UCLouvain

epl

École polytechnique de Louvain

Machine Learning Approaches for Skin Cancer Detection:

A Study Based on the ISIC 2024 Competition

Author: **Wei ZHOU**
Supervisor: **Charles PECHEUR**
Readers: **Quentin CAPPART, Hélène VERHAEGHE**
Academic year 2025–2026
Master [120] in Computer Science

Disclaimer

This thesis has benefited from the assistance of generative AI tools, such as ChatGPT (1), which were used exclusively to improve the clarity, structure, and overall readability of the manuscript. Their use was mainly limited to tasks such as rephrasing sentences to improve readability, and translating passages from French into English. All analyses, experiments, and conclusions remain entirely my own work.

Abstract

Skin cancer detection from clinical images remains challenging due to the high visual variability of skin lesions. In the context of the ISIC 2024 challenge, this thesis examines the relative impact of architectural modifications and pretraining strategies on model performance and behaviour. The results show that attention-based architectural changes provide only limited gains compared to the effect of pretrained initialization. In-domain self-supervised pretraining with BYOL achieves performance comparable to supervised ImageNet-1k initialization under restrictive computational conditions. In addition, the domain of the pretraining data influences how the model exploits visual information and focuses its attention.

Beyond image-only models, the thesis also investigates multimodal integration strategies. Multimodal experiments demonstrated that gradient boosting models combining metadata with CNN predictions consistently outperformed neural network fusion approaches. External validation on a dermoscopic dataset confirmed reasonable generalisation under strong domain shift, with the Triplet Attention variant outperforming the BYOL pretrained baseline and achieving performance comparable to the ImageNet pretrained model. Saliency map analysis further indicated that BYOL pretraining led to more lesion centred attention, which did not translate into better predictive performance.

Contents

1 Introduction

Skin cancer is among the most common cancers worldwide, and early identification of suspicious lesions can significantly improve outcomes. Computer vision and machine learning are increasingly used to support clinicians by analyzing images of cutaneous lesions. This thesis studies skin cancer detection using the ISIC 2024 challenge dataset, which includes both images and tabular descriptors capturing lesion characteristics.

The objective of this master's thesis is to provide an experimental analysis of modern deep learning based approaches for skin cancer detection. The study focuses on architectural design choices, pretraining strategies, and the integration of multimodal data. To achieve this goal, a complete inference pipeline is developed, starting from established models in the literature and adapting selected components to the studied problem. The final developed approach is evaluated on both the public and private sets of the ISIC 2024 challenge. In order to further assess robustness, additional experiments are conducted using an external dataset not included in the competition. The purpose of this work is not to achieve the highest possible competitive score, but rather to conduct an experimental study of machine learning approaches for skin cancer detection. The source code associated with this work is available on GitHub [1].

This thesis is organized as follows. First, the context and motivation of the study are introduced, followed by a review of relevant related work. A training and inference pipeline for image-based models is then presented. Subsequently, different strategies for integrating metadata with visual information are compared. An interpretability analysis based on Grad-CAM is performed to examine the image regions influencing model predictions. Finally, the generalization ability of the proposed approaches are evaluated using an external dataset.

[1] `https://github.com/Volgaar2410/TFE_2026-ISIC_skin_cancer_detection`

2 Context and Motivation

2.1 Skin Cancer

Skin cancer refers to a group of malignant diseases that originate from the epidermis. These cancers arise primarily because of cellular damage caused by long-term exposure to ultraviolet radiation. Such exposure can lead to mutations that interfere with normal cell growth and repair mechanisms. As a result, skin cancer has become one of the most frequently diagnosed cancers worldwide and remains the predominant cancer type among Caucasian populations. The continuing rise in cases observed in recent decades is mainly attributed to demographic aging and lifestyle factors that promote greater sun exposure. Among individuals over the age of 50, the incidence of melanoma alone is estimated to rise by approximately 0.6% per year (2).

Skin cancers are commonly divided into two broad categories. Non melanoma skin cancers include basal cell carcinoma and squamous cell carcinoma. Melanoma is less common, yet it accounts for the majority of deaths due to skin cancer. Representative visual examples of these three major skin cancer types are shown in Figure 1. In addition, their main visual characteristics are summarized in Table 1 to highlight the differences relevant for image based examination.

