Multitask Learning for Joint Semantic Segmentation and Classification of Ovarian Lesions in Ultrasound Scans

Dani Rogmans

Delft University of Technology

TU Delft

Multitask Learning for Joint Semantic Segmentation and Classification of Ovarian Lesions in Ultrasound Scans

by

Dani Rogmans

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Monday June 30

Student number: 4885279
Project duration: November 2024 - June 2025
Thesis committee: Nergis Tomen, Jouke Dijkstra, Jan van Gemert, Ricardo Marroquim

TU Delft

Acknowledgments

The project undertaken by this thesis was an interdisciplinary effort that combined the TU Delft, the LUMC gynecology department, the LUMC Division of Image Processing (LUMC-LKEB), and the Frisius MC gynecology department.

On the LUMC side, my supervisor was Jouke Dijkstra. He introduced me to the comfortable environment of LKEB, which was back then temporarily housed at the LUMC's Pelikaan building. Throughout the duration of my thesis, LKEB would turn out to be the place where I spent the most time, surrounded by motivated and like-minded researchers who also applied computer vision to medical problems. Jouke continuously offered his help, support and opinions during my thesis and was the link that connected the deep learning work that I brought with the medical insights from the doctors we met with.

On the TU Delft side, many thanks to Nergis Tomen for consistent meetings, where she offered specific deep learning advice, suggestions and new ideas. From the beginning of the thesis, she made it clear that my thesis should not only be clinically relevant but also further the field of computer vision, and this is what prompted me to explore multitask learning architectures. After I sent her my first draft, she gave me feedback within the next day and this drastically changed my thesis structure for the better. She also gave good lectures in the first year of my Master's on spiking neural networks.

Every two weeks, meetings were held with Bart Groen and Cor de Kroon. Without Bart and Cor's help and domain-knowledge related to ovarian tumor classification, the results presented in this thesis would not be as optimistic. Key ideas like introducing the type of center as a clinical variable and adding explicit focus to the ovarian lesion for classification were provided by them. For this reason, I would like to thank both Cor and Bart and I hope they will continue to advance this research in the coming years.

My many thanks to Jan van Gemert and Ricardo Marroquim for being part of the thesis committee.

Last but not least, my father, mother, and sister may not have contributed technical expertise, but their support in every other aspect of my life was invaluable.

Dani Rogmans
Delft, June 2025

Contents

1

Introduction

This project was first commissioned by Bart Groen, Jouke Dijkstra and Cor de Kroon. Early stage ovarian cancer detection is still problematic, and both false negatives and false positives carry significant consequences: missed diagnoses can delay treatment and worsen patient outcomes, particularly in malignant cases where early intervention is critical. False positives may lead to unnecessary surgical procedures such as laparotomies, misuse of limited healthcare resources, and psychological distress for patients who may undergo invasive interventions for benign conditions. Current methods for estimating the probability of malignancy in ovarian tumors rely on the clinician's level of expertise, which leads to variability in accuracy and reliability across different centers in the Netherlands. For these reasons, the applications of deep learning to automatically classify ovarian tumors are being widely explored.

The dataset used in this report consists of scans from three different Dutch hospitals, collected over the course of several years. As a result, the dataset is highly diverse; it includes considerable variation in terms of operators, ultrasound machine manufacturers and models, and scanning conditions. This complexity, as well as the relatively small size of the dataset (approximately 4000 scans, mostly benign), made classification initially challenging and necessitated the exploration of several potential improvements. These included the removal of burned-in medical annotations, the incorporation of clinical factors and implicit and explicit methods of adding focus to the lesion mask, which is the most important determinant of malignancy in ultrasound ovarian scans. Regular meetings with supervisors from both the deep learning and medical domains were incredibly helpful in bridging domain knowledge and guiding the implementation process.

The results presented in this thesis offer a promising next step to implementing and deploying deep learning-based solutions for ovarian tumor classification. The overall best-performing classifier was the one that incorporated all three of the proposed improvements. However, there were still some ambiguous and incorrectly classified cases in the test set, which in a real-world setting would require manual clinical involvement. Nonetheless, I am very optimistic about the applications of deep learning for ovarian tumor classification in ultrasound scans, and some adjustments to the data (e.g, using more training images, predicting on multiple slices per patient or using 3D volume scans) may be the final step needed to achieve the level of performance required for clinical use.

The remainder of this thesis is structured as follows: chapter 2 contains the main report, structured in a Computer Vision conference-style format similar to CVPR. This chapter includes related works, methodologies, experiments and results. Chapter 3 and 4 provide the reader with the necessary background knowledge on neural networks for classification and semantic segmentation. Finally, the appendices offer more details behind the experiment, the hyperparameters and the pre-processing steps. The code for this project can be found on my Github (`https://github.com/dtronmans`), which includes the code for the denoising autoencoder to remove medical annotations and the training scripts for single-task and multitask-learning networks, as well as pre-processing scripts.

2

Paper

Page left intentionally blank.

Multitask Learning for Joint Semantic Segmentation and Classification of Ovarian Lesions in Ultrasound Scans

Master's Thesis
Dani Rogmans, Jouke Dijkstra, Bart Groen, Cor de Kroon, Nergis Tomen

Abstract

Distinguishing between benign and malignant ovarian cysts is a challenging task that depends on subjective visual markers in ultrasound scans. Current manual methods remain prone to costly misdiagnoses and the application of these methods depend heavily on the clinician's level of expertise. Recent research demonstrates promising applications of Convolutional Neural Networks (CNNs) for ovarian tumor classification; however, we observed that their performance is limited when applied to a diverse and complex dataset. To address this, we propose, implement, and evaluate three improvements to a baseline classifier.

First, we use a deep learning-based approach to remove burned-in medical annotations and introduce a weighted mean squared error (MSE) loss to improve its effectiveness by emphasizing relevant regions. This aims to better recover the original image content prior to annotation and remove annotations which can act as confounders. Second, we enhance classification by fusing image features with two readily available clinical factors at an intermediate stage of the network. Third, and central to this study, we incorporate a segmentation path that acts as a regularizer, encouraging the shared encoder to learn lesion-specific features that benefit the classification head.

These three contributions are informed by domain-specific knowledge of ovarian lesions and collectively demonstrate promising directions for improving deep learning-based models in this setting.

1. Introduction

Ovarian cancer is one of the most lethal gynecological malignancies worldwide, with an estimated 324,000 new cases and 207,000 deaths annually [1]. In the Netherlands, ovarian cancer remains a significant health concern, with around 1,500 new cases and 1,100 deaths annually [2]. Accurately distinguishing between ovarian tumors at an early stage could have a drastic positive impact on the survival rate; survival rates drop to 27% for Stage III and 13% for Stage IV diagnoses [3].

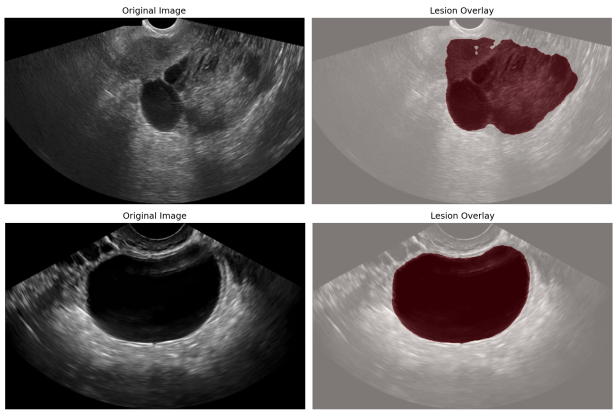


Figure 1. Examples of ovarian ultrasound scans. Each row shows a different case, with the left image displaying the raw ultrasound scan and the right image showing the same scan with the lesion area highlighted in red. Surrounding anatomical structures, such as the bladder and bowel, are visible in the raw scans but are not relevant for classification. The highlighted lesion regions contain the discriminative visual features used to distinguish between benign and malignant tumors, despite heterogeneity in the surrounding anatomy.

Given the significant benefits of detecting ovarian cancer at an early stage, the objective of this report is to develop and improve classifiers for distinguishing between benign and malignant tumors in ultrasound scans. This project is a collaborative effort between Frisius MC (Bart Groen) and LUMC (Cor de Kroon, Jouke Dijkstra), representing an initial step toward replacing current manual diagnostic methods with deep learning-based approaches. Over several years of data collection, a diverse dataset of transvaginal ultrasound scans has been compiled. This dataset encompasses scans from three different Dutch hospitals, various ultrasound machines, operators, and device models.

Recent works in deep learning have shown that applying classification models to 2D ultrasound scans can yield satisfactory results in distinguishing between benign and malignant tumors. However, in our case, such success with

a simple baseline classifier has not been observed. This is likely due to the limited size of our dataset, combined with the heterogeneity of the scans.

Given the unsatisfactory performance of classifiers on our dataset, this study investigates three enhancements to a baseline model. Each of these improvements integrates domain-specific knowledge related to ultrasound imaging, ovarian cancer, and tumor diagnosis into the convolutional neural network.

As a first improvement, we pre-process the images by removing medical annotations such as text and arrows. Most ultrasound scans in the dataset are annotated by clinicians, and these annotations can act as confounders that CNNs may learn instead of features relevant to ovarian tumor classification. Since these annotations are burned into the image and not stored as separate layers, we remove them using a denoising autoencoder. Although prior work has explored the use of deep learning for removing medical annotations, the proposed loss functions in those studies had certain limitations, which we address through a weighted loss function and the incorporation of skip connections into the autoencoder to better preserve high-frequency information.

The second potential improvement we investigate is the incorporation of clinical features. 2D ultrasounds can be ambiguous and challenging to interpret on their own, so the addition of clinical information such as menopausal status can give the classifier additional contextual information to support its decision-making. In this study, clinical tabular features are fused with image features at an intermediate stage of the network, prior to the final multilayer perceptron (MLP) that returns the classification output.

Finally, we evaluate the impact of guiding the classifier to focus on the lesion mask. As Figure 1 shows, the key visual characteristics in determining malignancy are derived physical properties of the ovarian lesion itself. The rest of the image, which can include the bladder, bowel and uterus are considered irrelevant in the visual assessment. This domain- specific knowledge can be used to construct classification models that focus solely on the lesion region, and reduce the influence of surrounding anatomical structures. This is primarily done through the addition of a semantic segmentation decoder, turning the classification task into a multitask learning scenario. The semantic segmentation decoder, as well as its associated loss term acts as a regularizer that encourages the encoder to learn lesion-specific features. While this type of multitask setup has been explored with U-Net architectures, we argue that an auxiliary segmentation path can be added to any baseline classifier to potentially improve performance. In our case, we implement the auxiliary segmentation path on an EfficientNetB0 architecture.

The contributions of our work can be summarized as follows:

- We devise an automatic method of generating clean-annotated image pairs to train a denoising autoencoder to remove medical annotations, using a weighted MSE loss.
- We propose and evaluate an intermediate fusion method for adding clinical factors, namely the oncology center and the menopausal status, to the classification head of deep learning architectures.
- We assess the potential performance improvement of multitask learning architectures for joint classification and semantic segmentation of ovarian lesions, compared to equivalent single-task baselines.
- We explore the effectiveness of a cascaded approach in which ovarian lesions are first segmented and then classified, and compare it to joint multitask learning methods.
- We extend existing joint classification–segmentation frameworks by developing and evaluating a multitask model based on an EfficientNetB0 [4] backbone.
- The existing dataset, with scans from three different Dutch hospitals, is expanded with more semantic segmentation labels. This was done in a collaborative effort by the author of this thesis, Maite Timmermans, Cor de Kroon and Bart Groen.

The report is structured as follows: we start by a treatment of already-existing deep learning methods for ovarian cyst classification, multitask learning in medical settings and annotation removal. Section 3 describes the methods, including the details behind the architectures we devised, trained and evaluated. Section 4 is a detailed overview of the experimental setup, including hyperparameters and preprocessing steps, in order to ensure reproducibility and to compare architectural changes under similar settings. In Section 5, results are shown under all the presented ablations, and Section 6 is a discussion of the results. We conclude our report in Section 7, and offer future improvements in Section 8.

2. Related Work

2.1. Visual Confounders and Annotation Artefacts

Medical annotations such as overlaid text, arrows, or measurement markers can act as confounding factors for CNNs. These models may inadvertently learn to rely on such annotations instead of lesion-specific features, potentially limiting their generalization and diagnostic performance. In our dataset, most scans include such annotations; examples are shown in Figure 2.

Therefore, several works propose deep learning-based approaches for removing medical annotations. The process

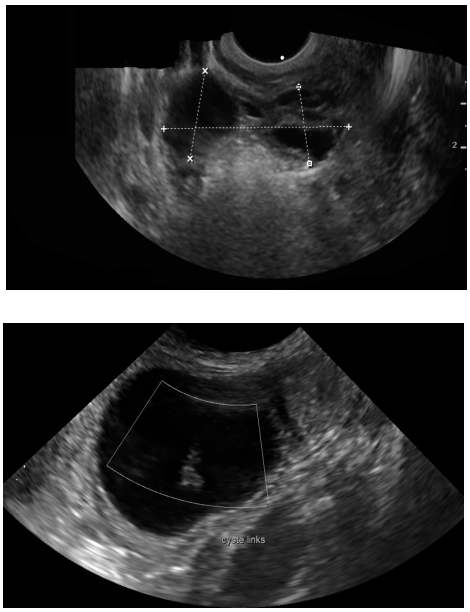


Figure 2. Transvaginal ultrasounds from our dataset that contain burned-in medical annotations. These annotations are in the form of arrows, text and bounding boxes, and can act as confounders. We aim to evaluate the diagnostic performance of classifiers in settings with and without medical annotations, to measure the impact of removing them.

involves using a standard convolutional autoencoder with a mean-squared error [5] or SSIM loss [6], and treating annotations as noise. However, these loss functions did not produce satisfactory results in our case. To address this, we developed a custom loss function that combines a weighted MSE with the SSIM loss.

The existing methods rely on original DICOMs being available before the clinician's annotations to gather clean-annotated image pairs, or a manual overlay of annotations on clean images. We propose an automated method for generating clean–annotated image pairs suitable for training denoising models for the task of removing medical annotations. Our approach leverages the existence of a subset of unannotated images in the dataset and uses this subset to synthetically model overlaid annotations, without requiring manual intervention or pre-annotation images.

2.2. Single and multi-modal deep Learning for Ovarian Tumor Classification

Past research shows promising results for the application of deep learning for classifying ovarian tumors. However, privacy and security concerns mean that there is no unified benchmark dataset for ovarian cancer classification. Furthermore, although many deep learning methods have been devised for classifying ovarian cancer, the data modalities used change drastically between each approach.

Ovarian tumor classification using deep learning on MRI scans has demonstrated diagnostic performance comparable to that of radiologists in multiple studies [7, 8]. Similar to our approach, first localizing the ovarian tumor and then feeding the cropped region into a classifier has also been shown to significantly enhance classification performance in MRI-based methods [9]. However, MRI remains a more expensive first-line imaging modality compared to ultrasound, and it does not consistently outperform simple diagnostic rules [10].

Some studies have directly compared the performance of deep learning-based approaches with expert subjective assessment (SA) on ultrasound scans [11, 12], demonstrating comparable or even superior results across key evaluation metrics such as sensitivity, specificity, and accuracy. However, despite also focusing on ultrasound imaging, these studies do not measure the impact of adding clinical information or incorporating semantic segmentation masks.

Past work also includes machine learning models that accurately classify ovarian cancer using purely tabular clinical information. In order to reach a satisfactory accuracy for early-stage diagnosis without imaging information, these models rely on up to 49 features [13], including biomarkers, ovarian cancer markers, and blood test results. This makes them complex to implement in typical hospital settings. In our case, using the available clinical features alone, we were unable to develop classifiers with sufficient accuracy.

Motivated by the limited performance of models trained solely on imaging or tabular data in our primary dataset, we extend prior work by developing multimodal classifiers that combine easily attainable imaging and clinical features.

