POLITECNICO
MILANO 1863

# AN2DL - Second Challenge Report
# The Termin-AI-tor: the return

Cristiano Battistini, Giulia Di Virgilio, Georgia Manioudaki, Francesco Giuseppe Tagliabue
cristianobattistini, giuliadivii, giorgiamanioudaki, francescotagliabue
244466, 271225, 271253, 271220

December 16, 2025

## 1 Introduction

This project adresses a multi-class image classification task using deep learning methods. The goal is to correctly classify images into four different breast cancer histology classes (**Luminal A**, **Luminal B**, **HER2(+)** or **Triple negative**) based on patterns in the tissue morphology. To tackle this problem, we adopted a stepwise approach, by starting with a vanilla CNN architecture and then increasing complexity, additionally implementing pre-trained models. The F1 score was the principal evaluation metric, as required by the project rules.

## 2 Problem Analysis

The training dataset consists of 691 RGB images of breast core tissue, categorized into four molecular subtypes of breast cancer[3]: **Luminal A**, **Luminal B**, **HER2-positive**, and **Triple negative**. Each image is also paired with a binary mask highlighting regions relevant for class identification. A major challenge of the project arose during pre-processing, as the dataset contained images "affected" by green slime artifacts and marker signs used to annotate relevant tissue regions. Overall, the training dataset included a non-negligible number of corrupted samples, namely *slime images* ($n = 50$) and *Shrek images* ($n = 60$). Finally, the dataset exhibits a mild class imbalance: the **Triple negative** class is underrepresented, while **Luminal B** is the most prevalent (Figure 1).

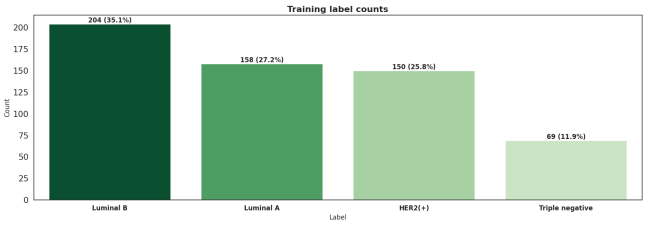


**Figure 1:** *Class imbalance in the cleansed training set*

## 3 Methods

### 3.1 Out-of-domain samples removal

To mitigate domain shift, corrupted and out-of-domain samples were removed. Since "Shrek-corrupted" images reused the *same mask* as genuine tissue samples, duplicate masks were identified using rotation- and flip-invariant perceptual hashing, enabling the removal of the associated non-tissue overlays. Slime artifacts were detected using a five-color signature: an image was flagged and discarded only if all sampled colors were present. After filtering, the dataset was split into training and validation sets using an 80 : 20 ratio.

### 3.2 Image Preprocessing

Image preprocessing was a critical initial step of the project, as understanding the nature of the images provided was essential for effective model training. In particular, background removal was performed to eliminate noise that did not contribute to the learning process of the models. To perform this step, we employed the masks that were provided along with

the training images, and used them to successfully crop the areas of interest.

### 3.3 Patches Cropping

To allow the model to extract meaningful features, patches were generated from each original image. Squares of 156×156 pixels were cropped around coordinates close to the image mask, and only the top 20% of patches, ranked by their mask coverage, were retained in the final dataset. A minimum distance between patch centers was enforced to reduce excessive overlap.

### 3.4 Data augmentation

To enhance generalization, we applied *on-the-fly* augmentation exclusively to the training set. Each sample first underwent random horizontal/vertical flips, followed by a small number of geometric transformations such as mild rotations, translations, and shear. Additionally, we applied *stain-safe photometric* adjustments to brightness and contrast, avoiding unrealistic color shifts. When a mask was present, it was transformed only using the same geometric operations as the image. Validation and test set images were left unaugmented.

### 3.5 Normalization

Normalization is crucial when fine-tuning backbones pretrained on *ImageNet* samples: RGB channels of training, validation and test images have been standardized using the reference ImageNet mean and dispersion, matching the input distribution expected by the pre-initialized weights.

### 3.6 Mask-conditioned synthetic augmentation (diffusion)

Due to the limited training data, we initially explored a mask-conditioned synthetic augmentation strategy using diffusion inpainting. Tissue masks guided a generative model to modify only relevant regions while preserving the background, with the resulting images intended for inclusion in the training set. Although promising, this approach was ultimately excluded from the final pipeline due to distribution shift concerns and time constraints, but it remains a potential avenue for future improvements.

## 4 Models

### 4.1 Baseline models: vanilla CNN

We first implemented a lightweight Convolutional Neural Network (CNN). Convolutional filters with shared weights learn visual features such as edges and textures with fewer parameters than dense layers, while pooling provides translation invariance that aids generalization[2]. The model consisted of four convolutional blocks with $3 \times 3$ kernels and increasing channels (32→64→128→256), each followed by ReLU and 2×2 max-pooling. The final feature map was flattened, regularized with dropout, and mapped to 4 output logits via a single fully connected layer.

