Breast Cancer Histopathological Image Classification

Kirthana Shri Chandra Sekar
chandrasekar.k@northeastern.edu

Samarth Saxena
saxena.sam@northeastern.edu

Vy Nguyen
nguyen.vy7@northeastern.edu

***Abstract* — Early and accurate diagnosis of breast cancer is crucial for improving treatment outcomes and survival rates. In this paper, we utilized advanced machine learning techniques to enhance the accuracy of breast cancer diagnosis. Our research involved analyzing 7,909 histopathological images of breast tumor tissue and categorizing them as either benign or malignant. We implemented various Neural Network architectures, including Multi-Layer Perceptron (MLP), Convolutional Neural Networks (CNN), and Residual Networks (ResNet), to perform this classification. To optimize performance, we conducted extensive fine-tuning by adjusting key parameters such as the optimizer, batch norm, and learning rate along with other hyperparameters. Our analysis critically evaluates the strengths and weaknesses of methods for breast cancer diagnosis using histopathological images, guiding future advancements in breast cancer diagnosis using histopathological images.**

I. Introduction

A. Overview

Breast cancer remains a global health issue, necessitating innovative approaches to enhance early detection and diagnosis. Histopathological examination of tissue samples is a crucial diagnostic technique, providing valuable insights into the nature of tumors. Our research aims to use deep neural networks, specifically Multi-Layer Perceptron (MLP), Convolutional Neural Networks (CNN), and Residual Networks (ResNet), to accurately classify breast cancer histopathological images. The dataset we used included both benign (non-cancerous) and malignant (cancerous) cases captured at varying magnification levels (40x, 100x, 200x, and 400x), which reflects the diversity encountered in real-life clinical settings.

B. Motivation

Our research aims to improve the accuracy and speed of breast cancer diagnosis. Current diagnostic methods sometimes struggle to differentiate between benign and malignant cases, which can cause delays in treatment. To address this issue, we analyze high-resolution breast tissue images and develop machine learning models that can reliably detect cancerous cells. Our goal is to significantly enhance diagnostic precision and expedite the decision-making process in breast cancer treatment.

C. Approach

Our main goal is to compare the performances of various architectures within three different models: MLP, CNN, and ResNet, to determine the most robust and performant architecture. We use the BreakHis dataset, which includes breast cancer histopathological images at various magnification levels, for our experiments. We train the models, which we re-implement from scratch, on the Training set and fine-tune them with diverse hyperparameter values using the Validation set. Finally, we select the best models based on their average accuracy across four magnifications on the Validation set and report their final average accuracy over these magnifications on the Test set. Fig. 1 demonstrates the pipeline of our methodology.

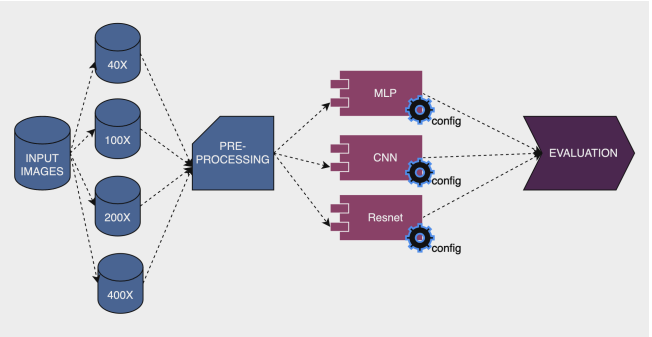


Fig. 1. *Methodology of classification process*

II. Related work

Several notable contributions have been made in recent efforts to enhance breast cancer diagnosis through histopathological image analysis. The BreakHis dataset serves as a benchmark for automated classification tasks, showcasing the challenges of the automated differentiation of benign and malignant images [2]. The authors used Support Vector Machines (SVM) and Random Forests (RF) for their classification.

Other papers utilized deep learning methods, such as a VGG16 network initialized with ImageNet weights, to achieve state-of-the-art performance in classification [3]. The incorporation of attention-based multiple instance learning (A-MIL) proved beneficial, enhancing classification accuracy and emphasizing the potential of deep learning frameworks in refining breast cancer diagnosis. Content-Based Medical Image Retrieval (CBMIR) further emphasized the utilization of a CNN-based Autoencoder method for improved feature extraction and higher performance in image retrieval [4].

To address the scarcity of labeled data, a self-supervised pre-training method - Magnification Prior Contrastive Similarity (MPCS) - demonstrated efficient representation learning without labels in histopathology. This method challenged the convention of relying heavily on human annotations [5]. The robustness of various deep learning architectures like ResNet [7] and MobileNet [8] was evaluated for breast cancer histopathological image analysis, and the WaveMix model stood out for its ability to provide stable and accurate classification results across the various magnification scales [6].