2.3. Clinical Data and Multimodal Fusion

Ovarian tumor diagnosis is inherently a multimodal task, with physicians relying on a combination of image-level and clinical features to assess the probability of malignancy. A widely used method for estimating the risk of malignancy in ovarian tumors in the IOTA Adnex [14], which relies on the combination of six visual markers and three clinical variables. Despite this, the integration of clinical and imaging features for ovarian tumor classification remains largely unexplored.

Fusing clinical and image features can be done at different levels. Early fusion involves combining raw clinical and imaging data before feature extraction, allowing the clinical information to guide the extraction of image features [15, 16]. Intermediate fusion takes place after image features have been extracted, and integrates these features with clinical information to improve classification performance. Fusing clinical information with extracted image features after the encoder stages has been shown to improve breast cancer classification when combined with MRI im-

age data [17]. Finally, late fusion involves training separate models for each modality, and aggregating decisions from each sub-model [18].

In our case, the tabular data consisted of only two clinical features, and standalone models trained on them failed to achieve meaningful performance. As a result, pursuing a late fusion strategy was not justified, and we chose to rely solely on the popular and widely successful intermediate-fusion strategy.

Numerous studies have shown that the risk of malignancy is higher in postmenopausal women when presenting with adnexal masses [19, 20]. Some studies cite age as a predictor of malignancy risk, which can be seen as a proxy for the menopausal status [21].

The type of center to which a patient is admitted is also a predictor of malignancy risk. Patients evaluated at oncology centers have a higher probability of malignancy than those assessed at other types of centers. This is likely because oncology centers tend to receive referrals for more complex or suspicious cases, leading to a higher pre-test probability of malignancy [22]. This variable is one of the three clinical features incorporated into the IOTA Adnex model [14].

Our study focuses on the multimodal fusion of ultrasound scans and easily obtainable clinical information, specifically the patient's menopausal status and whether the admission center is an oncology facility. Transvaginal ultrasound (TVS) is a widely accessible, low-cost, and safe imaging modality that does not expose patients to ionizing radiation [23, 24]. The clinical variables we incorporate are routinely collected during initial assessments and do not require any additional procedures. This makes our approach more practical and broadly applicable in real-world clinical settings, especially where access to advanced or expensive diagnostic tools is limited.

2.4. Multitask Learning in Medical Computer Vision

In the context of deep learning for computer vision, multitask learning refers to a network learning multiple tasks at the same time, by making use of a shared encoder and separate decoder heads [25]. In settings where tasks are related or make use of similar visual cues, and in settings where data is scarce, this can result in better performance across tasks compared to separate networks. This is because shared representations allow the network to generalize better by leveraging complementary information across tasks, acting as an inductive bias that helps regularize the learning process [26].

For this reason, multitask learning can be used in medical imaging scenarios where the outcome of the diagnosis depends on a specific Region of Interest (ROI). When visual characteristics of lesions determine the outcome, it could be beneficial to simultaneously learn semantic segmentation to localize the ROI and classification to determine the presence or severity of disease. The segmentation task guides the encoder to focus on relevant anatomical structures like ovarian masses, which helps the classification head avoid overfitting to irrelevant regions and focus on the most informative areas for diagnosis.

Shared encoders with different downstream tasks have shown promising results in medical settings where a given region-of-interest is the most important determinant of a classification outcome; use cases include primary bone tumors in radiographs, breast cancer classification in 3D ultrasound scans, histology images and Chronous Venous Disorders (CVD) [27–30].

The Multi-Modality Ovarian Tumor Ultrasound (MMOTU) image dataset is a dataset of 1,469 2d ultrasound scans with eight tumor classes and corresponding lesion segmentations for each scan [31]. The original study evaluated single-task classification and segmentation performance but did not evaluate the potential benefits of multitask learning. The addition of a segmentation loss as a feature-enforcing regularizer could be particularly useful in their case given the limited size of the dataset. For this reason, this report evaluates the effects of multitask learning not only on the primary Dutch hospital dataset but also on the MMOTU dataset.

The U-Net is one of the earliest and most influential works in deep learning-based semantic segmentation [32]. In U-Net backbones, multitask learning for joint semantic segmentation has been shown to slightly outperform single-task equivalents [33]. U-Net encoder blocks are not typically used for classification because they are designed for dense prediction tasks like segmentation. Their limited use and performance in classification settings prompted us to also evaluate EfficientNetB0-based multitask learning architectures, in order to extend our results to another backbone and reach better classification outcomes.

3. Methods

3.1. Denoising Autoencoder for Medical Annotations

We start with a brief overview of the denoising autoencoder, which is trained in a supervised manner to remove drawn text and arrows from ultrasound scans.

3.1.1 Self-supervised Annotations and Training

The architecture for removing embedded medical annotations such as arrows and text is a standard U-Net. Originally, a standard Convolutional Autoencoder architecture was used, but the lack of skip connections led to a loss of higher-frequency information and contrast in the reconstructed image.

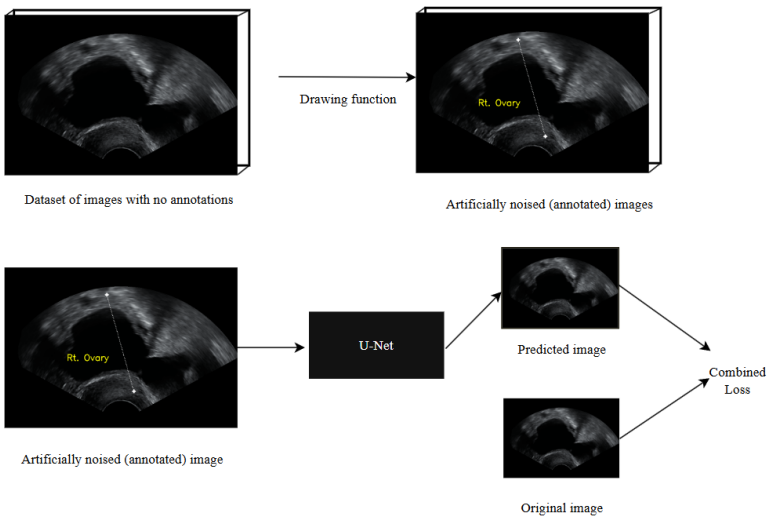


Figure 3. The process of generating images and training the denoising autoencoder

The steps in generating the training dataset, comprised of clean-annotated image pairs, were as follows:

1. Retrieved the subset of images from the dataset of images that do not contain medical annotations, as well as their ovarian mass masks.

2. Trained a U-Net on the images to automatically detect ovarian masses.

3. For each image, randomly select two starting and two endpoints on the edge of the ovarian mass, and drew arrows that use these points to denote their origin and end.

4. Randomly selected one entry from a text bank of common annotation, like ”Fluid” or ”Rt Ov” and drew the text at a random location on the ovarian mass.

The training process is as follows:

1. Sample a batch of artificially annotated images from the dataset.

2. Use the model to predict clean (inpainted) images from the annotated images.

3. Compute the reconstruction loss between the original clean image, and update the model parameters via backpropagation.

Figure 3 shows the pipeline for generating a clean-annotated pairs dataset and training the model.

3.1.2 Loss Function

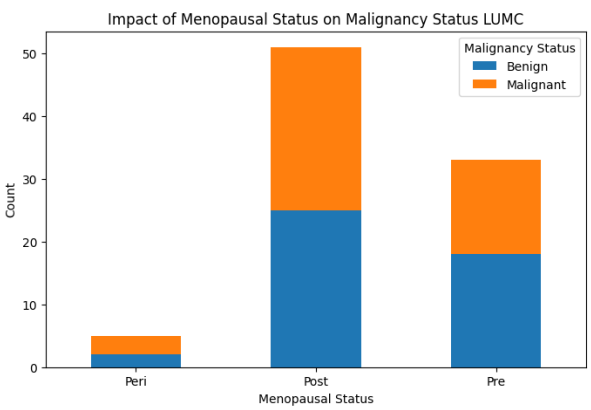
Although the mean-squared error (MSE) loss was used by previous studies, we encountered two potential shortcomings in our experiments:

- The task of removing annotations is inherently imbalanced, as the majority of pixels in an ultrasound image do not contain text or arrows. As a result, even a model that simply outputs the input image can achieve a low MSE loss without addressing the annotated regions

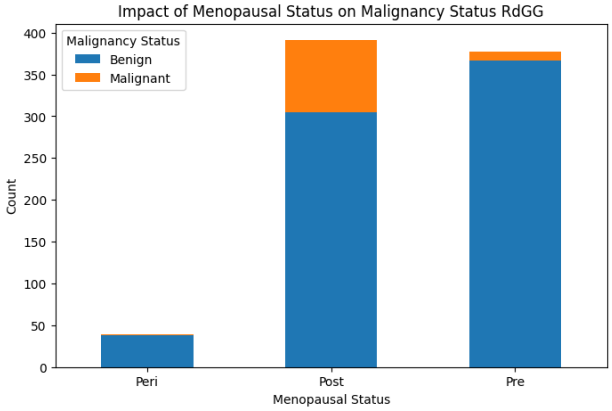
- Pixels in the predicted image may be numerically close to those in the ground truth but not perceptually similar. MSE does not account for structural or contextual information, leading to outputs that may lack visual fidelity. For this reason, perceptual losses like the Structural Similarity Index (SSIM) [34] are sometimes incorporated to capture human-perceived similarity and preserve anatomical structures

The loss function we have devised for this study is a weighted combination of two terms: an SSIM-based loss and a weighted MSE loss. The weighted loss is a traditional mean squared error that assigns higher importance to pixels affected by annotations (pixels that differ between the clean and annotated images), which allows the model to focus more on regions requiring restoration. The SSIM-based loss term is included to preserve perceptual and structural information, enforcing that the reconstructed images maintain the anatomical coherence of the clean images.
Let:

- $\hat{I}$ denote the predicted image (output from the U-Net).



(a) Benign and malignant distributions among Leiden University Medical Center (LUMC) patients, which is an oncology center



(b) Benign and malignant distributions among Reinier de Graaf Gasthuis (RdGG) patients, which is not an oncology center

Figure 4. Benign (orange) and malignant (blue) distributions according to menopausal status for different types of center. (a) LUMC, an oncology center, shows no identifiable pattern between the menopausal status and the ovarian malignancy, while (b), a non-oncology center, shows the majority of malignant cases are present in post-menopausal women.

- I_{clean} denote the clean target image (ground truth).
- I_{annot} denote the annotated input image (with added text/arrows).
- α and β be scalar weights for the pixel-wise weighted loss (e.g., $\alpha = 10.0$, $\beta = 0.10$).
- λ_{MSE} and λ_{SSIM} be weighting factors for the combined loss function.
- $\text{SSIM}(\cdot, \cdot)$ denote the Structural Similarity Index between two images.

$$M(x,y) = \begin{cases} 1 & \text{if } I_{\text{annot}}(x,y) \neq I_{\text{clean}}(x,y) \\ 0 & \text{otherwise} \end{cases}$$

$$w(x,y) = \alpha \cdot M(x,y) + \beta \cdot (1 - M(x,y))$$

$$\mathcal{L}_{\text{weighted}} = \frac{1}{\sum_{x,y} w(x,y)} \sum_{x,y} w(x,y) \cdot \left(I_{\text{clean}}(x,y) - \hat{I}(x,y) \right)^2 \tag{1}$$

$$\mathcal{L}_{\text{SSIM}} = 1 - \text{SSIM}(\hat{I}, I_{\text{clean}})$$

$$\mathcal{L}_{\text{total}} = \lambda_{\text{MSE}} \cdot \mathcal{L}_{\text{weighted}} + \lambda_{\text{SSIM}} \cdot \mathcal{L}_{\text{SSIM}}$$

Initial experiments were performed with values $\lambda_{\text{MSE}} = 1.0$, $\lambda_{\text{SSIM}} = 0.05$, $\alpha = 10.0$, and $\beta = 0.10$. However, through experimentation, we found that exaggerated values of α that were close to the ratio of annotated pixels to clean pixels worked very well, while preserving detail and higher-frequency information. Furthermore, there were no visual improvements from the addition of the SSIM term to the loss, and it was therefore removed. The denoising autoencoder used for the purpose of this study was trained with values $\lambda_{\text{MSE}} = 1.0$, $\lambda_{\text{SSIM}} = 0$, $\alpha = 300.0$, and $\beta = 1.0$.

Appendix A shows examples of the denoising autoencoder, trained with the hyperparameters mentioned above, being used to remove medical annotations.

3.2. Fusion of Clinical and Image Features

The clinical features used to enhance classification results are the menopausal status and the type of center (oncology vs. normal). These clinical features are appended to the classification head of the joint and classification-only networks.

As previously stated, a patient's menopausal status is a significant factor in assessing the risk of ovarian cancer malignancy. However, this relationship between the menopausal status and malignancy is not generally reflected in oncological centers, where more complex or high-risk cases are typically referred.

To illustrate this, Figure 4 presents the distribution of menopausal status among patients at LUMC - an oncological facility - and RdGG - a non-oncological facility, respectively. As the figures show, non-oncology centers like RdGG have very few pre-menopausal malignant patients, while oncology centers have a distribution of benign and malignant cases that does not seem impacted by the menopausal status. For this reason, the IOTA Adnex also relies on the type of center as one of the three clinical feature. Due to the low number of samples, peri-menopausal pa-

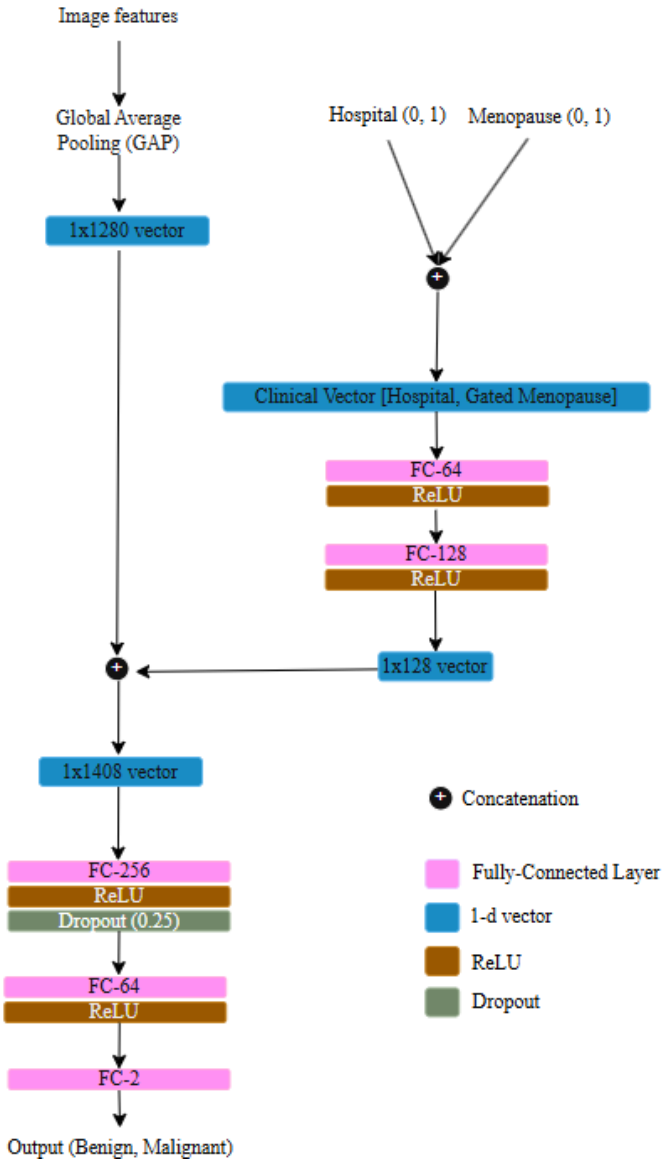


Figure 5. The fusion of image features and clinical features before feeding both into a classification MLP. This is a form of intermediate fusion, where pooled image features are combined with clinical features that are projected to a higher dimension.

tients have been combined with post-menopausal patients.

