POLITECNICO
MILANO 1863

AN2DL – Second Challenge Report
The Gradient Descenters

Alberto Occhipinti, Giulia Putelli, Matteo Morini, Tommaso Rossetti

albertoocchipinti, giuliaputelli, ilraccoglitore, tommasorossetti

301511, 301959, 308227, 301319

December 16, 2025

1 Introduction

This project addresses a **multi-class image classification** task where each training sample is composed of an RGB image and an associated binary mask highlighting regions of interest. The dataset consists of **691 training images and 691 corresponding masks**, each with different spatial resolutions.

The main objective is to correctly classify each image into one of **four classes**, leveraging both image content and structural information provided by the masks. Throughout the project, we explored different architectural choices, data cleaning strategies, and training paradigms, progressively improving performance from a simple CNN baseline to a transfer learning solution based on ConvNeXt.

2 Problem Analysis

An initial inspection of the dataset revealed multiple critical issues:

Data quality problems. The dataset contained duplicated samples and non-informative images, including **60 "Shrek" images** and **50 duplicated images containing mucosa-like artifacts**. After cleaning, the final training set was reduced to **581 valid images**.

Class imbalance. The dataset exhibited a clear imbalance among the four classes, as shown in Figure 1. This imbalance motivated the use of class-weighted loss functions to penalize misclassification of underrepresented classes.

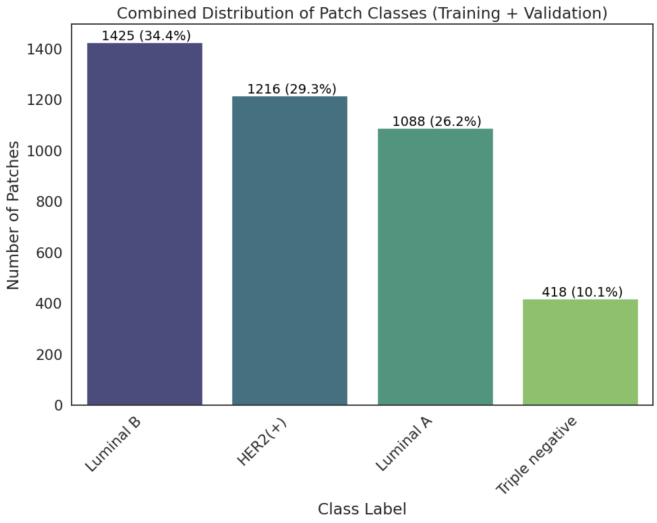

Figure 1: Distribution of patches among the 4 classes in the cleaned dataset.

Structural heterogeneity. Each mask often contained multiple disconnected regions of interest, suggesting that treating each image as a single entity might not be optimal.

3 Method

3.1 Baseline CNN

We initially implemented a custom CNN trained on full images without exploiting mask information. The architecture consisted of **four convolutional blocks**, each composed of:

- Convolutional layer
- ReLU activation

- MaxPooling (excluded in the last block)

The convolutional backbone was followed by a flatten layer, dropout regularization, and a fully connected head with softmax activation for four-class prediction. This baseline achieved an F1 score of approximately **25%**.

3.2 Data Cleaning and Augmentation

We introduced standard geometric augmentations (random flips, rotations, translations, and scaling), yielding marginal improvements.

A more substantial gain came from dataset cleaning. Duplicate samples were detected by applying an MD5 hash function **exclusively to the masks**. This approach revealed duplicated masks corresponding to "Shrek" images; the associated RGB images were then removed accordingly.

Images containing mucosa artifacts were not detected via hashing, as they were present in both "clean" and "contaminated" versions. These samples were therefore removed manually. After cleaning, the training set contained **581 images**, and performance increased to approximately **29% F1**.

3.3 Mask Exploitation and Patch Extraction

We explored multiple strategies to exploit the provided masks:

- Bitwise masking of RGB images
- Using the mask as an additional input channel

Both approaches yielded limited or negative results. Grad-CAM visualizations revealed that the model frequently focused on background regions outside the tissue region (as shown in Figure 2).

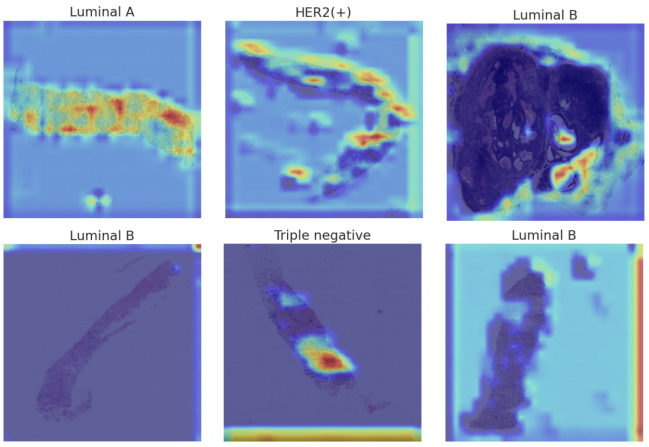

Figure 2: Grad-cam visualization

To address this issue, we adopted a **patch-based strategy**. For each connected white region in the mask, we extracted a bounding box and applied it to the original image, producing multiple patches per image. Training was then performed on individual patches rather than full images.