### 4.2 Pre-trained models and Transfer Learning

Since the baseline CNN showed limited generalization, we evaluated *ImageNet*-pretrained CNN backbones adapted to our histopathology multi-classification task. Transfer learning leverages features learned on large-scale datasets while specializing the final representations to our domain. Specifically, we replaced the original classifier with a 4-class head and applied **partial fine-tuning**, keeping early backbone layers frozen and training the new head along with the last backbone stage(s). We tested three backbone families within the same pipeline: **ResNet50**, with residual bottleneck blocks for efficient deep representations; **VGG16/19**, based on deep stacks of $3 \times 3$ convolutions and max-pooling; and **InceptionV3**, which uses multi-scale convolutional branches to capture patterns at different spatial resolutions. Within the pre-trained models, the VGG models were revealed to be the best-performing ones.

### 4.3 Ensemble

Inspired by heterogeneous CNN ensembles for breast cancer diagnosis[4], we combined multiple backbones using *majority voting* over predicted classes. In case of ties, we broke them in favor of the VGG models (prioritizing VGG19, and VGG16), since they performed better. This improved robustness and achieved our best test performance (F1 = 0.4697). The ensemble composition is reported in Appendix table 3.

**Table 1:** *Validation performance using the training setup (loss and hyperparameters) reported in Table 2. Best results are in* ***bold****.*

| Model | Accuracy | Precision | F1 score |
|---|---|---|---|
| Baseline CNN | 0.3124 | 0.3551 | 0.3231 |
| ResNet50 | 0.4103 | 0.4235 | 0.4088 |
| InceptionV3 | 0.3316 | 0.3346 | 0.3318 |
| VGG16 | 0.3844 | 0.3822 | 0.3749 |
| VGG19 | **0.4161** | **0.4360** | **0.4168** |

# 5 Loss Function

To address the class imbalance present in our dataset, we initially adopted the soft cross-entropy loss function, which helps mitigate model overconfidence. Subsequently, we employed a more stringent approach to handle class imbalance by utilizing the **Focal Loss** function.

# 6 Inference

At test time, we applied the same preprocessing as during training. For each cropped patch, the model outputs logits $z_i \in R^C$, converted to probabilities $p_i$ via softmax. Patch-level probabilities from the same image were aggregated by computing their mean. To further stabilize predictions, test-time augmentation (TTA) was applied by evaluating each patch with horizontal and vertical flips and averaging the logits before softmax.

# 7 Experiments

The experimental process involved evaluating each model's performance over the validation set. We compared the baseline CNN with the performance of the pretrained models and, at last, the ensemble. The results of the models employed can be seen in Table 1.

# 8 Results

The performance of the different pretrained models was broadly comparable, with no single architecture demonstrating a clear advantage over the others on its own. However, combining these models into an ensemble resulted in a substantial improvement over the baseline CNNs and individual pretrained networks. By aggregating the predictions of multiple architectures, the ensemble was able to capture complementary features and reduce the impact of individual model weaknesses, leading to more robust and consistent classification performance across all classes.

# 9 Discussion

In conclusion, we found that focusing on a single model proved less effective. While a standalone pretrained model can perform well under specific conditions, its generalization is limited, especially given the low resolution and small size of our dataset. An ensemble of pretrained models, however, leveraged complementary strengths across architectures, improving robustness, stability, and overall performance. For future work, our current ensemble relies on majority voting, which ignores prediction confidence. We plan to implement a probability-level ensemble, averaging softmax outputs from VGG16, VGG19, ResNet50, and InceptionV3. This approach has been shown to enhance stability and accuracy, preserving class-specific recall while improving overall performance[1].

# 10 Conclusions

Under limited data, *ImageNet*-pretrained backbones consistently outperformed the vanilla CNN, and a majority-vote ensemble further improved robustness, achieving our best score. While performance remains constrained by data scarcity and potential distribution shift, these results highlight the impact of transfer learning and simple ensembling in this setting.

## References


[1] Z. Hameed, S. Zahia, B. Garcia-Zapirain, J. Javier Aguirre, and A. Maria Vanegas. Breast cancer histopathology image classification using an ensemble of deep learning models. *Sensors*, 20(16):4373, 2020.

[2] K. O'Shea and R. Nash. An introduction to convolutional neural networks, 2015.

[3] S. Park, J. S. Koo, M. S. Kim, H. S. Park, J. S. Lee, J. S. Lee, S. I. Kim, and B. W. Park. Characteristics and outcomes according to molecular subtypes of breast cancer as classified by a panel of four biomarkers using immunohistochemistry. *Breast*, 21(1):50–57, 2012.

[4] H. Zerouaoui, O. E. Alaoui, and A. Idri. New design strategies of deep heterogenous convolutional neural networks ensembles for breast cancer diagnosis. *Multimedia Tools and Applications*, 83(24):65189–65220, 2024.

# A Appendix

## A.1 Hyperparameters

**Table 2:** *Training hyperparameters on the final model.*

| Hyperparameter | Value |
|---|---|
| Seed | 1 |
| Learning rate | 1e-4 |
| Batch size | 64 |
| Epochs | 500 |
| Patience | 50 |
| Dropout rate | 0.4 |
| Optimizer | Ranger |
| Weight decay | 5e-5 |

## A.2 Ensemble

**Table 3:** *Models included in the final ensemble and aggregation rule.*

| Component | Details |
|---|---|
| Voters | VGG19 (Ranger), VGG16 (Ranger), VGG16 (Adam), VGG16 (Adam, +synthetic), ResNet50 (Ranger), InceptionV3 (Ranger) |
| Checkpoints | Best validation $F_1$ checkpoint selected independently for each voter |
| Aggregation | Majority voting on predicted class; ties resolved in favor of the VGG family (VGG19 first, then VGG16 variants) |