The third and last clinical factor used by the IOTA Adnex model is the CA125 biomarker, and the impact of incorporating it will not be measured because too many patients had missing values for this feature.

The integration of these two clinical features in the classification head is shown in Figure 5. This exact configuration is applied to the classification head of every tested network.

3.3. Multitask Learning Architectures

We compare the performance of two multitask learning architectures with their single-task counterparts: the standard U-Net and a custom EfficientNetB0-based architecture. EfficientNetB0 was selected as an additional backbone due to its strong classification performance as a baseline classifier on the primary dataset compared to ResNet18, ResNet50 and DenseNet121.

3.3.1 EfficientNet backbone

Figure 6 illustrates the architecture of the EfficientNetB0-based multitask model. The downsampling path is identical to the original EfficientNetB0 encoder, and a the same multimodal classification head is used in single-task and multitask settings. By keeping the classification branches and hyperparameters consistent, we minimize incidental differences in performance, which allows us to more directly examine the effects of multitask learning.

3.3.2 U-Net backbone

Current multitask learning methods mainly rely on a U-Net backbone for joint classification and semantic segmentation. Image features are pooled into the classification head from different stages. For example, CMSVNetIter pools and concatentes image features from the second-last and the last down-block of the U-Net, as well as the first upsampling block [28]. For the purpose of our study, this setup is not possible because a multitask architecture that pools from an upsampling block cannot be compared to its single-task classification equivalent, as the classification-only network does not have access to this additional contextual information. Figure 7 shows the joint classification and semantic segmentation U-Net architecture used in our study.

3.3.3 Joint loss

During training, the joint architectures output both the segmentation loss and the classification loss. The loss term for the joint tasks is $\lambda_{\text{seg}} \cdot \mathcal{L}_{\text{Dice}} + \lambda_{\text{cls}} \cdot \mathcal{L}_{\text{CE}}$.

The choice of 0.3 is motivated by past work that shows that this ratio yields strong performance in joint classification and semantic segmentation tasks [28, 35]. Although parametric methods for task weighting like uncertainty weighting [36] and GradNorm [37] also exist, they are typically used in scenarios where both tasks are being optimized. In our study, the segmentation path is used solely as an auxiliary task to regularize the encoder, rather than as a primary objective.

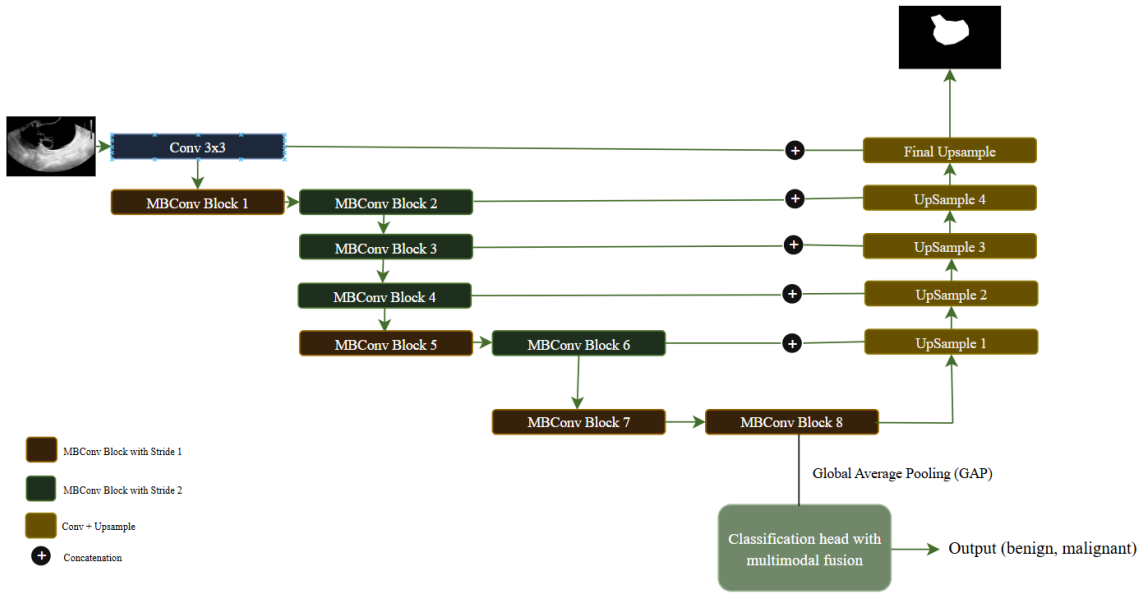


Figure 6. A joint classification and semantic segmentation network with an EfficientNetB0 backbone. This was done by adding an upsampling path and skip connections to the original EfficientNetB0 model.

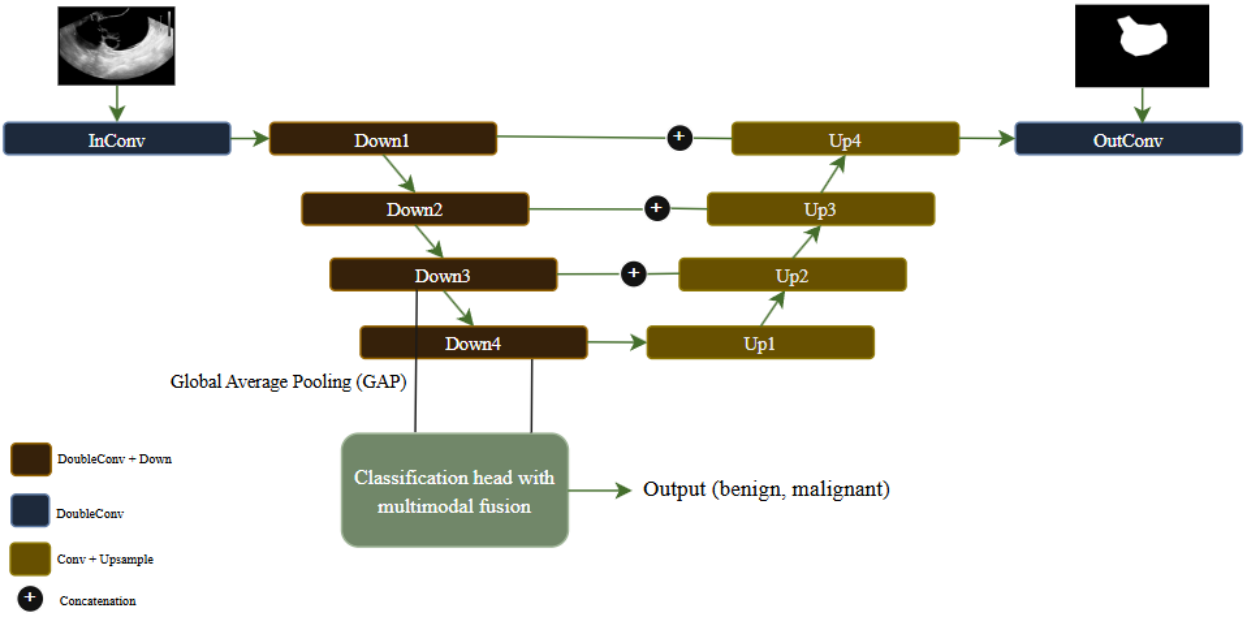


Figure 7. The Multitask U-Net-based architecture. A classification head is added to the traditional U-Net architecture, which performs global average pooling on the last two encoder blocks and concatenates these pooled features.

3.4. Evaluation Metrics

For classification-only and joint classification-segmentation networks, the following evaluation metrics are tested on the primary dataset:

- **Accuracy (%)**: the proportion of correctly predicted samples out of all samples.
- **Sensitivity (Recall)**: the proportion of true positives among all actual positives, computed as Sensitivity =

$\frac{\text{TP}}{\text{TP+FN}}$.

- **Specificity**: the proportion of true negatives among all actual negatives, computed as Specificity $= \frac{\text{TN}}{\text{TN+FP}}$.
- **AUC (Area Under the Curve)**: the area under the receiver operating characteristic (ROC) curve [38].
- **F1 score**: the harmonic mean of precision and recall.

For each trained model, the true positive (TP), true negative (TN), false positive (FP), and false negative (FN) rates are reported in Appendix C.

We compare the segmentation-only network to the MTL approach using the Intersection over Union (IoU) metric, which is defined as:

$$\text{IoU} = \frac{|\text{Prediction} \cap \text{Ground Truth}|}{|\text{Prediction} \cup \text{Ground Truth}|} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}}$$

The MMOTU dataset has 8 classes, compared to the primary dataset which has 2 classes. This makes it impractical to use metrics such as the F1 score or the AUC-ROC, which are binary classification metrics. For this reason, we solely report the IoU and accuracy results for the MMOTU dataset.

As for the primary dataset, the classification metrics used all rely on a threshold for the boundary, which creates a trade-off between sensitivity and specificity. This dependency, as well as the class imbalance present in the dataset, make it difficult to use these metrics to compare the different approaches. Furthermore, clinicians may adjust the threshold based on the individual patient or the perceived cost of false positives and false negatives [39]. For this reason, we use the Area Under the Curve (AUC) as our primary comparison metric, as it is independent of any specific threshold and not sensitive to class imbalances in the test set.

3.5. Datasets

3.5.1 Primary Dataset

This study includes data from 963 unique patients, collected across three different hospitals: Leiden University Medical Center (LUMC), Reinier de Graaf Gasthuis (RdGG), and Haga Ziekenhuis (HAGA).

Each patient in the dataset is associated with a variable number of 2D ultrasound slides, ranging from 1 to 20. Since slides from the same patient tend to be visually similar, dataset stratification was performed at the patient level to split the data into training, validation, and test sets. This approach ensures that the models learn to differentiate benign from malignant tumors based on morphological features rather than memorizing patient-specific characteristics. The 3895 benign and malignant images represent a total of 963 patients. Of these, 85% of patients were allocated to the training and validation sets, while the remaining 15% were used for testing.

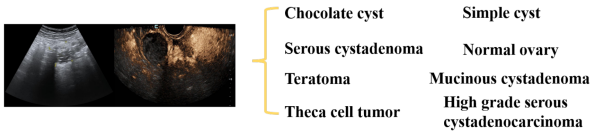


Figure 8. MMOTU classes

As lesion masks were necessary for this experiment, a joint effort was established to annotate lesion masks in the images of the dataset. This labeling effort includes contributions by the author of this thesis, Cor de Kroon, Maite Timmermans and Bart Groen. The Labelme [40] program was used to visualize, annotate and correct lesion masks.

3.5.2 MMOTU pre-training

In this study, we begin by pre-training our joint and single-task networks on the Multi-Modality Ovarian Tumor Ultrasound (MMOTU) dataset. The MMOTU dataset is an open online dataset with 1,469 ultrasound ovarian scans, each having class labels and corresponding semantic segmentation masks. This makes it possible to use the dataset to pre-train and evaluate multitask joint architectures.

The classification, segmentation and joint models trained on the MMOTU dataset are later fine-tuned on the primary dataset. Given the limited size of our own dataset, pre-training is useful because it means that the encoder can start with feature representations that are specific to ovarian lesions.

Furthermore, since MMOTU includes both classification labels and lesion masks, it allows us to assess the generalizability of multitask learning across datasets. By evaluating on MMOTU, we can better understand how multitask strategies perform in the context of ovarian mass classification in smaller datasets.

The eight classification classes present in the MMOTU dataset are shown in Figure 8.

4. Experimental Setup

4.1. Hyperparameters

Throughout the training process for the multi-task learning networks, the separate classification and the separate semantic segmentation networks, the set of hyperparameters remains constant. This was done in an effort to attribute differences in results to architectural choices instead of hyperparameter differences.

Appendix B shows the full list of hyperparameters employed for each dataset.

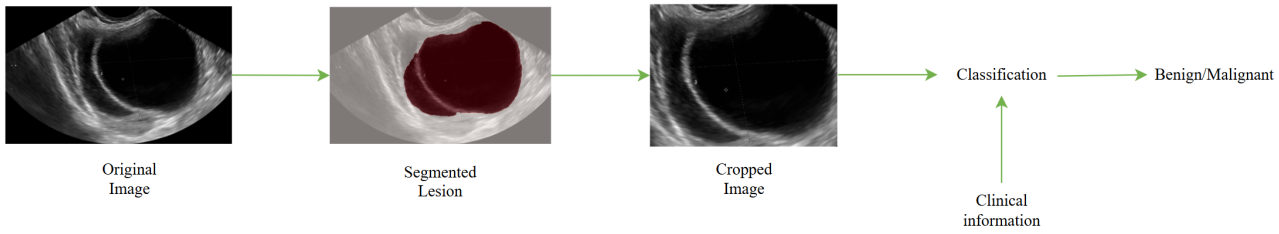


Figure 9. An overview of the multimodal cascaded approach. In this approach, the ovarian lesion is first segmented using the U-Net and the image is then cropped using this mask. This cropped image is resized to 164x164 and classification is then performed using clinical features.

4.2. Pre-processing of images

The full list of pre-processing steps is provided in Appendix C.

5. Results

5.1. MMOTU

The first evaluation of the multitask learning approach compared to single-task baselines was performed on the MMOTU dataset, in both original and denoised settings. Table 1 and 2 show the results. There were 233 test, 236 val and 1000 train images, which make a total 1469 images.

Architecture	IoU	Accuracy (%)
U-Net segmentation	0.66	–
U-Net classification	–	52.31
U-Net MTL	**0.70**	**57.76**
EfficientNet segmentation	0.70	–
EfficientNet classification	–	64.22
EfficientNet MTL	**0.75**	**69.83**

Table 1. MMOTU performance of the models in the original scenario

Architecture	IoU	Accuracy (%)
U-Net segmentation	**0.66**	–
U-Net classification	–	48.71
U-Net MTL	0.62	**64.66**
EfficientNet segmentation	0.65	–
EfficientNet classification	–	65.95
EfficientNet MTL	**0.71**	**73.71**

Table 2. MMOTU performance of the models in the denoised scenario

5.2. Primary Dataset

For the primary dataset, the IoU, sensitivity, specificity, precision, F1 score and AUC are computed for all entries in an independent test set comprised of 165, with a total of 731 slides.

Tables 3 to 6 show the results for the performance of the models with the U-Net backbone and the EfficientNet backbone, as well as the impact of adding clinical information, removing annotations and using multitask architectures instead of single-task equivalents.

5.3. Cascaded Approaches

In addition to exploring joint approaches for classifying and segmenting ovarian lesions, we also investigate a cascaded approach. In this approach, the lesion is first segmented, and the image is then cropped to only include the lesion. A classifier is then trained to only detect these lesion crops. This approach is motivated by the observation that the lesion mask is not strictly required for the final classification output. Figure 9 shows the full pipeline for the cascaded approach.

As the segmentation-only EfficientNetB0 backbone model performed the best on the MMOTU segmentation task, it was trained to localize lesions in the primary dataset then used to crop each image to only include its lesion. The cropped images were square-padded and resized to 164x164.

6. Discussion

6.1. MMOTU results

As the MMOTU dataset does not feature the needed clinical information, the impact of removing medical annotations and using auxiliary segmentation branches was evaluated, but not the impact of incorporating clinical features.

The results on the MMOTU dataset show clear benefits - in both U-Net architectures and EfficientNetB0 architectures, and in both original and denoised scenarios - of transforming a classification-network into a multitask learning network by adding an auxiliary segmentation path.

