MILF: Malignant Interface via Learned Features

Eleonora Liani (2078147), Alessia Toska (2063682), Aditi Das (2082184), Gerda Lukosiute (2085763)

Abstract

The MILF project presents a binary image classification system for breast cancer detection using a combination of computer vision techniques. Through the use of histopathological images, our model aims to distinguish between benign and malignant tissue samples. The approach involves training a convolutional neural network on image data as well as optimizing preprocessing steps and model architecture to enhance feature extraction and generalization. Our goal with this project is to significantly improve classification accuracy and support early cancer detection through automated image analysis.

1. Introduction

Breast cancer is characterized by unique epidemiological patterns, clinical manifestations and intricate pathogenesis (1). Fig. 1 shows that breast cancer surpassed lung cancer as the most commonly diagnosed cancer in women in 2020 (3).

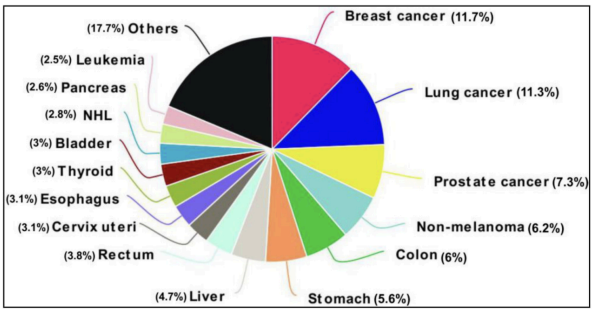

Figure 1. Global cancer statistics (2020), showcasing breast cancer's prominence among women (Data from GLOBOCAN 2020) (3)

In 2022, 2.3 million new cases and 670,000 deaths from breast cancer occurred globally (2). The mechanisms of tumorigenesis and progression of breast cancer have been central to scientific research, with investigations spanning various perspectives, such as tumor stemness, intra-tumoral microbiota, and circadian rhythms. Manual classification of histopathological images remains a difficult problem due to various limitations. Examination requires the professional background and rich experience of pathologists, which may increase the costs of diagnosis, the time it takes, or cause error. Technological advancements, particularly those integrated with artificial intelligence, have significantly improved the accuracy of breast cancer detection (1, 3).

1.1. Project Goals

In this project we aim to use a broad dataset of histopathological slides at various magnification levels to produce a user-friendly interface with a built-in binary classification model for breast cancer. We seek to build upon and improve the previous versions of similar classification systems that were based on the same data. In particular, a previous experiment on the same dataset performed by Chavengsaksongkram et al. reported that trying to generalize a single model across magnification classes fails due to poor convergence of training and validation accuracy. Thus, they settled on mainly exploring the 40X magnification class (9). Hence, we were inspired to try various approaches across classes and discuss how they affect the overall performance of our model, as well as provide possible reasons and solutions to problems we encounter.

1.2. Dataset Selection

Here we use the breast cancer image classification dataset (BreakHis) compiled by Spanhol et al. The data is in the form of images from slides of breast tissue, classified into benign and malignant samples. There are 4 subtypes of benign tumors (namely adenosis, fibroadenoma, phyllodes tumor and tubular adenoma) and the same amount of malignant subtypes (ductal carcinoma, lobular carcinoma, mucinous carcinoma and papillary carcinoma). In total, it contains 7909 images. Each subtype is captured under 40X, 100X, 200X and 400X magnification and comes from 82 patients. There are approximately 31 percent (2480) of benign and 69 percent (5429) of malignant samples (4).

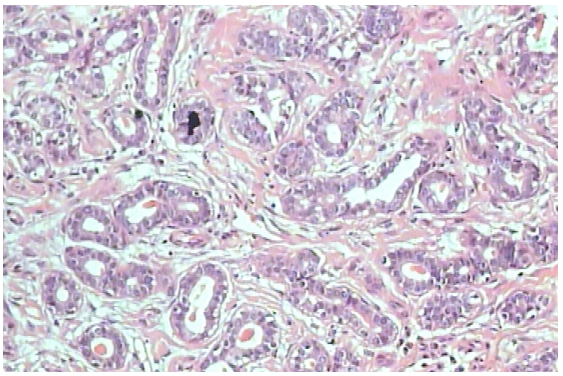

Figure 2A. A histopathological slide of a benign tumor of subtype adenosis under 100X magnification (4).