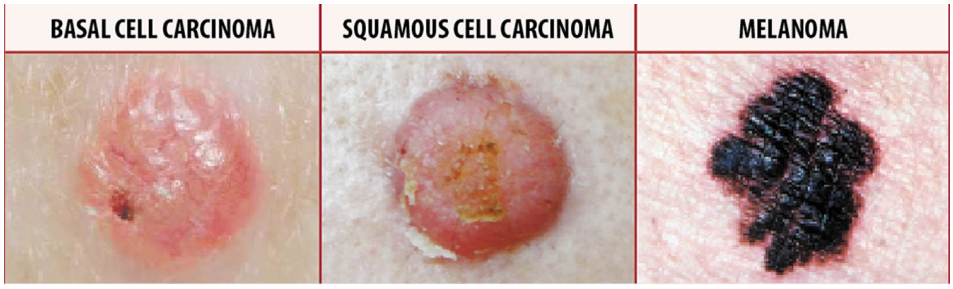


Figure 1: Visual examples of the three main types of skin cancer: Basal-Cell Carcinoma (BCC), Squamous-Cell Carcinoma (SCC), and Malignant Melanoma. Source: healthjade.net

Table 1: Types of skin cancer and their main visual characteristics. From (3)

Type of Skin Cancer	Visual Description
Basal-Cell Carcinoma (BCC)	Pearly or translucent bump, sometimes with small visible blood vessels or a central ulceration, and rarely metastasizing.
Squamous-Cell Carcinoma (SCC)	Red, scaly, or crusted lesion or bump that may ulcerate or bleed and can grow into a large mass if left untreated.
Malignant Melanoma	Irregularly shaped and unevenly pigmented lesion with variable colors and changing size or appearance.

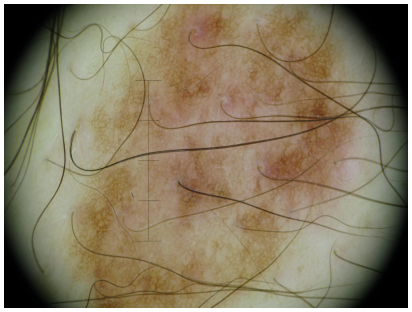
Note: A metastasis refers to the spread of cancer cells to distant organs or tissues.

The diversity of skin cancer appearances and the high mortality of melanoma make early and accurate detection essential. In clinical practice, diagnosis has relied on visual examination performed by dermatologists, supported by complementary tools such as dermoscopy, biopsy, and histopathological analysis. However, these approaches present several limitations, including diagnostic subjectivity, high costs, and the need for experienced medical professionals. These challenges have encouraged the development of computer-assisted and automated detection methods.

2.2 Dermoscopy

Dermoscopy is a non-invasive imaging technique that enhances the visualisation of morphological structures within the skin. It provides a detailed view of pigment distribution, which helps dermatologists detect melanoma and other skin lesions more accurately.

Dermoscopic imaging represents one of the main data sources used in automated skin lesion analysis. In parallel, non dermoscopic datasets, such as clinical photographs or smartphone acquired images, provide complementary perspectives under more diverse imaging conditions.



(a) Dermoscopic image (Derm12345 dataset)

(b) Non-dermoscopic image (ISIC Skin Cancer Detection 2024)

Figure 2: Examples of dermoscopic and non-dermoscopic skin lesion images.

2.3 Kaggle Competition ISIC 2024

Kaggle is an online platform that hosts machine learning competitions proposed by research institutions and industry partners. Each competition presents a defined predictive task along with an open dataset, allowing participants to design and evaluate algorithms. Beyond the competitive aspect, Kaggle offers a collaborative environment in which participants share code, methods, and discussions, supporting reproducibility in artificial intelligence research.

The ISIC 2024 Skin Cancer Detection with 3D-TBP competition, organized by the International Skin Imaging Collaboration and hosted on Kaggle, was held from June 2024 to September 2024. Its goal was to develop computer vision models capable of detecting malignant skin lesions from non-dermoscopic images captured through total body photography (3D-TBP). This edition marked a shift from previous ISIC challenges that primarily focused on dermoscopic imagery.

As with other Kaggle competitions, participants submitted predictions within a fixed time period, and performance was evaluated using a predefined metric. During the competition, results were displayed on a public leaderboard based on a subset of the test data. Once the competition concluded, the final rankings were determined using a separate, hidden portion of the test set.