This drastic increase in accuracy across all scenarios could most likely be due to the relatively small size of the

Architecture	IoU	Accuracy (%)	Sensitivity (Recall)	Specificity	Precision	F1 Score	AUC
U-Net segmentation	**0.7651**	–	–	–	–	–	–
U-Net classification	–	86.59	0.4653	0.9302	0.5165	0.4896	0.8501
U-Net MTL	0.5914	84.95	0.5545	0.8968	0.4628	0.5045	0.8403
U-Net classification clinical	–	93.99	0.7030	0.8619	0.4494	0.5483	0.8767
U-Net MTL clinical	0.6408	**81.81**	0.7426	**0.8302**	**0.4121**	**0.5300**	**0.8899**

Table 3. Primary dataset performance of the U-Net models, with and without clinical data and on original images

Architecture	IoU	Accuracy (%)	Sensitivity (Recall)	Specificity	Precision	F1 Score	AUC
U-Net segmentation	**0.7590**	–	–	–	–	–	–
U-Net classification	–	87.96	0.4455	0.9492	0.5844	0.5056	0.8655
U-Net MTL	0.6879	82.63	0.6832	0.8492	0.4207	0.5208	0.8701
U-Net classification clinical	–	88.65	0.6337	**0.9270**	0.5818	0.6066	**0.9243**
U-Net MTL clinical	0.6638	**89.19**	**0.7426**	0.9159	**0.5859**	**0.6550**	0.9218

Table 4. Hospital performance of the U-Net models, without clinical data and on denoised images

Architecture	IoU	Accuracy (%)	Sensitivity (Recall)	Specificity	Precision	F1 Score	AUC
EfficientNet segmentation	**0.8422**	–	–	–	–	–	–
EfficientNet classification	–	81.94	0.7822	0.8254	0.4180	0.5448	0.8916
EfficientNet MTL	0.7878	85.77	0.7129	0.8810	0.4898	0.5806	0.9024
EfficientNet classification clinical	–	88.65	**0.8317**	0.8952	0.5600	0.6693	**0.9460**
EfficientNet MTL clinical	0.8014	**90.15**	0.8020	**0.9175**	**0.6090**	**0.6923**	0.9424

Table 5. Primary dataset performance of the EfficientNetB0 models, with and without clinical data and on original images

Architecture	IoU	Accuracy (%)	Sensitivity (Recall)	Specificity	Precision	F1 Score	AUC
EfficientNet segmentation	**0.8264**	–	–	–	–	–	–
EfficientNet classification	–	84.95	0.7921	0.8587	0.4734	0.5926	0.8972
EfficientNet MTL	0.8018	83.45	0.8020	0.8397	0.4451	0.5724	0.8987
EfficientNet classification clinical	–	87.82	0.7723	0.8952	0.5417	0.6367	0.9294
EfficientNet MTL clinical	0.8050	**91.24**	**0.8416**	**0.9238**	**0.6391**	**0.7265**	**0.9493**

Table 6. Hospital performance of the EfficientNetB0 models, without clinical data and on denoised images

MMOTU dataset. With such limited number of images and 8 classes, classification models may learn to rely on non-lesion regions of the scan for classification. In settings with so few images, encouraging the encoder to learn lesion-specific features through an auxiliary loss seem to improve the performance and reduce overfitting and reliance on irrelevant regions.

Removing medical annotations also led to improved classification performance, again because these annotations could act as confounders on which the network relied. This improvement may also be partly attributed to the limited size of the dataset, as neural networks trained on larger datasets might still learn lesion-specific features even in the presence of annotations.

Our results also demonstrate that using an EfficientNetB0 backbone outperformed the U-Net backbone for both classification and segmentation tasks. While this finding is specific to our use case, it suggests that researchers should not default to U-Net architectures when designing and evaluating multitask networks.

6.2. Primary Dataset

The results of our approach show that, when looking at the AUC as a metric, the addition of clinical information brought the best improvements to the classification metrics. In all cases, there was an increase in the AUC when clinical information was incorporated. This is likely because the menopausal status and the type of center offer additional contextual information to the classifiers when cases are ambiguous and image features are not enough to distinguish between benign and malignant cases.

Employing multitask architectures instead of single-task equivalents led to similar or slightly higher AUCs, but not in all cases. For example, using an efficientnet-based joint

Architecture	Accuracy (%)	Sensitivity (Recall)	Specificity	Precision	F1 Score	AUC
Full image clinical	87.82	0.7723	0.8952	0.5417	0.6367	0.9294
Cascaded	85.09	0.6634	0.8810	0.4718	0.5514	0.8912
Cascaded with clinical	**89.33**	**0.7921**	**0.9095**	**0.5839**	**0.6723**	**0.9367**

Table 7. Results of cascaded models, using the EfficientNetB0 classifier

architecture with denoised images and clinical information improved all classification metrics, as Table 6 shows. However, this improvement cannot be seen in all cases: for example, there was no improvement in the AUC when multitask architectures were used instead of single-task equivalents on original iamges.

The use of a multitask learning architecture as a regularizer proved beneficial across some settings, but their improvements were not seen consistently. The largest gain being an increase of 0.02 in denoised images with fused clinical and image features. Labeling ovarian lesions remains a time-consuming and tedious task, and the observed improvements may not justify the additional labeling annotation effort.

As Table 7 shows, the cascaded approach that first segments the lesion, crops it then classifies this cropped region performed better than running a classifier on the entire image. This is likely because the classifier is forced to only rely on the lesion for its outcomes, and cannot overfit on surrounding anatomical structures. However, even with this mechanism for explicit focus on the lesion, the cascaded approach performed slightly worse than the multitask learning approach, most likely because the cropping process was not always precise and was prone to over and under-cropping lesions.

In both the EfficientNet and U-Net architectures, the IoU was higher when using single-task semantic segmentation architectures. This is likely because there is a small amount of interference in the encoder when both tasks are learned simultaneously. Nonetheless, the segmentation path was employed to add an auxiliary loss to improve classification, and not with the goal of reaching satisfactory segmentation outcomes. Therefore, the observed drop in segmentation accuracy is an acceptable trade-off.

As per our hypothesis, when evaluating using the AUC metric, using an EfficientNet backbone for multitask learning brought improvements in all cases compared to the U-Net equivalent. The U-Net backbones also struggled with reaching satisfactory sensitivity (recall) scores when compared to their EfficientNet equivalents. This difference in classification performance could be a positive sign to rely on classification-tailored encoders in joint classification and semantic segmentation settings for clinical applications.

6.3. Clinical relevance of results

As the results indicate, the applications of deep learning to ovarian mass classification yield promising results, when the proposed improvements are implemented. As Table 6 shows, the combination of removing medical annotations, incorporating clinical information and employig a multitask network yielded an accuracy of 91%, a sensitivity socre of 0.84, a specificity score of 0.92, and an AUC of 0.95.

Segmenting the image lesion, cropping the image then classifying the crop also led to satisfactory classification outcomes; as Table 7 shows, this method yielded an accuracy of 89.33%, a sensitivity of 0.7921 and a specificity of 0.9095.

However, the benefit of mechanisms that encourage the encoder to learn lesion-specific features appears to be more pronounced in smaller datasets. The results show that the MMOTU dataset, which contains fewer images, exhibited greater improvement from the multitask learning approach compared to the primary dataset. This suggests that as dataset size increases, the classifier becomes more capable of learning morphological features directly, and the relative benefit of an auxiliary loss decreases.

Given that the gains from a joint architecture decrease when the size of the dataset increases, and given the labor cost and expertise needed to annotate semantic segmentation masks, it is generally more practical to improve classification performance by collecting additional images. In this clinical context, however, acquiring more ovarian scans is a tedious and costly process, mostly due to privacy and security constraints.

The addition of clinical information, namely the menopausal status and the type of center, proved to be crucial and providing the greatest benefit to the classifier. These two clinical variables are both readily available and routinely collected for each patient, and should be included in any model that aims to accurately classify ovarian tumors.

Although each patient has a variable number of ultrasound slides, the study was restricted to independent classification on single slides. Relying on single slides is dependent on the operator and can miss contextual information for a patient that is present in other slides. A simple extension that could increase the accuracy and reliability of the model would be to run the model on all patient slides and aggregate the results through majority voting. This ap-

proach would also give a measurable uncertainty metric for patients; when the model agrees on a diagnosis outcome for all patient slides, it could be deemed more certain than when its decisions are split between patient slides.

Given the promising performance of CNNs on 2D ovarian ultrasound scans, another potential extension would be to train CNNs on 3D volumes. This would reduce reliance on operator-selected slides and provide a more holistic view of the lesion to the classifier.

7. Conclusion

This study aimed to evaluate the performance of a multi-task network for the classification and semantic segmentation of ovarian lesions, with the segmentation path serving as a regularizer to guide the shared encoder toward learning lesion-specific features beneficial for downstream classification.

We demonstrated that, across most settings, joint architectures outperformed their single-task counterparts in classification. However, this advantage diminished as the dataset size increased, possibly showing that collecting more samples for a dataset might lead to the same effect as using joint architectures as regularizers.

Our results also indicated that the removal of medical annotations did not lead to significant performance changes, whereas fusing clinical and image features consistently led to significant improvements.

8. Future work

There are several potential approaches of improving the accuracy, robustness and explainability of the classifier, which could be explored in future work.

First, the incorporation of clinical information should not be limited to the menopausal status and center type. If available, biomarkers such as CA125 and CAE significantly enhance the accuracy of our approach. CA125 in particular is a strong indicator of malignancy in ovarian lesions and is the third clinical factor used in the IOTA Adnex model. As previously stated, we did not include these biomarkers in our study due to the lack of availability for all patients.

Another limitation of our study lies in the explainability of the models. CNNs are inherently difficult to interpret, and even saliency-based methods such as Grad-CAM have known limitations. Furthermore, the relative contribution of clinical versus image-based features should be evaluated to better understand their influence on the final prediction. Among the methods explored, the cascaded approach offers the greatest level of interpretability, as it allows clinicians to assess whether a misdiagnosis stems from incorrect lesion cropping or from misclassification of the cropped image.

Although our study demonstrates improvements when incorporating clinical features, their impact at inference time remains difficult to assess. As clinical models increasingly rely on multi-modal inputs rather than image data alone, recent work on multi-modal clinical explainability by Leiden University Medical Center (LUMC) [41] could offer valuable insights and be adapted to enhance the interpretability of our models.

References

[1] Junjie Huang, Wing Chung Chan, Chun Ho Ngai, Veeleah Lok, Lin Zhang, Don Eliseo Lucero-Prisno, Wanghong Xu, Zhi-Jie Zheng, Edmar Elcarte, Mellissa Withers, and Martin C. S. Wong. Worldwide burden, risk factors, and temporal trends of ovarian cancer: A global study. *Cancers*, 14(9):2230, April 2022. 1

[2] Freddie Bray, Mathieu Laversanne, Hyuna Sung, Jacques Ferlay, Rebecca L. Siegel, Isabelle Soerjomataram, and Ahmedin Jemal. Global cancer statistics 2022: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, 74(3):229–263, April 2024. 1

[3] Usha Menon, Aleksandra Gentry-Maharaj, Matthew Burnell, Naveena Singh, Andy Ryan, Chloe Karpinskyj, Giulia Carlino, Julie Taylor, Susan K Massingham, Maria Raikou, Jatinderpal K Kalsi, Robert Woolas, Ranjit Manchanda, Rupali Arora, Laura Casey, Anne Dawnay, Stephen Dobbs, Simon Leeson, Tim Mould, Mourad W Seif, Aarti Sharma, Karin Williamson, Yiling Liu, Lesley Fallowfield, Alistair J McGuire, Stuart Campbell, Steven J Skates, Ian J Jacobs, and Mahesh Parmar. Ovarian cancer population screening and mortality after long-term follow-up in the uk collaborative trial of ovarian cancer screening (ukctocs): a randomised controlled trial. *The Lancet*, 397(10290):2182–2193, June 2021. 1

[4] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. 2019. 2

[5] Yuyeon Jung, Taewan Kim, Mi-Ryung Han, Sejin Kim, Geunyoung Kim, Seungchul Lee, and Youn Jin Choi. Author correction: Ovarian tumor diagnosis using deep convolutional neural networks and a denoising convolutional autoencoder. *Scientific Reports*, 13(1), January 2023. 3

[6] Yuanheng Zhang, Nan Jiang, Zhaoheng Xie, Junying Cao, and Yueyang Teng. Ultrasonic image's annotation removal: A self-supervised noise2noise approach, 2023. 3

[7] Tsukasa Saida, Kensaku Mori, Sodai Hoshiai, Masafumi Sakai, Aiko Urushibara, Toshitaka Ishiguro, Manabu Minami, Toyomi Satoh, and Takahito Nakajima. Diagnosing ovarian cancer on mri: A preliminary study comparing deep learning and radiologist assessments. *Cancers*, 14(4):987, February 2022. 3

[8] Mingxiang Wei, Yu Zhang, Cong Ding, Jianye Jia, Haimin Xu, Yao Dai, Guannan Feng, Cai Qin, Genji Bai, Shuangqing Chen, and Hong Wang. Associating peritoneal metastasis with ¡scp¿t2-weighted mri¡/scp¿ images in epithelial ovarian cancer using deep learning and radiomics: A

multicenter study. *Journal of Magnetic Resonance Imaging*, 59(1):122–131, May 2023. 3

[9] Yida Wang, He Zhang, Tianping Wang, Liangqing Yao, Guofu Zhang, Xuefen Liu, Guang Yang, and Lei Yuan. Deep learning for the ovarian lesion localization and discrimination between borderline and malignant ovarian tumors based on routine mr imaging. *Scientific Reports*, 13(1), February 2023. 3

[10] S. Gueriero, M.A. Pascual, A. Piras, E. Musa, S. Ajossa, I. Rodriguez, M. Perniciano, L. Saba, V. Mais, and J.L. Alcazar. Cost-effective evaluation of magnetic resonance after use of simple rules in ovarian cancer. *Australasian Journal of Ultrasound in Medicine*, 22(2):145–145, April 2019. 3

[11] Filip Christiansen, Emir Konuk, Adithya Raju Ganeshan, Robert Welch, Joana Palés Huix, Artur Czekierdowski, Francesco Paolo Giuseppe Leone, Lucia Anna Haak, Robert Fruscio, Adrius Gaurilcikas, Dorella Franchi, Daniela Fischerova, Elisa Mor, Luca Savelli, Maria Àngela Pascual, Marek Jerzy Kudla, Stefano Guerriero, Francesca Buonomo, Karina Liuba, Nina Montik, Juan Luis Alcázar, Ekaterini Domali, Nelinda Catherine P. Pangilinan, Chiara Carella, Maria Munaretto, Petra Saskova, Debora Verri, Chiara Visenzi, Pawel Herman, Kevin Smith, and Elisabeth Epstein. International multicenter validation of ai-driven ultrasound detection of ovarian cancer. *Nature Medicine*, 31(1):189–196, January 2025. 3

[12] Xin He, Xiang-Hui Bai, Hui Chen, and Wei-Wei Feng. Machine learning models in evaluating the malignancy risk of ovarian tumors: a comparative study. *Journal of Ovarian Research*, 17(1), November 2024. 3

[13] Md. Martuza Ahamad, Sakifa Aktar, Md. Jamal Uddin, Tasnia Rahman, Salem A. Alyami, Samer Al-Ashhab, Hanan Fawaz Akhdar, AKM Azad, and Mohammad Ali Moni. Early-stage detection of ovarian cancer based on clinical data using machine learning approaches. *Journal of Personalized Medicine*, 12(8):1211, July 2022. 3

[14] B Van Calster, K Van Hoorde, W Froyman, J Kaijser, L Wynants, C Landolfo, C Anthoulakis, I Vergote, T Bourne, and D Timmerman. Practical guidance for applying the ADNEX model from the IOTA group to discriminate between different subtypes of adnexal tumors. *Facts Views Vis. ObGyn*, 7(1):32–41, 2015. 3, 4

[15] Songxiao Yang, Xiabi Liu, Zhongshu Zheng, Wei Wang, and Xiaohong Ma. Fusing medical image features and clinical features with deep learning for computer-aided diagnosis, 2021. 3