This approach improved spatial focus and raised performance to approximately **32% F1**, representing the best result obtained with a custom CNN.

3.4 Transfer Learning

To further improve performance, we transitioned to transfer learning using pretrained models. We evaluated several architectures, including MobileNetV1, EfficientNetB0, EfficientNetV2-M, and ConvNeXt-Large.

While lightweight models yielded modest improvements, larger networks consistently suffered from overfitting—likely due to the limited dataset size—resulting in inferior performance. The best trade-off between capacity and generalization was achieved using **ConvNeXt-Tiny**, combined with the **Lion optimizer**, reaching **39% F1**. This configuration was selected as our final backbone.

3.5 Fine-Tuning and Patch Filtering

We initially tried standard fine-tuning approaches, such as freezing the backbone and training only the classifier head, followed by full unfreezing.

We therefore implemented **progressive unfreezing**, gradually unfreezing the ConvNeXT backbone:

- The classifier head was trained first
- Two backbone blocks were unfrozen every 20 epochs
- All blocks were progressively unfrozen

This approach improved stability and increased performance to **42.31% F1**.

Additionally, we observed that very small patches carried limited semantic information and introduced noise. We therefore removed patches smaller than **200 pixels** from both training and inference pipelines. This final refinement yielded our best score of **42.84%**.

During inference, the same patch extraction strategy was applied to test images. Patch-level predictions were aggregated using softmax averaging to obtain image-level predictions.

4 Failed Experiments

Several techniques commonly used in computer vision did not improve performance:

- **Group Normalization:** did not lead to improvements since batch sizes were already stable and ConvNeXt relies on LayerNorm, making additional normalization unnecessary.
- **CutMix and MixUp:** degraded performance because mixing patches disrupted spatial consistency, producing samples that did not correspond to realistic regions of interest.
- **Test-Time Augmentation (TTA):** did not improve results as patch-level predictions were already noisy.
- **Dual-stream models:** combining full-image and patch-based branches increased model complexity, but full images often contained large irrelevant regions that did not provide complementary information.
- **Multi-scale patches:** introduced scale inconsistencies between patches, making it harder for the network to learn stable and transferable representations.
- **Contrastive learning:** was ineffective due to the limited dataset size and high intra-image variability, which made it difficult to form meaningful positive pairs.
- **Synthetic augmentation:** produced samples that were visually plausible but semantically inconsistent, introducing noise and reducing generalization.

5 Results

Our final model achieved a validation F1 score of **42.84%**, representing a **17-point improvement** over the baseline CNN.

Table 1: Evolution of model performance through major architectural changes. Best result in bold.

Model Configuration	**F1 Score (%)**
Baseline CNN + FCN	25.0
+ Data Cleaning	29.0
+ Patch Extraction	32.0
+ ConvNeXt-Tiny + Lion Optimizer	39.0
+ Progressive Unfreezing	42.31
+ Patch Filtering	**42.84**

Key achievements include:

- Effective patch-based classification driven by mask geometry
- Robust transfer learning with ConvNeXt-Tiny
- Stable fine-tuning via progressive unfreezing
- Significant gains from data-centric cleaning

6 Discussion

Strengths. The patch-based approach allowed the model to focus on meaningful regions while maintaining computational efficiency. Progressive unfreezing enabled effective adaptation of pretrained features without catastrophic forgetting. Data cleaning proved to be the most impactful factor.

Limitations. The relatively small dataset size limits the applicability of larger architectures. Patch filtering thresholds were manually tuned and may not generalize to different datasets. Performance plateaued around 43%, suggesting intrinsic dataset complexity.

Unexpected findings. Several commonly effective techniques did not improve performance. These results highlight the importance of domain- and data-specific design choices.

7 Conclusions

This work demonstrates that effective image classification pipelines benefit from **data-centric optimization** as much as from **architectural complexity**. It also demonstrates the effectiveness of combining pretrained models with fine-tuning, and highlights the importance of data augmentation in building strong and robust models.

Our final system achieves competitive performance through:

- Rigorous dataset cleaning
- Mask-guided patch extraction
- Transfer learning with progressive adaptation
- Careful filtering of noisy samples

Future work may explore ensemble strategies, adaptive patch weighting mechanisms, and semi-supervised learning to further leverage unlabeled data.