This master's thesis uses the dataset released for the ISIC 2024 competition. Although this work does not constitute official participation, it relies on the available data and official evaluation metric to explore deep learning approaches for skin cancer detection.

3 Related Work

3.1 Deep Learning for Image-Based Skin Cancer Detection

Deep learning has become the leading approach in automated skin cancer detection. This development marks a shift from traditional feature-engineered methods to learning-based models capable of extracting representations directly from image data. This transition has more recently expanded with the introduction of transformer-based architectures such as Vision Transformers. The following subsections outline this evolution.

3.1.1 From Traditional Computer Vision to Deep Learning

Before the advent of deep learning, early skin lesion classification systems relied on handcrafted features such as color histograms, texture descriptors, and shape analysis combined with classical machine learning classifiers. A representative example of this paradigm is the work of Celebi *et al.* (4), who proposed an approach to dermoscopic image classification based entirely on handcrafted feature extraction.

In their system, the lesion area was first segmented before extracting geometric descriptors such as asymmetry, compactness, and aspect ratio, along with color and texture statistics. These handcrafted features were then used to train a support vector machine classifier for lesion classification. This workflow shows the limitations of pre-deep-learning pipelines, whose performance depended heavily on the quality of manual features and may reflect limited robustness to clinical variability. Celebi *et al.* noted practical issues, including segmentation failures and the exclusion of images with hair obstruction or poor contrast conditions.

Figure 3: Schematic of the lesion classification system proposed by Celebi *et al.*, adapted from (4).

3.1.2 CNN-Based Approaches and Transfer Learning

Following the handcrafted descriptors, convolutional neural networks became the dominant framework for automated skin lesion classification. This shift was supported by transfer learning from large-scale image datasets such as ImageNet, which helped to mitigate the limited availability of medical imaging data. A

study by Esteva *et al.* (5) demonstrated the clinical potential of CNN by achieving dermatologist-level performance in separating high-risk skin lesions from low-risk ones.

The study employed an Inception-v3 convolutional neural network pretrained on the ImageNet-1k dataset and subsequently fine-tuned on 129,450 dermatological images. The resulting model was evaluated against 21 certified dermatologists and achieves comparable diagnostic accuracy.

Comparative Evaluations of CNN Architectures In the context of the ISIC 2020 challenge, Cassidy *et al.* (6) conducted a comparative analysis of widely used CNN backbones. Their study compared a broad range of CNN architectures without pretraining on the ISIC 2020 dataset, providing insight into the impact of architectural choices on classification performance. While some high-capacity models such as VGG19 achieved the highest performance, they did so at the cost of substantially increased model complexity, which limits their practical applicability in resource-constrained, such as mobile applications. In contrast, architectures such as DenseNet121, ResNet50, and EfficientNetB0 offered a more effective balance between performance and computational cost.

Table 2: Performance comparison of baseline CNN models on the ISIC 2020 testing set without pre-training, data from (6).

Model	**Params**	**Best epoch**	**AUC**
DenseNet121	7,039,554	25	0.77
DenseNet169	12,646,210	47	0.66
EfficientNetB0	4,052,133	37	0.75
EfficientNetB1	6,577,801	30	0.76
EfficientNetB2	7,771,387	67	0.73
InceptionResNetV2	54,339,810	10	0.64
InceptionV3	21,806,882	47	0.50
ResNet50	23,591,810	27	0.71
ResNet50V2	23,568,898	41	0.73
ResNet101	42,662,274	21	0.70
ResNet101V2	42,630,658	37	0.76
VGG16	134,268,738	21	0.77
VGG19	139,578,434	30	**0.80**
Xception	20,865,578	16	0.75

Interpretability Challenges While CNNs have achieved remarkable performance in skin lesion classification, their use has raised concerns about the interpretability of deep learning models. The feature representations learned by CNNs make it difficult to understand how visual cues contribute to a prediction. This limitation is commonly referred to as the *black box* nature of deep models and has motivated the development of explainability techniques aimed at improving the transparency of CNN-based systems.

Among these methods, saliency-based approaches such as Gradient-weighted Class Activation Mapping (Grad-CAM) and its variants are widely used in skin cancer detection. They generate visual heatmaps that highlight the parts of the image that contribute most to the model's prediction. These visualisation tools provide valuable insight and have become a standard component in the evaluation of CNN-based models.