[16] Can Cui, Haichun Yang, Yaohong Wang, Shilin Zhao, Zuhayr Asad, Lori A Coburn, Keith T Wilson, Bennett A Landman, and Yuankai Huo. Deep multimodal fusion of image and non-image data in disease diagnosis and prognosis: a review. *Progress in Biomedical Engineering*, 5(2):022001, April 2023. 3

[17] Gregory Holste, Savannah C. Partridge, Habib Rahbar, Debosmita Biswas, Christoph I. Lee, and Adam M. Alessio. End-to-end learning of fused image and non-image features for improved breast cancer classification from mri. In *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 3287–3296, 2021. 4

[18] Shih-Cheng Huang, Anuj Pareek, Roham Zamanian, Imon Banerjee, and Matthew P. Lungren. Multimodal fusion with deep neural networks for leveraging ct imaging and electronic health record: a case-study in pulmonary embolism detection. *Scientific Reports*, 10(1), December 2020. 4

[19] A Sayasneh, L Wynants, J Preisler, J Kaijser, S Johnson, C Stalder, R Husicka, Y Abdallah, F Raslan, A Drought, A A Smith, S Ghaem-Maghami, E Epstein, B Van Calster, D Timmerman, and T Bourne. Multicentre external validation of iota prediction models and rmi by operators with varied training. *British Journal of Cancer*, 108(12):2448–2454, May 2013. 4

[20] P. G. Moorman, B. Calingaert, R. T. Palmieri, E. S. Iversen, R. C. Bentley, S. Halabi, A. Berchuck, and J. M. Schildkraut. Hormonal risk factors for ovarian cancer in premenopausal and postmenopausal women. *American Journal of Epidemiology*, 167(9):1059–1069, February 2008. 4

[21] B. Van Calster, K. Van Hoorde, L. Valentin, A. C. Testa, D. Fischerova, C. Van Holsbeke, L. Savelli, D. Franchi, E. Epstein, J. Kaijser, V. Van Belle, A. Czekierdowski, S. Guerriero, R. Fruscio, C. Lanzani, F. Scala, T. Bourne, and D. Timmerman. Evaluating the risk of ovarian cancer before surgery using the adnex model to differentiate between benign, borderline, early and advanced stage invasive, and secondary metastatic tumours: prospective multicentre diagnostic study. *BMJ*, 349(oct07 3):g5920–g5920, October 2014. 4

[22] Adrianne R. Mallen, Claire C. Conley, Mary K. Townsend, Ali Wells, Bernadette M. Boac, Sarah Todd, Anjalika Gandhi, Michelle Kuznicki, Bianca M. Augusto, McKenzie McIntyre, Brooke L. Fridley, Shelley S. Tworoger, Robert M. Wenham, and Susan T. Vadaparampil. Patterns and predictors of genetic referral among ovarian cancer patients at a national cancer institute-comprehensive cancer center. *Clinical Genetics*, 97(2):370–375, November 2019. 4

[23] Rasha Kamal, Soha Hamed, Sahar Mansour, Yasmine Mounir, and Sahar Abdel Sallam. Ovarian cancer screening—ultrasound; impact on ovarian cancer mortality. *The British Journal of Radiology*, 91(1090):20170571, October 2018. 4

[24] Evelyne M.J. Meys, Lara S. Jeelof, Bram L.T. Ramaekers, Carmen D. Dirksen, Loes F.S. Kooreman, Brigitte F.M. Slangen, Roy F.P.M. Kruitwagen, and Toon Van Gorp. Economic evaluation of an expert examiner and different ultrasound models in the diagnosis of ovarian cancer. *European Journal of Cancer*, 100:55–64, September 2018. 4

[25] Michael Crawshaw. Multi-task learning with deep neural networks: A survey, 2020. 4

[26] Jonathan Baxter. A model of inductive bias learning. *CoRR*, abs/1106.0245, 2011. 4

[27] Claudio E. von Schacky, Nikolas J. Wilhelm, Valerie S. Schäfer, Yannik Leonhardt, Felix G. Gassert, Sarah C. Foreman, Florian T. Gassert, Matthias Jung, Pia M. Jungmann,

Maximilian F. Russe, Carolin Mogler, Carolin Knebel, Rüdiger von Eisenhart-Rothe, Marcus R. Makowski, Klaus Woertler, Rainer Burgkart, and Alexandra S. Gersing. Multitask deep learning for segmentation and classification of primary bone tumors on radiographs. *Radiology*, 301(2):398–406, November 2021. 4

[28] Yue Zhou, Houjin Chen, Yanfeng Li, Qin Liu, Xuanang Xu, Shu Wang, Pew-Thian Yap, and Dinggang Shen. Multi-task learning for segmentation and classification of tumors in 3d automated breast ultrasound images. *Medical Image Analysis*, 70:101918, May 2021. 4, 7

[29] Simon Graham, Quoc Dang Vu, Mostafa Jahanifar, Shan E Ahmed Raza, Fayyaz Minhas, David Snead, and Nasir Rajpoot. One model is all you need: Multi-task learning enables simultaneous histology image segmentation and classification, 2022. 4

[30] Bruno Oliveira, Helena R. Torres, Pedro Morais, Fernando Veloso, António L. Baptista, Jaime C. Fonseca, and João L. Vilaça. A multi-task convolutional neural network for classification and segmentation of chronic venous disorders. *Scientific Reports*, 13(1), January 2023. 4

[31] Qi Zhao, Shuchang Lyu, Wenpei Bai, Linghan Cai, Binghao Liu, Guangliang Cheng, Meijing Wu, Xiubo Sang, Min Yang, and Lijiang Chen. Mmotu: A multi-modality ovarian tumor ultrasound image dataset for unsupervised cross-domain semantic segmentation, 2022. 4

[32] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. 4

[33] Shirin Kordnoori, Maliheh Sabeti, Mohammad Hossein Shakoor, and Ehsan Moradi. Deep multi-task learning structure for segmentation and classification of supratentorial brain tumors in mr images. *Interdisciplinary Neurosurgery*, 36:101931, June 2024. 4

[34] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, April 2004. 5

[35] Naga Raju Gudhe, Hamid Behravan, Mazen Sudah, Hidemi Okuma, Ritva Vanninen, Veli-Matti Kosma, and Arto Mannermaa. Area-based breast percentage density estimation in mammograms using weight-adaptive multitask learning. *Scientific Reports*, 12(1), July 2022. 7

[36] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics, 2017. 7

[37] Jae-Han Lee, Chul Lee, and Chang-Su Kim. Learning multiple pixelwise tasks based on loss scale balancing. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, page 5087–5096. IEEE, October 2021. 7

[38] Mark R. J. Junge and Joseph R. Dettori. Roc solid: Receiver operator characteristic (roc) curves as a foundation for better diagnostic tests. *Global Spine Journal*, 8(4):424–429, May 2018. 9

[39] Lang S et al Westwood M, Ramaekers B. Risk scores to guide referral decisions for people with suspected ovarian cancer in secondary care: a systematic review and cost-effectiveness analysis, 2018. 9

[40] Kentaro Wada, Mpitid, Martijn Buijs, N. Zhang Ch., , Bc. Martin Kubovčík, Alex Myczko, Latentix, Lingjie Zhu, Naoya Yamaguchi, Shohei Fujii, Iamgd67, IlyaOvodov, Akshar Patel, Christian Clauss, Eisoku Kuroiwa, Roger Iyengar, Sergei Shilin, Tanya Malygina, Kento Kawaharazuka, Jonne Engelberts, Aleksi J, AlexMa, Changwoo Song, , Charlie, Daniel Rose, Douglas Livingstone, , Doug, , Erik, and Henrik Toft. wkentaro/labelme: v4.6.0, 2021. 9

[41] Mafalda Malafaia, Thalea Schlender, Peter A. N. Bosman, and Tanja Alderliesten. Multifix: An xai-friendly feature inducing approach to building models from multimodal data, 2024. 13

3

Neural Networks

This chapter gives the relevant background information behind neural networks, convolutions and the process of using convolutional neural networks to learn to classify between images of different classes.

3.1. Perceptron

The most basic and fundamental building block of a neural network is the **perceptron**. Loosely inspired by the neurons in the human brain, a perceptron computes a linear weighted combination of its inputs and adds a bias term. More formally, the perceptron can be expressed as:

$$y = \phi\left(\sum_{i=1}^{n} w_i x_i + b\right)$$

where x_i are the input features, w_i are the corresponding weights, and b is the bias term. In this report, all input features, weights, and bias terms are floating-point numbers. To enable the perceptron to model non-linear relationships, its output is passed through a non-linear transformation known as an *activation function*. This activation function is represented by ϕ in the above formula for the perceptron. Common choices for activation functions include the Rectified Linear Unit (ReLU), the sigmoid function, and the step function. Figure 1 shows a single perceptron unit, known as a neuron.

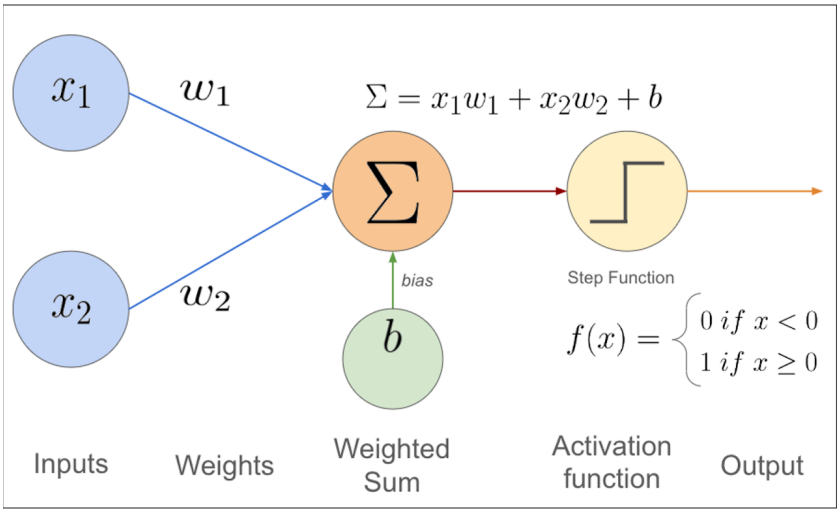


Figure 3.1: The perceptron, from [2]. The inputs *x1* and *x2* are weighted by the learnable parameters *w1* and *w2* and added together, as well as a bias term *b*. The output is passed through a non-linear activation function; in this example, it is a step function.

3.2. Multi-layer perceptron

While a single perceptron is capable of solving only linearly separable problems, its representational capacity to solve complex, non-linear and multi-dimensional problems is limited. the **Multi-layer Perceptron (MLP)** is a collection of multiple layers of connected perceptrons, each followed by a non-linear activation, that addresses these limitations.

Three types of layers constitute the building blocks of the MLP:

- **Input layer**: This layer receives the input features, and its size dictates the required dimensionality of the input. This layer does not perform any computation, and passes the input value to the next layer.
- **Hidden layers**: These are intermediate layers that each contain a given, fixed number of perceptrons. Each perceptron (known as a neuron) applies a weighted sum and an activation function of the output of all preceding neurons.
- **Output layer**: This layer produces the final output of the network. The number of output neurons depend on the nature of the task (e.g., regression, binary classification, multi-class classification).

The output of each neuron in a hidden layer becomes the input to neurons in the subsequent layer. For this reason, an MLP is also known as a **fully-connected network**. Mathematically, the transformation through a hidden layer can be expressed as:

$$\mathbf{h}^{(l)} = \phi\left(\mathbf{W}^{(l)}\mathbf{h}^{(l-1)} + \mathbf{b}^{(l)}\right)$$

where:

- $\mathbf{h}^{(l-1)}$ is the output of the previous layer (or input if $l = 1$),
- $\mathbf{W}^{(1)}$ is the weight matrix for layer l,
- $\mathbf{b}^{(l)}$ is the bias vector for layer l, and
- ϕ is the application function applied element-wise

The weight matrix $\mathbf{W}^{(l)}$ and the bias vector $\mathbf{b}^{(l)}$ for each layer l are the learnable parameters of the network, gradually adjusted during training so that the network can learn non-linear patterns in the data.

The multi-layer perceptron architecture can be seen in Figure 3.2.

3.3. Loss function

During training, a neural network requires a quantitative measure of how well it is performing on its task. This measure is provided by a **loss function**, which compares the model's predictions to the actual target values at each learning iteration. The loss function must be differentiable with respect to the network's parameters, as it is not only used to evaluate performance but also to guide the adjustment of the weights during training. At each iteration, the model moves closer to an optimal solution.

The loss function that is most essential to this report is the Cross-Entropy (CE) loss function. This loss function is used in classification tasks, and its binary (two-class) form can be written as:

$$\mathcal{L}_{\text{CE}} = -\left[y \log(\hat{y}) + (1-y)\log(1-\hat{y})\right]$$

where $y \in \{0, 1\}$ is the true label, and $\hat{y} \in (0, 1)$ is the predicted probability that the input belongs to class 1. In our report, class 0 refers to the benign class and class 1 refers to the malignant class.

The cross-entropy loss penalizes confident but incorrect predictions more heavily than less confident ones, making it particularly suitable for probabilistic classification models.

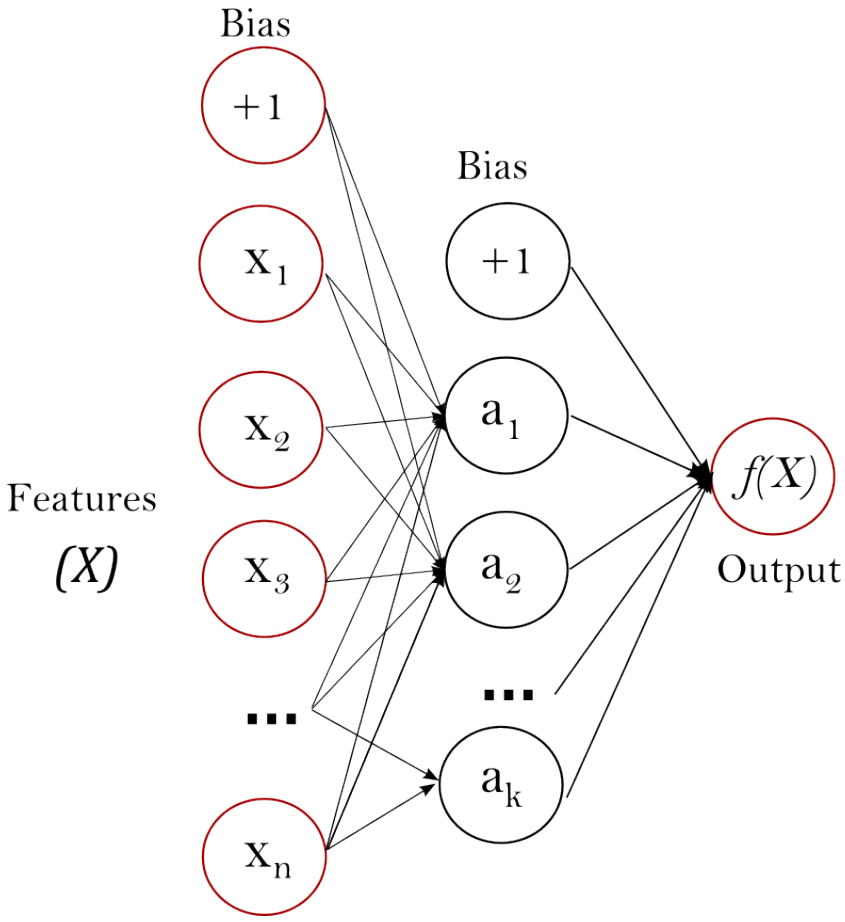


Figure 3.2: The multi-layer perceptron, from Scikit-Learn's documentation [1]. The set of neurons $\{x_i|x_1, x_2, ..., x_m\}$ represents the input features, and each intermediate (hidden) layer transforms the values from the previous layers through a weighted linear sum. In this example MLP, there is one hidden layer.