All in all, these collective contributions highlight the constantly developing landscape of breast cancer diagnostics by integrating advanced machine learning

techniques and datasets to improve the accuracy and interpretability of histopathological image analysis.

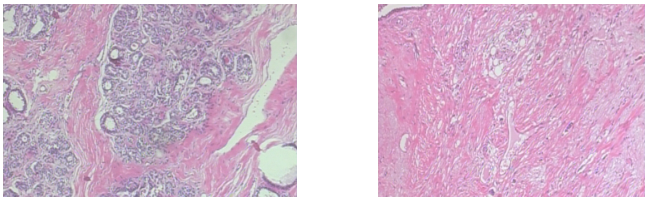
III. Dataset and Methodology

A. Dataset

The Breast Cancer Histopathological Database (BreakHis) [1] is a publicly available dataset that we used in this paper. The dataset contains 7,909 microscopic images of breast tumor tissue obtained from 82 patients. These images are divided into two major classes: benign and malignant, and captured at four magnification levels: 40X, 100X, 200X, and 400X, as shown in Figure 2. Each image in the BreakHis dataset is of size 700 x 460. We split the dataset into three parts: 60% for the training set, 20% for the validation set, and 20% for the test set. Note that the proportion of each magnification is maintained in each set. See Table 1 for the statistical number of the dataset.

Table 1. *Image distribution by magnification factor and class*

Magnification	Benign	Malignant	Total
40X	625	1,370	1,995
100X	644	1,437	2,081
200X	623	1,390	2,013
400X	588	1,232	1,820
Total Images	2,480	5,429	**7,909**



(a) Benign (b) Malignant

Fig. 2. *Representative images from BreakHis dataset*

B. Data Preprocessing

We first load and preprocess image datasets of different magnifications. This process includes converting images into the RGB format, reshaping them into uniform tensor sizes (e.g., 224x224 pixels), and standardizing pixel values. Specifically, we normalize these pixel values by centering them around zero and applying a standard deviation of one. This preprocessing step is crucial for enhancing the model's adaptability to variations in lighting and color. Consequently, it boosts the efficiency of the training process, fosters more effective convergence during the training phase, and improves the model's ability to generalize effectively when exposed to novel test data.

C. MLP Models

During the development of the Multi-Layer Perceptron (MLP) model, we experiment with different architectures by adjusting the number of layers. We test models with three and five layers and fine-tune the hyperparameters by trying out different learning rates, such as 0.001, 0.0001, and 0.00001. We also test the models with regularization values of 0.0001 and 0.00005, as well as without any regularization. To further improve the model's performance, we use different optimizer functions, namely Adam and Stochastic Gradient Descent (SGD), to investigate their impact.

Our goal is to find the optimal combination of MLP configurations that maximizes accuracy and generalization on the dataset. This involves an exploration of architectural dimensions and hyperparameter spaces.

D. CNN Models

During the configuration of the Convolutional Neural Network (CNN) model, we focus on maximizing feature extraction and classification accuracy by designing the architecture. The CNN architecture includes convolutional layers for hierarchical feature learning, pooling layers (using max or average pooling) for dimensionality reduction, and fully connected layers for final classification. We use a kernel size of 3 for all CNN models. However, we adjust the number of convolutional layers and the number of kernels in each layer to evaluate their impact on the model's performance. We apply the ReLU activation function across all models. In addition, we evaluate models with and without pooling layers. To further test the performance of these models, we employ different learning rates (0.001 and 0.0001) and optimizer functions (Adam, SGD). This aims to find an optimal setup for the CNN model in the image classification task.

E. ResNet Models

To explore more advanced CNN-based architectures, we conducted experiments on the ResNet family. Our model is denoted as ResNet-N, where N represents the depth of the network. Generally, ResNet comprises one convolutional stem, followed by four groups, and finally a fully connected layer. Each group consists of several blocks and one optional shortcut. For this study, we implement each block with two conv3x3 layers. We experiment with three models: ResNet-18, ResNet-26, and ResNet-34. One interesting question we seek to answer is whether models with deeper layers would yield better results. To investigate the impact of various learning rates and optimizer functions on the overall performance of the models, we apply them to the ResNet models, following the same approach we use for MLP and CNN architectures. Additionally, we conduct an ablation study to compare the performance of models with and without BatchNorm layers.