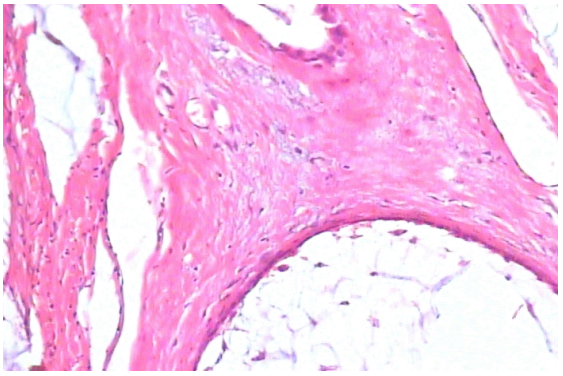

Figure 2B. A histopathological slide of a malignant tumor of subtype mucinous carcinoma under 100X magnification (4)

2. Methods

Due to its seamless integration of GPU resources, user-friendly notebook interface

and easy to access datasets we decided to use Kaggle as our development environment. Kaggle provides free limited access to powerful NVIDIA Tesla GPUs, which significantly accelerated model training and evaluation. In addition, it has built-in support for popular Python libraries that were extensively used in our project.

2.1. Data Preprocessing

To ensure consistency and prevent data leakage, we performed a structured preprocessing of the BreakHis dataset. Initially we collected the image paths from each class (i.e. benign/malignant) across the possible magnification levels. We tried limiting the number of images per class across the magnifications to maintain class balance and manageable training time. We postulated that the maximum number of images we can safely use per class is 600, given by the number of images in the smallest category divided across magnifications (i.e. ~2400/4). Patient identifiers were extracted from file paths and used to split the data into training (70%), validation (15%), and test sets (15%). This ensured no patient overlap was occurring across splits. During training we applied data augmentation techniques to increase variability and reduce overfitting. These included random horizontal and vertical flips, rotations, resizing to 224x224 pixels, and normalization. Finally, we experimented with different batch sizes (e.g., 32/64) to optimize GPU utilization and training stability.

2.2. Model Selection and Training

For this task we leveraged the ResNet-50 convolutional network for binary classification of histopathology images. ResNet-50 is a pre-built model which has been trained on the ImageNet dataset across different images of 1000 classes. It was chosen due to its proven effectiveness in medical imaging tasks and deep residual architecture (5). We initialized the model with ImageNet weights and applied transfer learning: the early convolutional layers were kept frozen to retain general image features, while the final block (layer4) and fully connected layers (fc) were fine-tuned to adapt to our task. We trained the model on augmented data using PyTorch. Pixel values were normalized to match ImageNet statistics. During training we used the Adam optimizer (learning rate = 1e-4, weight decay = 1e-4) and cross-entropy loss weighted by class frequency to correct for class imbalance. Training was performed up to 20 epochs, with early termination in case of non

improvement of the best validation F1-score for 5 consecutive epochs. This ensured we manage to avoid overfitting and save time on execution. The procedure was performed across all magnifications, using different batch sizes and different training set sizes.

2.3. Model Evaluation

After training the model, performance was assessed using the test dataset splits specific to each magnification and configuration. Three primary metrics were used:

1. **Confusion matrix**

A matrix whose entries are defined and arranged as follows (6):

1.1. True Positives (TP) - the number of correctly classified samples.

1.2. True negatives (TN) - the number of correctly classified negative samples.

1.3. False Positives (FP) - the number of samples incorrectly classified as positive.

1.4. False Negatives (FN) - the number of samples incorrectly classified as negative.

$$\mathbf{M} = \begin{pmatrix} \mathrm{TP} & \mathrm{FN} \\ \mathrm{FP} & \mathrm{TN} \end{pmatrix},$$