3.4. Optimizers

Once the network has calculated the loss using a loss function, it needs a way to update its parameters (weights and biases) to improve future predictions. This is the role of an **optimizer**. The optimizer looks at the current loss and adjusts the parameters in a direction that is expected to reduce the loss in the next iteration.

Most optimizers use a method called *gradient descent*, which relies on the gradient (or slope) of the loss function with respect to each parameter. These gradients tell the network how to change each parameter to reduce the loss. The amount by which the parameters are changed is controlled by a value called the **learning rate**, denoted by η: this is a small positive number that determines the step size during each update.

Assuming we use the binary cross-entropy loss function shown in section 3.3

$$\mathcal{L}_{\text{CE}} = -\left[y \log(\hat{y}) + (1-y)\log(1-\hat{y})\right],$$

where $y \in \{0, 1\}$ is the true label and $\hat{y} = f_\theta(x)$ is the predicted probability from the MLP with parameters θ, we can compute the gradient of the loss with respect to the parameters θ as

$$\nabla_\theta \mathcal{L}_{\text{CE}} = \frac{\partial \mathcal{L}_{\text{CE}}}{\partial \hat{y}} \cdot \frac{\partial \hat{y}}{\partial \theta}.$$

Then, using gradient descent, the parameters are updated according to:

$$\theta^{(t+1)} = \theta^{(t)} - \eta \cdot \nabla_\theta \mathcal{L}_{\text{CE}},$$

where $\theta^{(t)}$ is the parameter vector at iteration t.

Although the above formula describes the gradient descent optimizer, there are several variations of gradient descent used in practice. One of the most widely used is the **Adam optimizer** [6], which combines ideas from momentum and adaptive learning rates to make learning faster and more stable.

3.5. Backpropagation

Backpropagation is the algorithm used to efficiently compute the gradients of the loss function with respect to each parameter in a neural network. It applies the chain rule of calculus to propagate the error from the output layer back through the network, layer by layer.

In a feedforward neural network, each layer computes an output based on the inputs it receives from the previous layer. During training, after the forward pass computes the predictions and the loss is calculated, backpropagation begins the *backward pass*.

Suppose the output of a layer is given by

$$z = Wx + b, \quad a = \phi(z),$$

where W and b are the weights and biases, x is the input to the layer, z is the pre-activation, and ϕ is the activation function. The gradients are propagated backward using the chain rule:

$$\frac{\partial \mathcal{L}}{\partial W} = \frac{\partial \mathcal{L}}{\partial a} \cdot \frac{\partial a}{\partial z} \cdot \frac{\partial z}{\partial W},$$

and similarly for the biases:

$$\frac{\partial \mathcal{L}}{\partial b} = \frac{\partial \mathcal{L}}{\partial a} \cdot \frac{\partial a}{\partial z} \cdot \frac{\partial z}{\partial b}.$$

By applying these derivatives from the output layer to the input layer, backpropagation computes all necessary gradients to perform parameter updates using gradient descent or another optimization algorithm.

3.6. Convolutions

Multi-layer perceptrons are effective for many tasks, but their success is limited when applied to the task of image classification. This is because there are special structural patterns in images that are not representable by an MLP. For example, in images, nearby pixels (in the horizontal and vertical direction) are often more closely related than distance ones. Furthermore, specific patterns related to our objects of interest like edges or textures can appear in multiple locations.

To better capture these spatial relationships, we use a different kind of neural network called a **convolutional neural network**. Instead of relying on a linear combination of all previous neurons in a layer, a convolutional neural network uses the convolution operation to compute filter scores at each layer. This section explains the convolution operation, and the next one builds a simple convolutional neural network.

In convolutional layers, the basic building blocks are no longer individual perceptrons holding a single weight. Instead, they are small matrices called **kernels** or **filters**, which slide over the input data and perform a mathematical operation known as a **convolution**. These filters are learnable, meaning their values are adjusted during training just like weights and biases in fully connected layers.

Each filter is designed to detect specific patterns in the input. for example, one filter might return a strong output to vertical edges, while another might focus on corners. As the filter moves across the input, it produces a new output called a **feature map**, which highlights where the pattern associated with the filter is found.

Formally, a 2D convolution between a filter K and an input I can be expressed as:

$$S(i,j) = \sum_m \sum_n I(i+m, j+n) \cdot K(m,n)$$

where $S(i,j)$ is the output value at position (i,j), and the sums run over the dimensions of the filter K.

The process of sliding a kernel through an input image can be seen in Figure 3.3, where the convolution operation is performed.

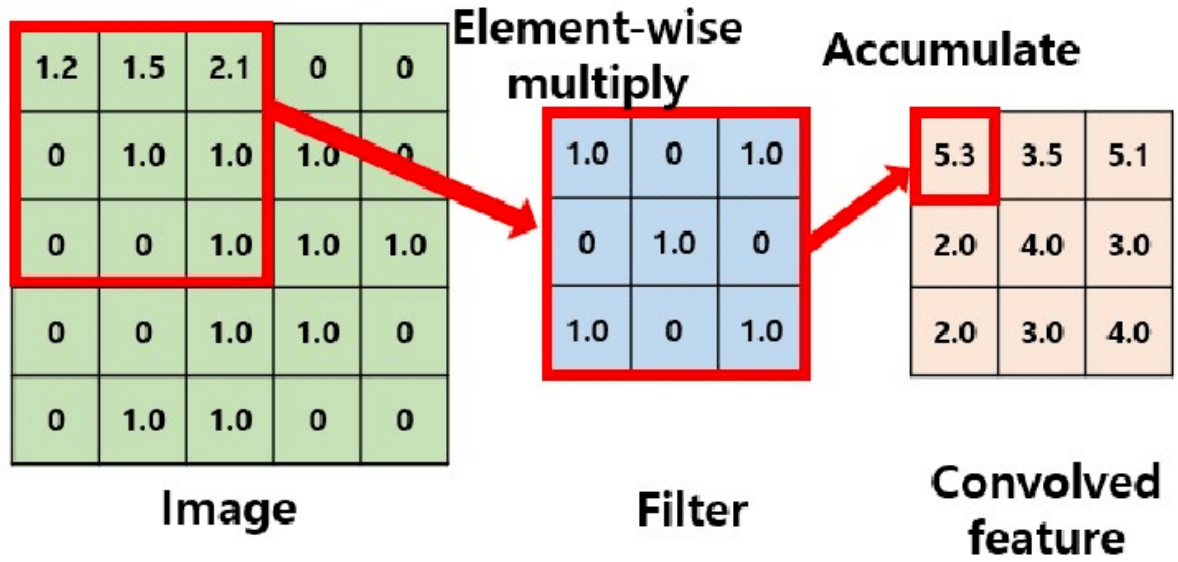


Figure 3.3: The convolution operation [5].

3.7. Convolutional Neural Networks

In this section, we use the definition of the convolution operator introduced previously to build a simple Convolutional Neural Network (CNN). While the CNNs discussed in the main report may differ in structure and depth, the fundamental building blocks described here provide sufficient background to understand their architecture and function.

Assume an input image of size 28×28 pixels with three color channels (Red, Green, and Blue). The first layer in our CNN is a convolutional layer with a set of learnable filters (also called kernels). Each filter processes all three color channels at once: it has one small 2D kernel for each channel, and the results from these are added together to form a single feature map. If the layer uses 6 filters, it will produce 6 such feature maps.

Once this feature block is generated, it is typically passed through a pooling layer to reduce its spatial dimensions while retaining the most relevant information. This is important because reducing the spatial size helps decrease the computational load and number of parameters, and more importantly, increases the receptive field. The receptive field is the region of the input image that affects a particular element in a feature map; increasing it allows subsequent filters to capture patterns from a larger context in the original image.

To achieve this, we introduce the max pooling operation. Max pooling partitions each feature map into non-overlapping 2x2 regions and outputs the maximum value from each region. This downsampling operation both reduces spatial dimensions and adds a form of translation invariance to the network.

Following the convolution and pooling layers, additional convolutional and pooling layers can be stacked to build hierarchical representations of the input. Eventually, the resulting feature maps are flattened and passed to a multi-layer perceptron, which performs the final classification.

Successive alternations of convolution and pooling operations are the core of nearly all modern CNN architectures. As an example, the LeNet architecture, which uses all described components so far, is shown in Figure 3.4.

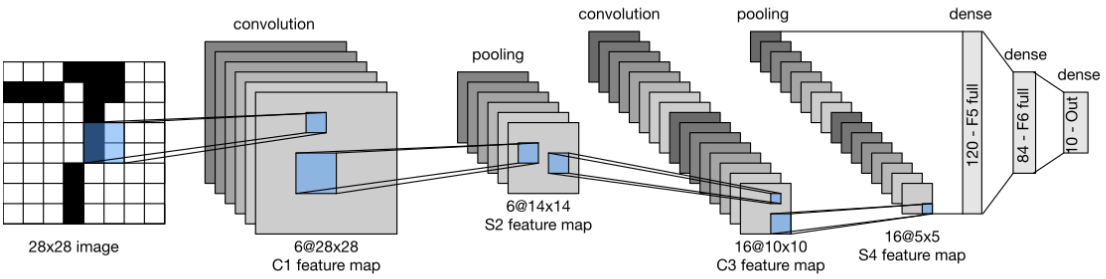


Figure 3.4: The LeNet-5 architecture, which was implemented and used in 1998 for digit recognition [7]. This architecture relies on a succession of convolution and max-pooling blocks that were described in this section.

3.8. EfficientNet

The single and multi-task architectures employed by this report are based on the traditional U-Net backbone and the EfficinetNetB0 backbone. In this section, the EfficientNetB0 architecture and its most important building block - **the MBConv block** - is explained. The next chapter will explain the task of semantic segmentation as well as the U-Net architecture.

EfficientNet is a family of convolutional neural networks developed by Google that achieves high accuracy with fewer parameters and less computations [10]. In order to understand the EfficientNet architecture, the most important aspect to understand is the MBConv block.

3.8.1. Depthwise convolutions

Contrary to the previously shown standard convolution operation, depthwise convolutions do not mix information across channels. The following example is an illustration of depthwise convolutions.

For the sake of the example, we use a 64x64x6 feature map. This means that the height of the feature map is 64, the width is 64 and the channels (depth) is 6.

Using a standard 2D convolution, each filter spans all 6 channels of the input feature map. This means that, using the standard 3x3 kernel, the actual filter size is 3x3x6. If 32 filters are used, the 64x64x6 feature map is transformed into a 64x64x32 feature map.

On the other hand, with depthwise convolutions, each input channel is assigned to one filter, and each filter is applied independently to this one channel. This means that, for a 3x3 kernel, there will be 6 filters in total for our feature map, each with dimensions 3x3x1. The final feature map will preserve the original 64x64x6 output shape.

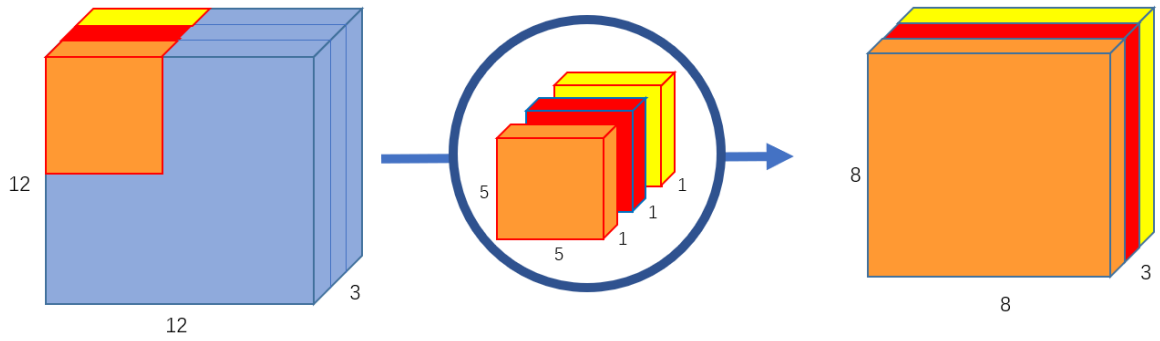


Figure 3.5: Depthwise convolution [8]. In this example, there are 3 learnable filters, each of depth 1, for an input feature block with 3 channels

3.8.2. Pointwise convolutions

As the example above demonstrated, in the example of a 64x64x6 input feature map, depthwise convolutions apply a separate filter to each channel and return a feature map of the same dimension. However, there is no information mixing between different channels in the input. In order to remedy this, the EfficientNet blocks use standard convolutions but with a 1x1 filter. These convolutions are called **pointwise convolutions**, and are applied across all channels. For example, using 32 filters would give the output shape 64x64x3.

The combination of depthwise and pointwise convolutions results in a much lower parameter count than standard 3×3 convolutions, and still enables effective mixing of spatial and cross-channel information.

3.8.3. Squeeze and excitation blocks

The squeeze and excitation block [4] is an efficient method of reweighting the channels of a feature map to emphasize important information. The process is as follows:

1. Squeeze: Global Average Pooling (GAP) is applied to each channel of the feature map, which condenses the information into a single numerical descriptor per channel. In Section 3.7, the max-pool operation was introduced. The global average pooling operation differs in that it computes the average of all values within each feature map, rather than taking the local maximum in a 2x2 filter. As a result, it produces a vector where each element represents the mean activation of a channel across its entire spatial (height, width) extent. The output of this operation is a one-dimensional vector that has the same number of elements as the number of channels in the input feature map.
2. Excitation: the resulting vector is a channel descriptor, with one real number per channel in the input feature map. This vector is passed through a two-layer fully connected network. The first fully-connected layer applies non-linearity through a ReLU activation, and the second one through a Sigmoid activation. This modified vector is now used to reweigh the original feature map through channel-wise multiplication.

3.8.4. MBConv block

Using all the previously-described components, the MBConv block can be constructed. Figure 3.6 shows the stacked components of the MBConv block, which follow a given order:

1. Pointwise convolution
2. Depthwise convolution
3. Squeeze-excitation block
4. Pointwise convolution
5. Residual connection: this is an identity mapping from the input signal before any processing, that is concatenated to the output of the last pointwise convolution

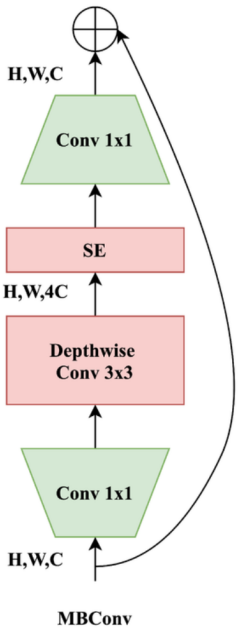


Figure 3.6: The MBConv block [3].

3.8.5. EfficientNetB0 architecture

The EfficientNetB0 architecture, as shown in Figure 3.7, is built using successive MBConv blocks. Instead of relying on traditional max-pooling operations for downsampling, EfficientNetB0 performs downsampling by setting the stride parameter of certain convolutional layers to 2. The stride of a convolution refers to the number of pixels the filter moves across the input. A stride of 2 means that every second pixel is sampled. Assuming the filter size and padding remain unchanged, this effectively reduces the spatial dimensions of the output by half.