In dermatological applications, saliency maps can be used to reveal cases where CNNs focus on parts of the image that are unrelated to the lesion. Artefacts such as pen markings, measurement scales, or surrounding skin structures may draw the model's attention away from the lesion and affect its prediction. Figure 4 illustrates several examples of attention patterns influenced by such artefacts.

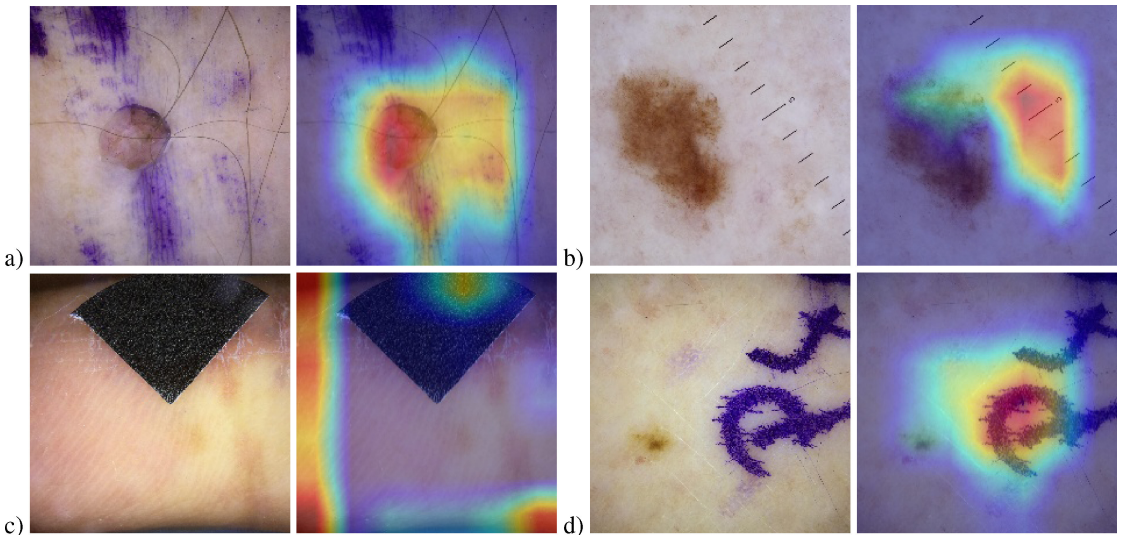


Figure 4: Grad-CAM visualisations showing CNN attention affected by common artefacts in clinical images, from (6). Examples include focus on pen markings (a, d), measurement overlays (b), and surrounding skin and wound dressing (c).

3.1.3 Advances in ViTs and Hybrid Models

Recent advances in deep learning have seen the introduction of transformer-based architectures. Originally developed for natural language processing, the transformer architecture has been adapted to vision tasks through self-attention mechanisms. Vision Transformers (ViT) (7) represent images as sequences of patches and use self-attention mechanisms to capture long-range spatial relationships within the image.

Role of Transfer Learning in Vision Transformers Vision Transformers rely heavily on large-scale pretraining, as their performance is strongly influenced by the size and quality of the data used during this stage. Without substantial pretraining, Vision Transformers tend to struggle in limited-data conditions.

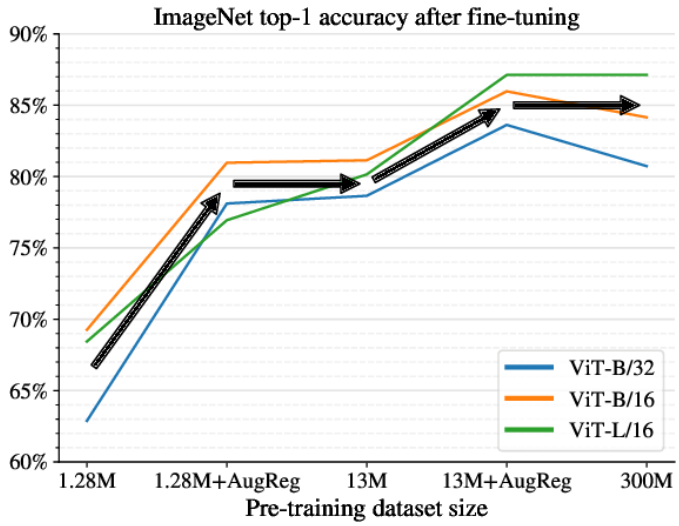


Figure 5: Fine-tuned ImageNet accuracy of Vision Transformer models under different pretraining dataset sizes and pretraining regimes. *AugReg* denotes augmentation and regularization. From (8)

This dependency is illustrated in Figure 5. The figure highlights that Vision Transformers benefit considerably from large-scale pretraining, with performance improving significantly as the size of the pretraining dataset increases. In some configurations, accuracy increases by nearly 20%.