Figure 3. Visualization of the confusion matrix (6)

2. **Accuracy**

Ratio between the correctly classified samples and the total number of samples in the evaluation dataset (6).

$$\mathrm{ACC} = \frac{\#\,\text{correctly classified samples}}{\#\,\text{all samples}} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FP} + \mathrm{TN} + \mathrm{FN}}$$

Figure 4. Formula for accuracy calculation (6)

3. **Weighted F1 score**

This metric calculates the F1 score for each class, but the average is weighted by the number of true instances each class has (7).

$$F1_{\text{weighted}} = \sum_{i=1}^{N} w_i \frac{2 \times Precision_i \times Recall_i}{Precision_i + Recall_i}$$

$$w_i = \frac{TP_i + FN_i}{\sum_{j=1}^{N} (TP_j + FN_j)}$$

Figure 5. Formula for weighted F1 score calculation (7)

In addition to using the aforementioned metrics on the final model, we tracked F1 score and accuracy during each training epoch to monitor model progression. The best models for each magnification class were selected based on the highest validation F1 score. A training log plot of accuracy across epochs was saved for comparing training and validation accuracies.

2.4. GradCAM

To enhance the interpretability of the models, we applied Gradient-weighted Class Activation Mapping (GradCAM). GradCAM can provide insights into the decision-making process of the model by highlighting the regions of an image that are most important in making a particular prediction. Similarly, it also highlights the tricky areas (8). For each run of training, we chose an image with the corresponding magnification to visualize the GradCAM heatmap alongside the model's prediction.

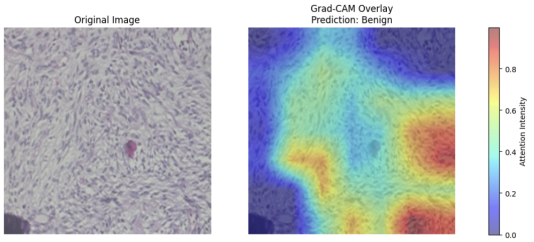

Figure 6. A histopathology slide of a benign phyllodes tumor at 100X magnification next to its GradCAM overlay showing the models prediction

2.5. GUI

We developed a user-friendly web-based Graphical User Interface (GUI) using Streamlit to interact with our deep learning model. The interface allows users to upload a microscopic image of breast tissue and receive a prediction on whether the sample is benign or malignant. We also introduced dynamic support for multiple magnifications: 40X, 100X, 200X, and 400X. To simplify the interface and improve the classification reliability, the GUI allows users to manually select the magnification level of the uploaded image. Based on this selection, the system loads the appropriate deep learning model trained specifically for that resolution, ensuring optimal performance. This approach avoids the need for automatic magnification detection and reduces the risk of mismatched model inference.

Overall, the GUI provides the following features:

1. Upload interface for JPG, PNG, or JPEG histological images.
2. Manual Magnification Selection: 40X, 100X, 200X, 400X.
3. Real-Time Prediction: clear classification output (Benign or Malignant).
4. Output of model performance metrics including Accuracy and F1-score specific to the magnification used.

3. Results

With each trial the configuration of batch size and number of images per class was altered. After numerous trials, we concluded that the best results are usually produced by setting the batch size to 32 or 64 and the number of images to 500 or 600. Hence, moving forward we alternated between the possible combinations of

these values to and attain the balance which provides the best outcome.

3.1. 40X Magnification

The best outcome for this category of magnification was achieved using a batch size of 64 and 500 images per class. This model demonstrated a validation accuracy of 86.72% and achieved a training accuracy exceeding 99% by the final epoch. The accuracy curve represents a stable overall training performance, however the convergence between training and validation accuracy remained quite poor and indicative of slight overfitting (Figure 7). The confusion matrix displays strong classification performance, with 93% of malignant samples and 81% of benign samples being correctly identified (Figure 8). This suggests good generalization, particularly for malignant samples, which is favorable in a clinical setting.