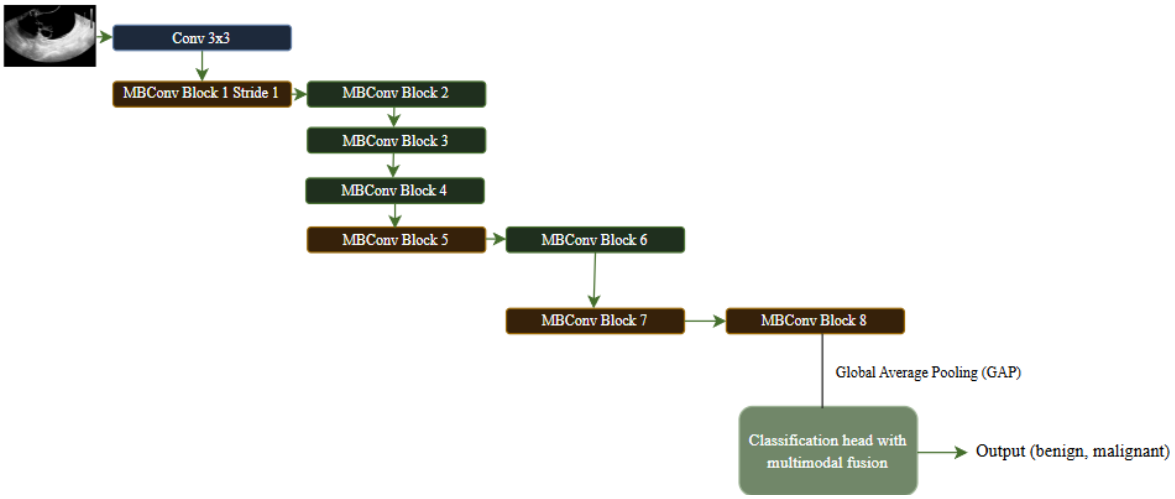


Figure 3.7: The EfficientNetB0 architecture.

4

Semantic Segmentation

In our report, we detect lesion masks of arbitrary shapes and contours by assigning all pixels in the image to either non-lesion or lesion classes. This task is referred to as semantic segmentation, a common application of convolutional neural networks (CNNs) in computer vision.

The U-Net remains one of the most widely used architectures for semantic segmentation, especially in medical image analysis. In this report, we use the U-Net architecture in two contexts: once as a denoising autoencoder that removes medical annotations from input images, and once as a segmentation model that produces pixel-level predictions of lesion masks.

4.1. Encoder path

The encoder path in semantic segmentation resembles a typical CNN used for image classification. It consists of a sequence of convolutional blocks, each typically followed by a non-linear activation function (such as ReLU) and a downsampling operation (usually max-pooling). With each successive block, the spatial resolution (height and width) of the feature maps decreases, while the number of feature channels increases.

4.2. Decoder path

After the encoder path, because of the repeated application of max-pooling operations, the resulting feature map is smaller in width and height than the original image. This resulting feature map is called the bottleneck. Unlike classification tasks, which typically flatten this feature map and apply fully connected layers, semantic segmentation tasks require a pixel-level output with the same spatial dimensions as the input image.

To achieve this, the decoder path progressively upsamples the feature maps using transposed convolutions or interpolation followed by convolution. These upsampling layers gradually reconstruct the spatial resolution of the feature maps while reducing the number of channels. The goal is to produce a full-resolution output map where each pixel contains a class prediction.

4.3. Skip connections

At the end of the encoder path, the spatial resolution is significantly reduced due to repeated pooling. While this bottleneck captures global context, it loses fine-grained spatial details that are essential for precise segmentation. This is especially important in medical settings, where small structures or boundaries matter.

To recover this lost spatial information, U-Net introduces skip connections between corresponding layers in the encoder and decoder at the same level. These connections concatenate feature maps from the encoder with the upsampled feature maps in the decoder. By reintroducing high-resolution features from earlier layers, skip connections help the decoder reconstruct accurate lesion boundaries and preserve finer details in the output map.

4.4. U-Net architecture

The U-Net architecture combines the encoder path, decoder path, and skip connections into a symmetrical U-shaped structure. As shown in the Figure 4.1, there is a corresponding upsampling block for each encoder block.

After the final upsampling operation, a 1×1 convolution is applied. This layer maps each pixel's feature vector to the desired number of output classes, producing a final segmentation map. The depth (i.e., number of channels) of this 1×1 convolution corresponds to the number of classes—in our case, two: lesion and non-lesion. In order to determine the final class label of each pixel, a pixel-wise argmax is applied as a last step.

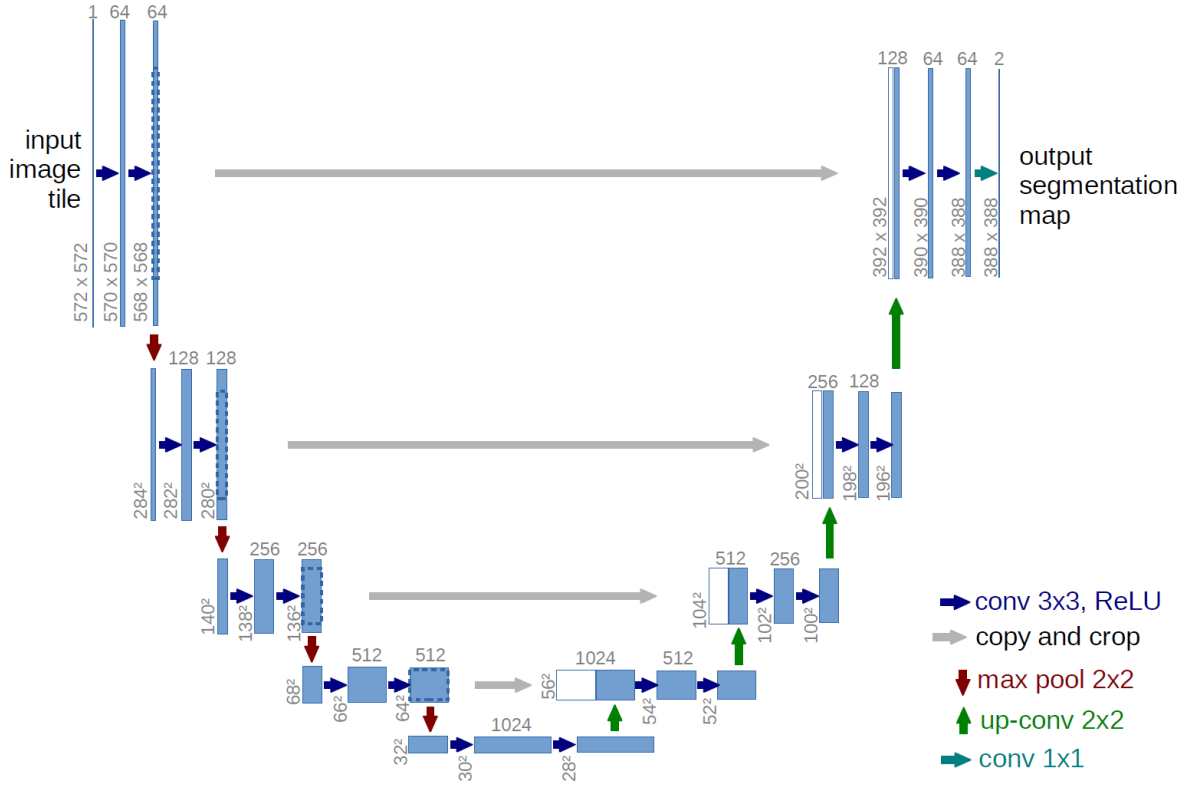


Figure 4.1: The U-Net architecture [9], which shows the componenets presented by this section: the encoder path (left) which resembles the encoding path of a traditional CNN used for classification, the bottleneck (center), the decoding/upsampling path(right), and the skip connections (grey arrows).

4.5. Loss function

The output of a semantic segmentation network is a feature map that has the same height and width as the original image, and a pre-determined depth. The depth of the feature map depends on the number of output classes. In our case, the feature map has 1 channel, as it only needs to indicate whether a pixel belongs to the lesion class or not.

In this binary setting, predictions are floating-point numbers between 0 and 1, with numbers being closer to 1 indicating a lesion pixel and numbers closer to 0 corresponding to non-lesion pixels. In order to compare the output of the predicted segmentation map with the actual labeled segmentation map, the **Mean Squared Error (MSE)** loss function can be used. For semantic segmentation, this function computes the average of the squared differences between the predicted values and the ground truth values for all pixels:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} (\hat{y}_{i,j} - y_{i,j})^2$$

where H and W are the height and width of the segmentation map, $\hat{y}_{i,j}$ is the predicted value at pixel (i, j), and $y_{i,j}$ is the corresponding ground truth value. In the binary setting, $y_{i,j} \in \{0, 1\}$ and $\hat{y}_{i,j} \in [0, 1]$.

MSE penalizes large differences between the predicted and true values, which encourages the model to produce outputs that closely match the ground truth segmentation. However, since most of the image contains non-lesion pixels, the model can easily be biased towards predicting mostly non-lesion pixels as a way of minimizing the MSE loss. For this reason, losses that are agnostic to the imbalance in classes have been devised for the task of semantic segmentation. In this report, we used the **Dice Loss**, which is expressed as follows:

$$\mathcal{L}_{\text{Dice}} = 1 - \frac{2 \sum_{i,j} \hat{y}_{i,j} y_{i,j}}{\sum_{i,j} \hat{y}_{i,j} + \sum_{i,j} y_{i,j} + \epsilon}$$

where $\hat{y}_{i,j}$ and $y_{i,j}$ are the predicted and ground truth values at pixel (i, j), respectively, and ϵ is a small constant added to prevent division by zero.

Dice Loss directly optimizes the overlap between the predicted segmentation and the ground truth, making it well-suited for imbalanced datasets where one class (e.g., lesion pixels) is much smaller than the other. A Dice Loss of 0 indicates perfect overlap.

References

[1] *1.17. Neural network models (supervised) — scikit-learn.org*. https://scikit-learn.org/stable/modules/neural_networks_supervised.html. [Accessed 20-06-2025].

[2] Muneeb S. Ahmad. *Deep Learning 101: Lesson 7: Perceptron — muneebsa.medium.com*. https://muneebsa.medium.com/deep-learning-101-lesson-7-perceptron-f6a698d81be8. [Accessed 20-06-2025].

[3] Abdul Basit et al. "Comparison of CNNs and Vision Transformers-Based Hybrid Models Using Gradient Profile Loss for Classification of Oil Spills in SAR Images". In: *Remote Sensing* 14.9 (Apr. 2022), p. 2085. ISSN: 2072-4292. DOI: 10.3390/rs14092085. URL: http://dx.doi.org/10.3390/rs14092085.

[4] Jie Hu et al. *Squeeze-and-Excitation Networks*. 2019. arXiv: 1709.01507 [cs.CV]. URL: https://arxiv.org/abs/1709.01507.

[5] Min-Jae Kim et al. "Functionality-Based Processing-in-Memory Accelerator for Deep Convolutional Neural Networks". In: *IEEE Access* 9 (2021), pp. 145098–145108. ISSN: 2169-3536. DOI: 10.1109/access.2021.3122818. URL: http://dx.doi.org/10.1109/ACCESS.2021.3122818.

[6] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2014. DOI: 10.48550/ARXIV.1412.6980. URL: https://arxiv.org/abs/1412.6980.

[7] Y. Lecun et al. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324. ISSN: 0018-9219. DOI: 10.1109/5.726791. URL: http://dx.doi.org/10.1109/5.726791.

[8] *Papers with Code - Depthwise Convolution Explained — paperswithcode.com*. https://paperswithcode.com/method/depthwise-convolution. [Accessed 23-06-2025].

[9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *CoRR* abs/1505.04597 (2015). arXiv: 1505.04597. URL: http://arxiv.org/abs/1505.04597.

[10] Mingxing Tan and Quoc V. Le. "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks". In: (2019). DOI: 10.48550/ARXIV.1905.11946. URL: https://arxiv.org/abs/1905.11946.

A

Examples

In this Appendix, examples of images with removed annotations are shown, as well as inference results of the joint architecture.

A.1. Removing medical annotations

Figures A.1 and A.2 show inference results when using the trained denoising autoencoder to remove medical annotations. As Figure A.1 shows, the arrow than spans the ovarian lesion was correctly inpainted, and there are no arrow artefacts visible in the image. However, the written text on the left side of the image has not successfully been removed, likely because the denoising autoencoder was not trained on this type of text. In Figure A.2, both the arrow that spans the lesion and the diameter marking on the top-left are successfully removed.

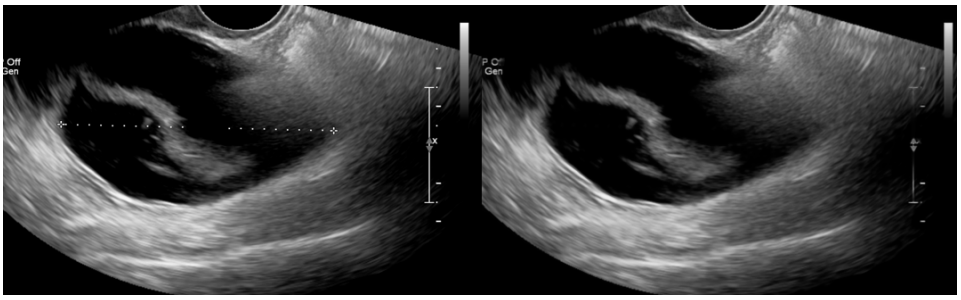


Figure A.1: A pre-menopausal patient from a non-oncology center being correctly classified as benign.

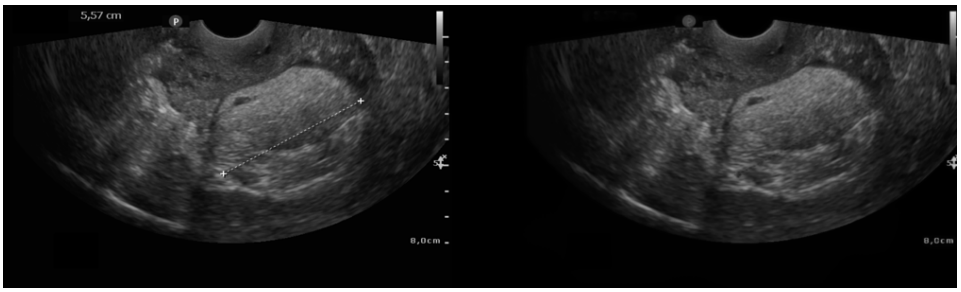


Figure A.2: A pre-menopausal patient from a non-oncology center being correctly classified as benign.

A.2. Inference results

The best results in the report were obtained using the following combination:

- Removing the annotations on images
- Incorporating clinical information
- Using multitask learning

As shown in section 5 of the report, this combination achieved a test accuracy of 91%, a sensitivity of 0.84, a specificity of 0.92, a precision score of 0.64, an F1 score of 0.73 and an AUC of 0.95.

In this section, we use the constructed classifier to run inference on some images in the test set to give examples of correct and incorrect predictions. These examples are present in Figures A.1-6.

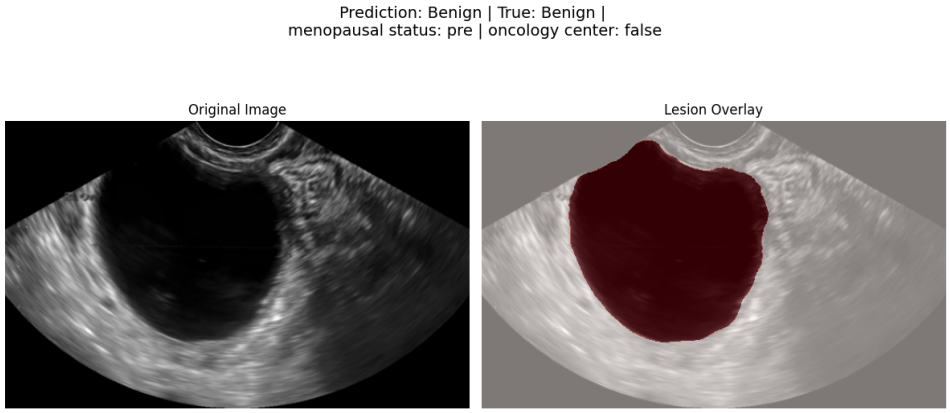


Figure A.3: A pre-menopausal patient from a non-oncology center being correctly classified as benign.