IV. Experimentation and Results

A. Experimental Setup

We implement all of our models from scratch using PyTorch. We train each model on the training set for 30 epochs with Cross-Entropy loss. We report the performance of the top-1 accuracy on the test set for all four magnifications. To determine the best model, we compute

the average accuracy of the four accuracies. Additionally, we report the AUC-ROC (Area Under the Receiver Operating Characteristic) curve to provide a more comprehensive understanding of performance, considering both true positive and false positive rates. To avoid overfitting, we employ early stopping by evaluating the accuracy of the validation set. Our goal is to create models that are not only highly accurate but also demonstrate robust and consistent performance across diverse configurations and magnification datasets, ensuring their reliability and effectiveness in real-world scenarios.

To monitor the learning progress and identify the best configurations, we use logging to record essential metrics such as train accuracy, loss, validation accuracy, loss, and test accuracy. This approach helps us effectively address overfitting and identify the configurations that yield the highest accuracy.

B. Results

MLP models. In our initial experiments, we trained various MLP network architectures. The results for different MLP networks are presented in Figure 3, where each line in the plot represents a distinct combination of hyperparameters. Generally, networks configured with 5 hidden layers perform better than those with 3 hidden layers. This improvement can be attributed to the deeper networks' enhanced ability to learn and represent more complex patterns within the data.

Moreover, when comparing optimization algorithms, networks using the Adam optimizer outperform those using the SGD optimizer. The Adam optimizer integrates aspects of both the AdaGrad and RMSProp algorithms, adapting learning rates for each parameter. This adaptive approach facilitates faster convergence and more effective navigation through steeper gradient landscapes, resulting in more efficient learning processes. It is important to note that using an appropriate learning rate is crucial for achieving the best performance. Using a learning rate that is too high (e.g., 0.001) can be detrimental to the model's performance.

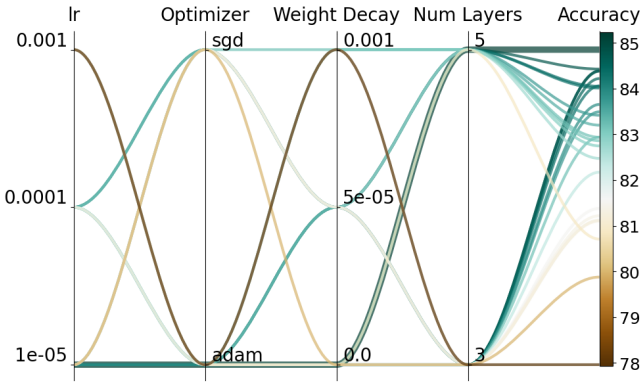


Fig. 3. *Performance of different MLP models. Each line in the plot represents a different combination of the hyper-parameters. We can trace the most effective configuration by tracking the lines that reach the highest point on the accuracy axis (rightmost column).*

CNN models. After conducting experiments with MLP models, we shifted our focus to training CNNs to further boost the performance. CNNs are another type of neural network known for its effectiveness in processing grid-like data, such as images. In our experiments, we adjust key parameters, including the number of convolutional layers, the choice of optimizer, pooling methods, and learning rate.

The outcomes of these varying CNN configurations are detailed in Figure 4. Although MLP models have a much larger number of parameters (~12 million) compared to CNN models (~ 6 million), the overall performance of CNN networks is significantly better than that of MLP models. The convolutional layers in CNNs efficiently capture local features in spatial data, and their pooling layers reduce the number of parameters while preserving essential information. This design enhances the CNNs' ability to process high-dimensional data, such as images, and improves generalization, thereby reducing overfitting. Empirical evidence from image classification tasks confirms the superior accuracy and performance of CNNs, as demonstrated in our experiments.

From Figure 4, it can be observed that CNN models utilizing the Adam optimizer perform better than those using SGD, similar to MLP models. Moreover, deeper networks tend to yield superior results compared to shallower ones. Additionally, CNN models that incorporate pooling layers significantly outperform those without. This improvement can be attributed to the ability of pooling layers to reduce dimensionality while preserving essential features, resulting in more robust learning and less overfitting. As a result, the model becomes more efficient in learning and can handle small distortions better.