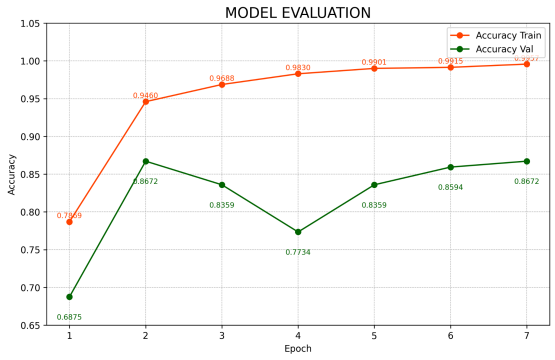

Figure 7. Model evaluation graph of the best performing model in the 40X magnification group

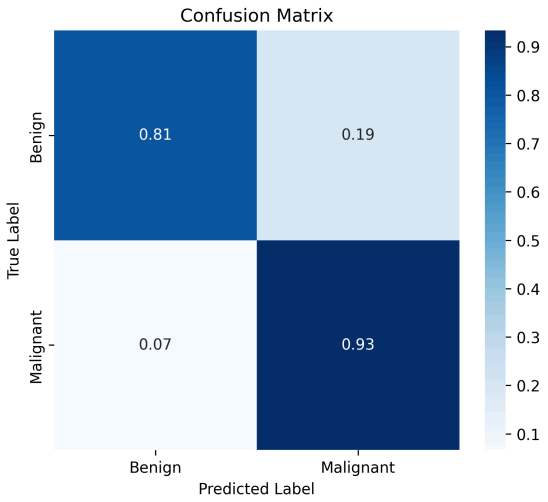

Figure 8. The confusion matrix for the best performing configuration of the 40X magnification model

3.2. 100X Magnification

The best performance for this magnification class was achieved using a batch size of 64 and 600 images per class. As shown in the model evaluation plot (Figure 9), the training accuracy steadily increased to 96.9%, while the validation accuracy peaked at 94.4%. In this case, there was less fluctuation in the validation curve than in the best 40X magnification model, suggesting better convergence. The confusion matrix shows that the model correctly classified 85% of malignant cases, and 80% of benign cases (Figure 10).

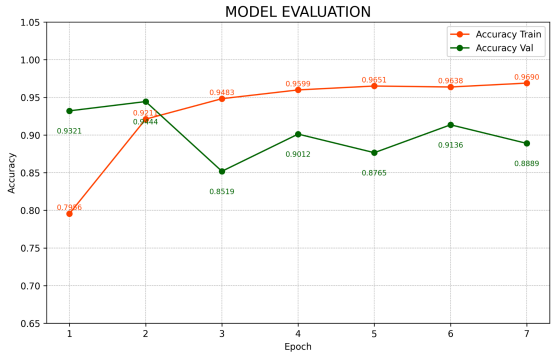

Figure 9. Model evaluation graph of the best performing model in the 100X magnification group

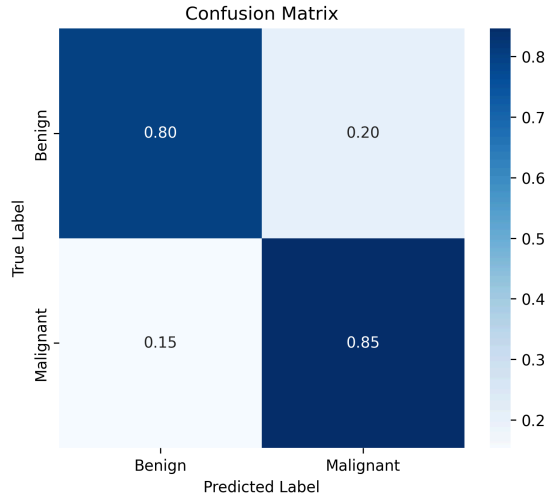

Figure 10. The confusion matrix for the best performing configuration of the 100X magnification model

3.3. 200X Magnification

In this class we had a similar performance of 2 configurations. The first one had a batch size of 64 and 500 images per class, and showed a better benign class accuracy of 95% (Figure 12). However, it was indicative of overfitting, due to low convergence of validation and training accuracy and a sharp drop in validation stability at epoch 10 (Figure 11).