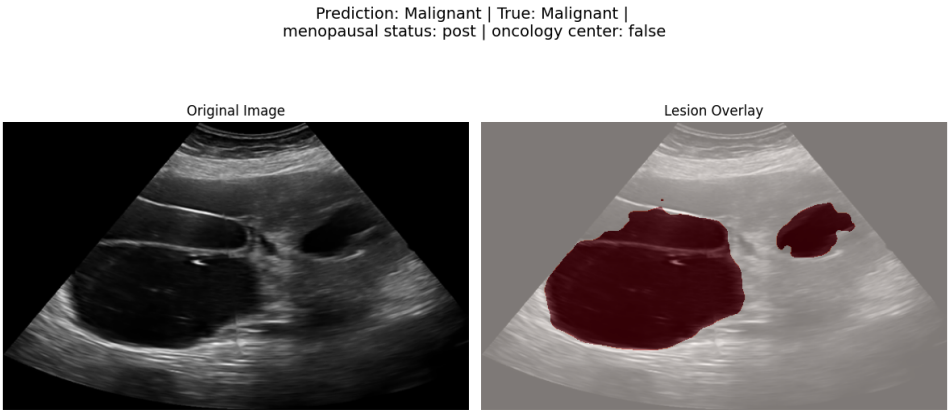


Figure A.4: A post-menopausal patient from a non-oncology center being correctly classified as malignant. However, there are some doubts regarding the correctness of the segmentation mask.

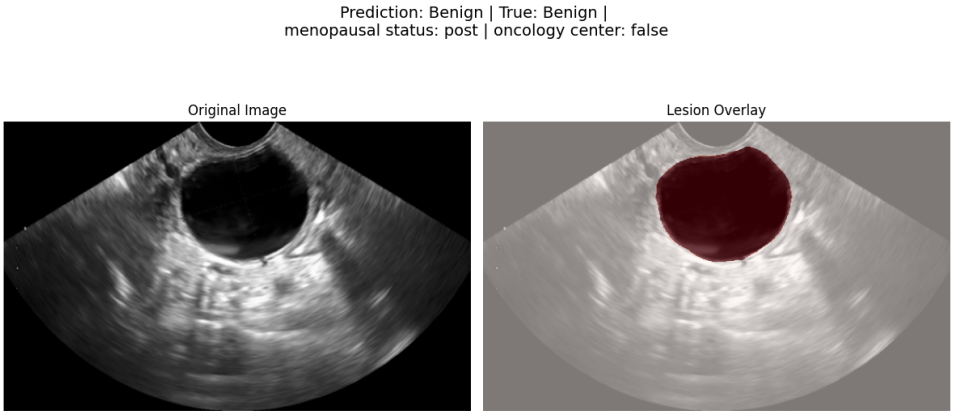


Figure A.5: A post-menopausal patient from a non-oncology center being correctly classified as benign.

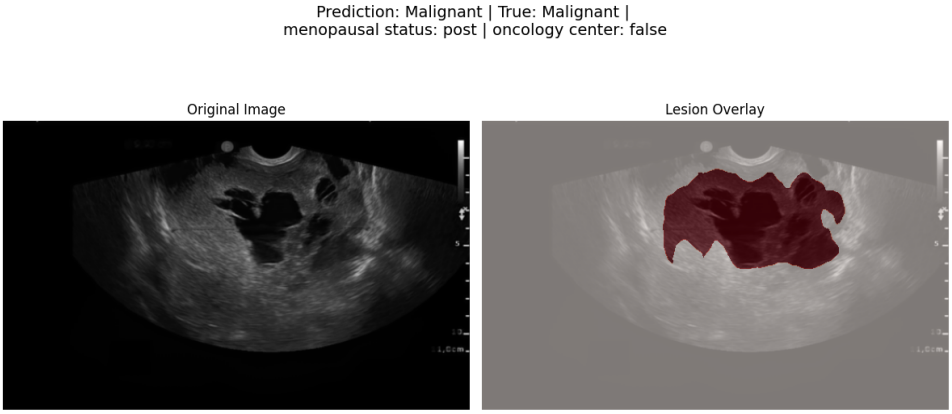


Figure A.6: A post-menopausal patient from a non-oncology center being correctly classified as malignant. However, there are some doubts regarding the correctness of the segmentation mask.

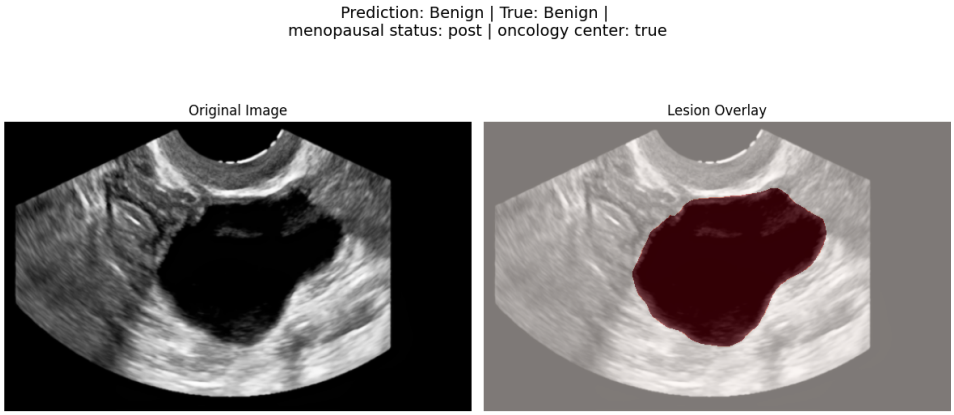


Figure A.7: A post-menopausal patient from an oncology center (LUMC) being correctly classified as malignant.

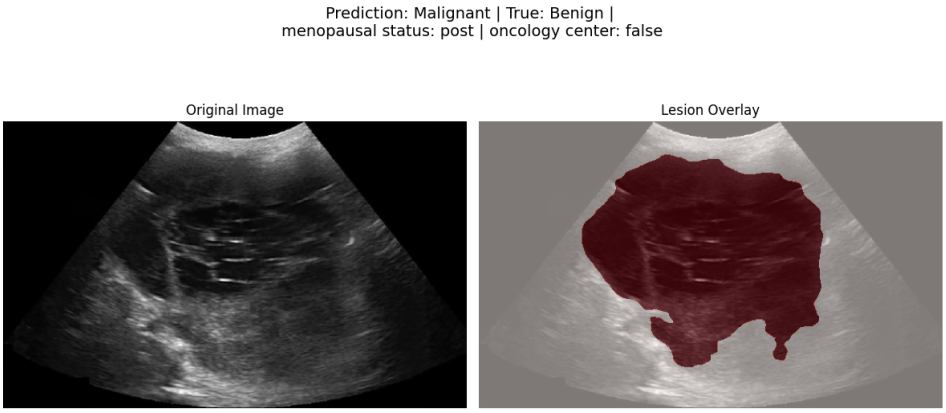


Figure A.8: A post-menopausal patient from a non-oncology center being incorrectly classified as malignant, while they are benign.

A.2.1. Example confusion matrices

Figure A.9 shows the confusion matrix for the baseline classifier on the test set. The upper-left corner represents correctly classified benign instances and the bottom-right corner represents correctly classified malignant instances. The confusion matrix also shows false positives (top-right) and false negatives (bottom-left).

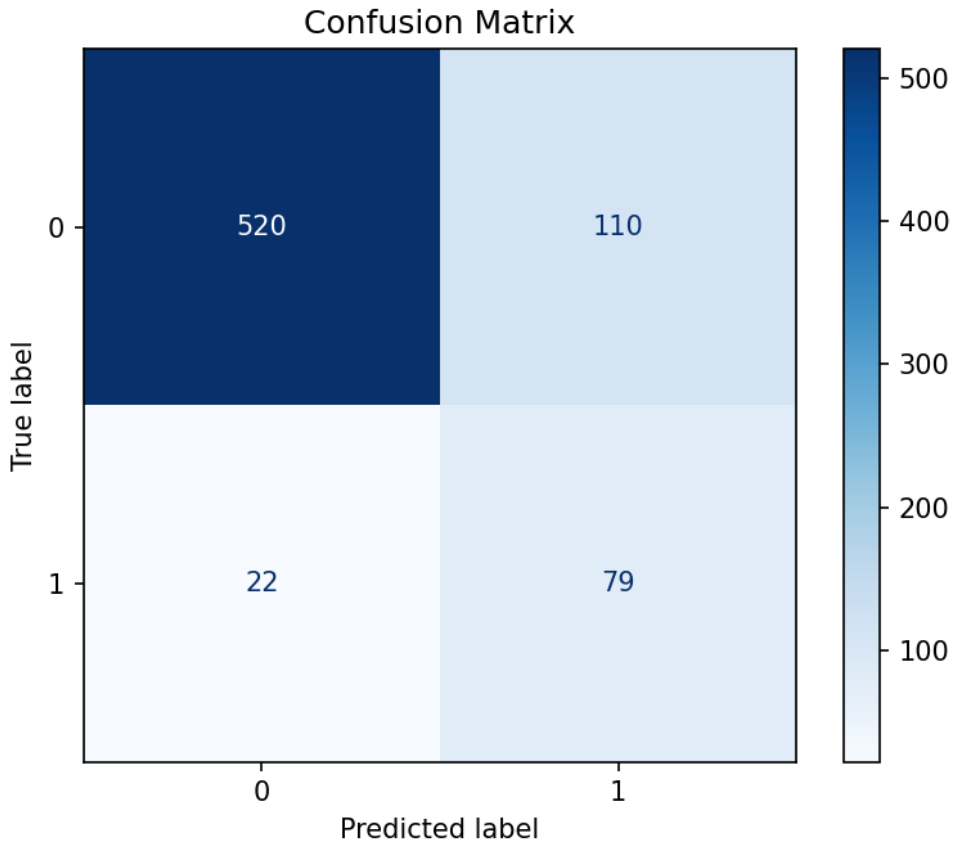


Figure A.9: The confusion matrix for the baseline classifier on the test set shown in the main report.

Figure A.10 shows the confusion matrix for the classifier with all incorporated improvements.

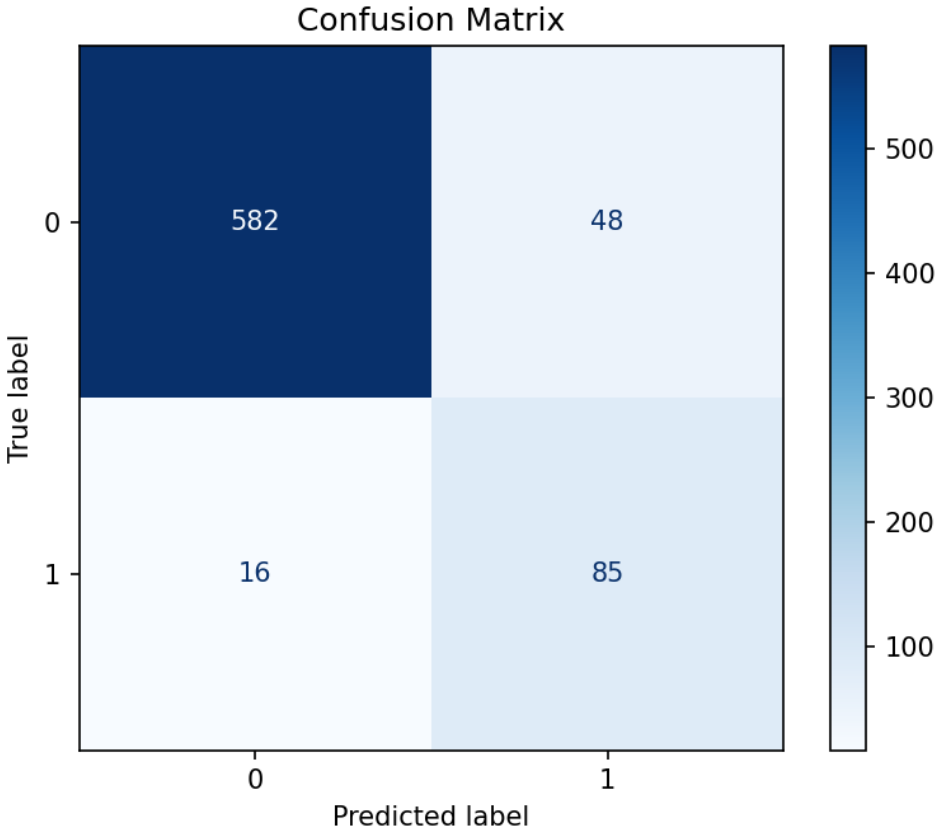


Figure A.10: The confusion matrix for the best-performing classifier on the test set shown in the main report. This classifier incorporates all three suggested improvements: the removal of medical annotations, the fusion of image and clinical features and the use of an auxiliary semantic segmentation path.

B

Hyperparameters

This section is a full overview of the hyperparameters used during the training stages, in both the MMOTU datasets and the primary dataset. This is done in an effort to ensure that the experiment is repeatable and the results are reproducible.

The hyperperameters are architecture-agnostic, meaning that the same set of hyperparameters is used for the U-Net and the EfficientNetB0 architectures. The nature of the task (joint vs. classification only) also does not change the set of hyperparameters, as the goal of the report is to attribute improvements in performance to changes in architecture and not differences in hyperparameters.

The set of hyperparameters for training the denoising autoencoder to remove medical annotations:

- Epochs: 200
- Batch size: 4
- Resize size: 336 x 544
- Optimizer: Adam with default PyTorch parameters
- Learning rate: 0.001
- Save strategy: model from the last epoch
- Hardware: NVIDIA RTX 6000

The set of MMOTU hyperaparameters is as follows:

- Epochs: 200
- Resize size: 336 x 544
- Batch size: 8
- Optimizer: SGD with momentum 0.9
- Learning rate: 0.001
- Save strategy: lowest validation loss
- Loss function classification: focal loss
- Loss function semantic segmentation: dice loss
- Loss balancing: $0.3 \cdot \mathcal{L}_{\text{Dice}} + 1.0 \cdot \mathcal{L}_{\text{Focal}}$.
- Train/test split: 85/15 (same train and test sets used in all settings)
- Hardware: NVIDIA Tesla P100

The set of hyperparameters for the training of networks on the primary (hospital) dataset is as follows:

- Epochs: 200
- Resize size: 336 x 544 (164 x 164 if cropped to the lesion mask)

- Batch size: 8
- Optimizer: Adam with default PyTorch parameters
- Learning rate: 0.001
- Save strategy: lowest validation loss
- Loss function classification: cross-entropy with class weights [1.0, 2.0]
- Loss function semantic segmentation: dice loss
- Loss balancing: $0.3 \cdot \mathcal{L}_{\text{Dice}} + 1.0 \cdot \mathcal{L}_{\text{CE}}$.
- Train/test split: 85/15 (same train and test sets used in all settings)
- Hardware: NVIDIA RTX 6000

C

Pre-processing of images

Although pre-processing code is provided, this section describes all necessary pre-processing steps in plaintext.

The original images were ultrasound scans supplied in the form of DICOMs, which are Digital Imaging and Communications in Medicine files. This is a standard format used for storing, transmitting, and handling medical imaging data that includes both image information and associated metadata (such as patient details, imaging modality, and acquisition parameters).

The preprocessing steps are as follows:

1. Use the metadata about the manufacturer name and model name to accordingly crop out patient-identifying information and metadata. Each ultrasound model appends text metadata of a certain height, and this can be deterministically removed. For example, the Philips Affiniti 70W requires 40 pixels to be cut from the top of the image. The script that performs this step was written by Floris Luitjes, a previous medical student who worked on the early steps of this project.
2. Annonymize the DICOM, to remove patient-identifying metadata. This script was also written by Floris Luitjes.
3. Convert the DICOM to PNG.
4. Use OpenCV's binary thresholding and contour detection functions to locate the ovarian scan.
5. Change every pixel outside the detected ovarian scan to black in the image.
6. Expand the ovary to occupy the entire width and height of the image.
7. If medical annotations need to be removed, use the denoising autoencoder to infer the clean image.
8. If the patient is perimenopausal, change the status to postmenopausal.