Comparison Between Vision Transformers and CNNs The comparison reported in Table 3, from Dagnaw *et al.* (9), evaluates Vision Transformer and CNN models pretrained on ImageNet and fine-tuned on a subset of the ISIC dataset

available on Kaggle. The results show that the transformer-based models included in the study reach performance levels close to the best-performing convolutional networks. However, these transformer variants require considerably more parameters: models such as ViT-L and Swin-B are substantially larger, whereas lightweight architectures like MobileNetV2 offer a more balanced trade-off between accuracy and computational cost.

Table 3: Comparison of CNN- and Vision Transformer-based models in terms of classification performance and model complexity.

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**Params (M)**
ResNet18	82.4	80.5	81.0	11.6
ResNet50	88.8	86.9	88.6	25.5
DenseNet201	83.2	85.1	76.3	7.9
MobileNetV2	85.3	83.5	84.3	3.5
ViT-B	88.6	88.6	86.0	85.8
ViT-L	88.6	89.2	85.3	304.2
Swin-S	87.7	89.5	82.6	48.8
Swin-B	87.8	86.6	86.6	86.7
Swin-T	87.7	90.1	82.0	27.5

3.1.4 Data Augmentation and Preprocessing

Beyond model architecture, the effectiveness of image-based skin cancer detection also depends on how input data are prepared during training. Variations in acquisition conditions such as lighting or device type can significantly affect model performance, making data preprocessing and augmentation essential steps in most pipelines. Following the categorization proposed by Shorten and Khoshgoftaar (10), augmentation techniques are generally divided into basic image manipulation and deep learning-based approaches. The following sections briefly outline these two complementary approaches.

Basic Image Manipulation. Basic image manipulation methods include a range of traditional operations used to enhance visual diversity and model robustness. They are commonly categorized into geometric transformations, color space transformations, kernel-based filtering, and random erasing. Geometric operations such as rotation, flipping, or cropping modify the spatial structure of images to promote invariance to orientation and scale, while color space transformations adjust illumination or contrast. Kernel filters and random erasing act as regularization techniques by altering texture or masking small image regions.

Deep Learning–Based Approaches. Beyond traditional image manipulation, deep learning techniques have been employed to generate or refine training data. Generative models such as GANs have been used to synthesize realistic dermoscopic images, helping to address class imbalance and increase data diversity. In addition to data generation, deep learning has also been used to enhance image quality through preprocessing operations. In dermatological imaging, this includes lesion segmentation and hair removal to isolate lesions and reduce artefacts. However, recent studies indicate that these operations bring limited benefit to modern architectures. Mahbod et al. (11) reported that segmentation could even degrade performance by removing contextual cues, while Goyal et al. (12) observed only a 0.2% accuracy gain after hair removal, suggesting that current models are already robust to such artefacts.

3.1.5 Challenges in Skin Cancer Detection

Although deep learning has significantly improved the performance of automated skin cancer detection, several important challenges still limit the robustness and generalization of current models. These difficulties are mostly related to the characteristics of dermatological data itself, such as class imbalance, variability between imaging sources, and the presence of visual artefacts that can mislead the model.

Class imbalance. A first major issue is the class imbalance inherent to most skin lesion datasets. Benign lesions vastly outnumber malignant cases, which can cause models to favor the majority class and result in high overall accuracy but low sensitivity to melanoma.

Domain shift. Another limitation is the domain shift between different imaging modalities and acquisition conditions. Datasets used for skin cancer detection originate from various imaging modalities, including dermoscopic and standard clinical photographs captured with devices such as dermatoscopes or digital cameras. Differences in lighting and zoom across these acquisition settings can affect the visual appearance of lesions. A model trained on one type of data may perform poorly on another.