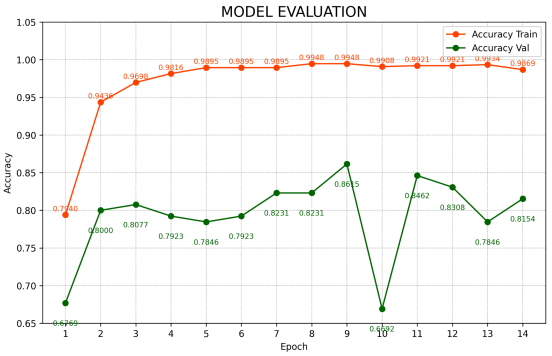

Figure 11. Evaluation graph of the 200X_b64_n500 model

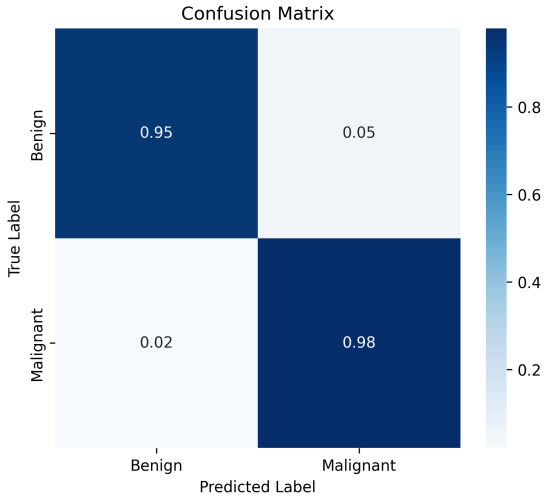

Figure 12. The confusion matrix of the 200X_b64_n500 model

The second configuration at this magnification had a batch size of 32 and the same amount of images per class. Here, the validation and training accuracy convergence is also indicative of slight overfitting, but it lacks the sharp drop in validation stability (Figure 13). In contrast, it has a lower benign precision of 88%, but maintains the same malignant precision value (Figure 14). Given this discrepancy we chose to go with the second one, indicative of less overfitting.

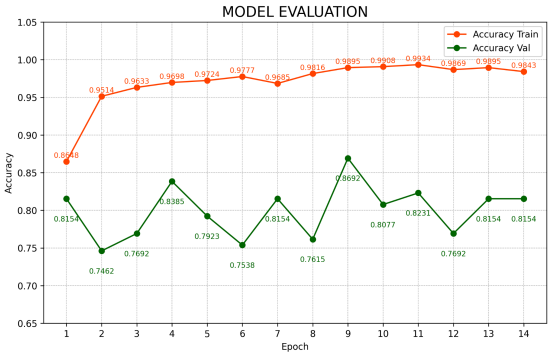

Figure 13. Evaluation graph of the 200X_b32_n500 model

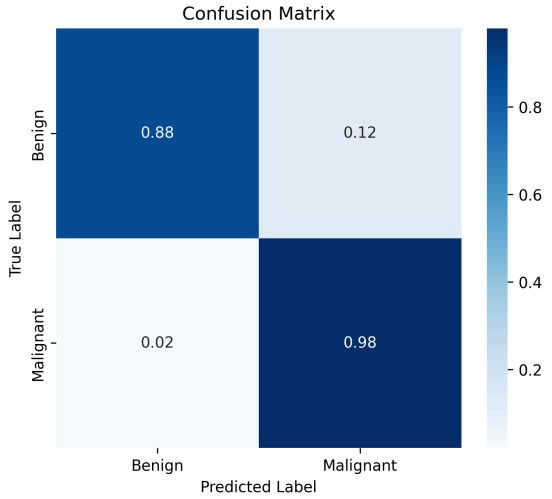

Figure 14. The confusion matrix of the 200X_b32_n500 model

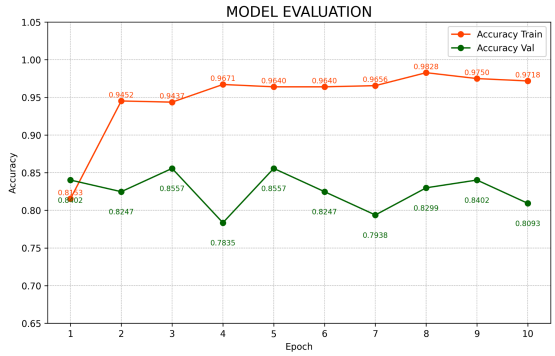

Figure 15. Model evaluation graph of the best performing model in the 400X magnification group

