. (2017). Deep learning in medical image analysis. *Annual Review of Biomedical Engineering*, 19(1), 221–248. https://doi.org/10.1146/annurev-bioeng-071516-044442

Shen, Y., Shamout, F. E., Oliver, J. R., Witowski, J., Kannan, K., Park, J., Wu, N., Huddleston,

C., Wolfson, S., Millet, A., Ehrenpreis, R., Awal, D., Tyma, C., Samreen, N., Gao, Y., Chhor, C., Gandhi, S., Lee, C., Kumari-Subaiya, S., … Geras, K. J. (2021). Artificial Intelligence System reduces false-positive findings in the interpretation of breast ultrasound exams. *Nature Communications, 12(1)*. https://doi.org/10.1038/s41467-021-26023-2

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for Deep Learning. *Journal of Big Data*, 6(1). https://doi.org/10.1186/s40537-019-0197-0

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2015). *Rethinking the inception architecture for computer vision*. arXiv.org. Retrieved October 29, 2022, from https://doi.org/10.48550/arXiv.1512.00567

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, L., & Polosukhin, I. (2017). Attention Is All You Need. *arXiv.org*. Retrieved October 29, 2022, from https://arxiv.org/abs/1706.03762

Vignesh, S. (2020). The world through the eyes of CNN. *Medium*. Retrieved October 29, 2022, from https://medium.com/analytics-vidhya/the-world-through-the-eyes-of-cnn-5a52c034dbeb

World Health Organization. (2021). Breast cancer. Retrieved October 15, 2022, from https://www.who.int/news-room/fact-sheets/detail/breast-cancer

Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J. M., & Luo, P. (2021). SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. *arXiv.org*. Retrieved October 29, 2022, from https://arxiv.org/abs/2105.15203

Appendix A: Code and Artifacts

The GitHub repository with all the code for this project:

https://github.com/FlorinAndrei/datascience_capstone_project

The two main Jupyter notebooks containing the training and hyperparameter search for the best U-Net and SegFormer models:

https://github.com/FlorinAndrei/datascience_capstone_project/blob/main/unet_fine_tune.ipynb

https://github.com/FlorinAndrei/datascience_capstone_project/blob/main/segformer_fine_tune.ipynb

The Jupyter notebook with the training code for the single-class SegFormer:

https://github.com/FlorinAndrei/datascience_capstone_project/blob/main/segformer_single_class.ipynb

The Jupyter notebook with the consistency metrics for both main models. It also generates the video frames for the video artifact:

https://github.com/FlorinAndrei/datascience_capstone_project/blob/main/merge_unet_segformer.ipynb

Dataloader library:

https://github.com/FlorinAndrei/datascience_capstone_project/blob/main/bus_data.py

Video:

https://www.youtube.com/watch?v=en4aTGsbp3U