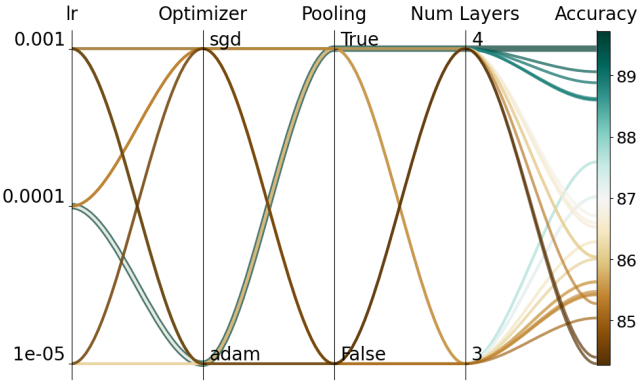


Fig. 4. *Performance of different CNN models*

ResNet Models. Recognizing the advantages of CNN models over basic MLP models, our research has delved into exploring more sophisticated CNN models, particularly within the ResNet family. We focused on three ResNet architectures: ResNet-18, ResNet-26, and ResNet-34. In our experiments, we adjust several critical parameters, such as learning rate, optimizer type, and the use of Batch Normalization (BatchNorm), to determine the optimal model configuration.

Our findings, presented in Figure 5, highlight the accuracy of various ResNet models equipped with BatchNorm. Notably, deeper models utilizing the Adam optimizer exhibited superior performance. Furthermore, Figure 6 analyzes ResNet models using the Adam optimizer, both with and without BatchNorm. Models incorporating BatchNorm consistently outperformed those lacking this feature, underscoring the effectiveness of BatchNorm in enhancing model performance.

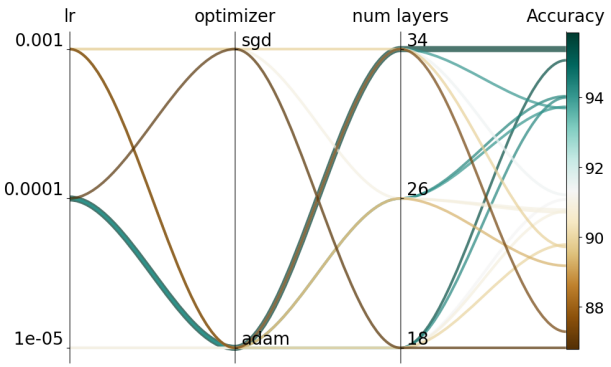


***Fig. 5**. Performance of different ResNet models with BatchNorm*

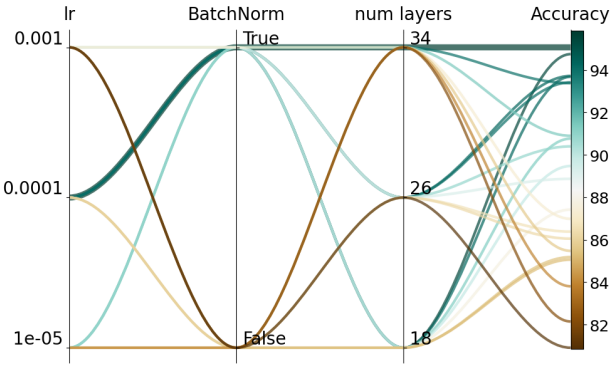


***Fig. 6.** Performance of different ResNet models with Adam optimizer*

Fig. 7 shows the accuracy over epochs of ResNet34 on training and validation sets. We can observe that on the validation set, Magnification 400X is the most challenging set. Additionally, while training accuracy keeps increasing, the accuracy on the validation set starts to drop after 26 epochs.

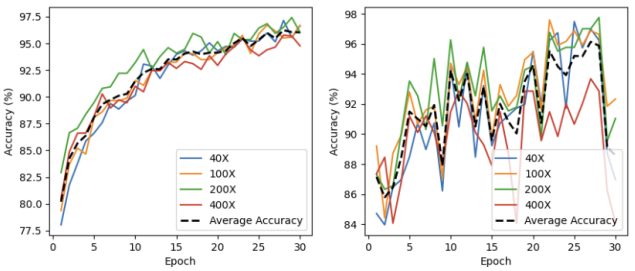


***Fig. 7.** Accuracy over epochs for ResNet-34. Left: Training set, Right: Validation set.*

Fig. 8 illustrates ROC by magnification of ResNet-34 on the dataset. In general, the model performs very well on all magnifications, in particular, 200X, where the AUC=1.