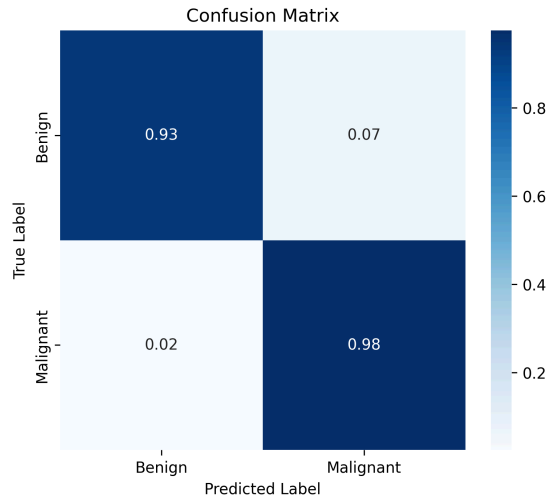

Figure 16. The confusion matrix for the best performing configuration of the 400X magnification model

3.4. 400X Magnification

The best performing model configuration for this magnification was a batch size of 32 and 500 images per class. As shown in the confusion matrix, the model correctly classified 98% of malignant and 93% of benign cases, indicating a well balanced classification ability (Figure 16). The model evaluation plot shows consistent training accuracy and some fluctuation in validation accuracy. Overall, this configuration delivered a relatively good performance at high magnification.

3.5. Conclusion

In this project we developed a breast cancer histopathological image classification pipeline using a ResNet-50-based deep learning model, trained across multiple magnification levels. By using transfer learning, data augmentation techniques and patient-wise splitting, we ensured both performance and generalizability. Among various configurations, we found that the 100X magnification model with batch size of 64

and 600 images per class achieved the most balanced performance. Additionally, the integration of Grad-CAM provided valuable insights into the regions that influenced predictions. These results show the potential of deep learning to support clinical diagnosis.

References

1. Xiong X, Zheng LW, Ding Y, et al. Breast cancer: pathogenesis and treatments. *Signal Transduct Target Ther*. 2025;10:49.

2. Kim J, Harper A, McCormack V, et al. Global patterns and trends in breast cancer incidence and mortality across 185 countries. *Nat Med*. 2025;31:1154–1162.

3. Jassam IF, Mukhlif AA, Nafea AA, Tharthar MA, Khudhair AI. A review of breast cancer histological image classification: Challenges and limitations. *Iraqi J Comput Sci Math*. 2025;6(1):Article 1.

4. Spanhol FA, Oliveira LS, Petitjean C, Heutte L. A dataset for breast cancer histopathological image classification. *IEEE Trans Biomed Eng*. 2016;63(7):1455–1462.

5. Behar N, Shrivastava M. ResNet50-based effective model for breast cancer classification using histopathology images. *Comput Model Eng Sci*. 2022;132(3):819–832.

6. Hicks SA, Strümke I, Thambawita V, et al. On evaluation metrics for medical applications of artificial intelligence. *Sci Rep*. 2022;12:5979.

7. Hinojosa Lee MC, Braet J, Springael J. Performance metrics for multilabel emotion classification: Comparing micro, macro, and weighted F1-scores. *Appl Sci*. 2024;14(21):9863.

8. Wang S, Zhang Y. Grad-CAM: Understanding AI models. *Comput Mater Continua*. 2023;77(3):3509–3513.

9. Chavengsaksongkram T, Watson L, Palomino-Garibay A. Breast cancer classification using convolutional neural network. University of Edinburgh, Machine Learning Project 4. 2020.