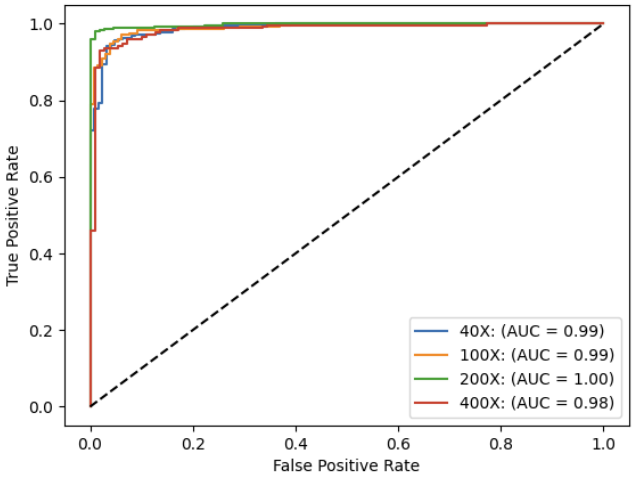


***Fig. 8.** ROC of ResNet-34 by magnification.*

Table 2 represents the confusion matrix of ResNet-34 by magnification. It can also be seen that 200X has the best performance among the 4 magnifications.

***Table 2**. Confusion matrix of ResNet-34 by magnification.*

	40X		**100X**		**200X**		**400X**	
	B	**M**	**B**	**M**	**B**	**M**	**B**	**M**
B	114	11	121	9	108	4	98	12
M	8	266	8	278	4	287	9	245

Summary results. Table 3 provides a summarized comparison of the best-performing models for each type: MLP, basic CNN, and ResNet. The results show that ResNet models deliver the highest performance, followed by basic CNNs and then MLPs. In general, models with deeper layers consistently outperform shallower ones.

Table 3. *Summary of accuracy of best-performing models for MLP, basic CNN, and ResNet. The best performance results are highlighted in bold.*

Model	**Magnification factor**				**Average accuracy over all magnification**
	40X	**100X**	**200X**	**400X**	
MLP-3 layers	83.46	81.01	89.58	84.89	84.74
MLP-5 layers	83.71	80.53	89.58	87.09	85.23
CNN-3 layers	86.97	86.3	92.06	85.71	87.76
CNN-4 layers	90.48	88.22	92.8	87.36	89.72
ResNet-18	**95.74**	94.47	96.77	**95.05**	95.51
ResNet-26	92.23	94.23	97.27	93.96	94.42
ResNet-34	95.24	**95.91**	**98.01**	94.23	**95.85**

V. Conclusion

In this paper, we explore the effectiveness of various neural network architectures in solving the classification problem related to Breast Cancer. We implemented various models and conducted extensive experiments with different configurations and found that models with more layers tend to perform better. Additionally, using BatchNorm and Adam optimization techniques helped us to achieve better results. We observed that simple CNN models outperformed MLP models, even with fewer parameters. Furthermore, using more sophisticated models such as ResNet significantly improved the performance. Exploring more advanced models such as Vision Transformer (ViT) is an interesting direction that we would like to investigate in future work.

References

[1] https://web.inf.ufpr.br/vri/databases/breast-cancer-histopathological-database-breakhis/

[2] F. A. Spanhol, L. S. Oliveira, C. Petitjean and L. Heutte, "A Dataset for Breast Cancer Histopathological Image Classification," in IEEE Transactions on Biomedical Engineering, vol. 63, no. 7, pp. 1455-1462, July 2016, doi: 10.1109/TBME.2015.2496264.

[3] A. Patil, D. Tamboli, S. Meena, D. Anand and A. Sethi, "Breast Cancer Histopathology Image Classification and Localization using Multiple Instance Learning," 2019 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE), Bangalore, India, 2019, pp. 1-4, doi: 10.1109/WIECON-ECE48653.2019.9019916.

[4] A. E. Minarno, K. M. Ghufron, T. S. Sabrila, L. Husniah and F. D. S. Sumadi, "CNN Based Autoencoder Application in Breast Cancer Image Retrieval," 2021 International Seminar on Intelligent Technology and Its Applications (ISITIA), Surabaya, Indonesia, 2021, pp. 29-34, doi: 10.1109/ISITIA52817.2021.9502205.

[5] P. Chhipa, et al., "Magnification Prior: A Self-Supervised Method for Learning Representations on Breast Cancer Histopathological Images," in 2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 2023 pp. 2716-2726. doi: 10.1109/WACV56688.2023.00274

[6] P., Pranav Jeevan & Kurian, Nikhil & Sethi, Amit. (2023). Magnification Invariant Medical Image Analysis: A Comparison of Convolutional Networks, Vision Transformers, and Token Mixers.

[7] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[8] Howard, Andrew & Zhu, Menglong & Chen, Bo & Kalenichenko, Dmitry & Wang, Weijun & Weyand, Tobias & Andreetto, Marco & Adam, Hartwig. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.