Visual artefacts and contextual cues. In addition, visual artefacts and contextual information such as hair occlusions, ruler markings, or surrounding skin patterns can bias the model's attention. Deep networks sometimes learn to rely on these irrelevant cues instead of the lesion itself, which can lead to unreliable predictions.

3.2 Exploiting Tabular Data and Multimodal Fusion

While image-based models remain central to automated skin cancer detection, many recent studies have explored the integration of additional non-visual information, such as patient demographics or clinical metadata. A multimodal model refers to an architecture designed to process and combine multiple heterogeneous data modalities. This paradigm has become relevant in dermatological challenges such as ISIC 2024, where metadata (e.g., anatomical location, image acquisition metadata, or quantitative lesion descriptors) complements visual information.

In their review, Huang et al. (13) structure existing approaches to multimodal fusion around three widely adopted integration levels: early, intermediate, and late. These categories describe how and when visual and non-visual data interact within the architecture and provide a useful framework for analyzing multimodal fusion strategies. Figure 6 illustrates the typical architectural patterns associated with these three fusion paradigms.

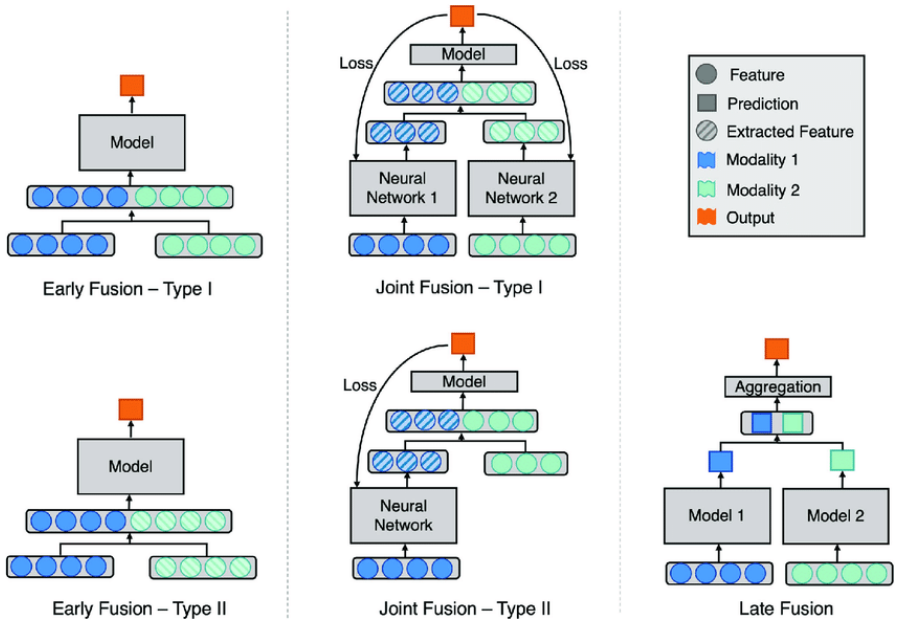


Figure 6: Fusion strategies using deep learning. From Huang et al. (13).

Early Integration

Early integration incorporates metadata directly into the model input before feature extraction. Typical implementations include concatenating metadata with the image input at an early stage of the network. Depending on the level of

processing applied to the metadata prior to fusion, two closely related variants can be distinguished:

- *Early integration type I*: raw metadata are directly fused with the image data, for instance by concatenating them as additional channels before any feature extraction is performed.
- *Early integration type II*: the same fusion principle is applied after the metadata have been transformed into intermediate representations, for example through a dedicated neural network such as a multilayer perceptron.

Intermediate Integration

Intermediate integration combines visual and non-visual features at deeper stages of the network, after the extraction of intermediate image representations. This strategy enables interaction between image and metadata while preserving the structure of the visual encoder. Unlike early integration, intermediate integration relies on jointly optimized neural networks, allowing the prediction loss to be propagated back to the feature extraction modules.

Two variants of intermediate integration can be distinguished depending on whether feature extraction is performed for all input modalities.

- *Intermediate integration type I*: both image and metadata are processed through dedicated feature extraction networks, and the prediction loss is jointly propagated back to all encoders.
- *Intermediate integration type II*: only the image modality is processed through a feature extraction network, while metadata are used as fixed features and integrated at an intermediate stage of the network.

Late Integration

Late integration combines the outputs of separate models trained on different modalities. Each type of input is processed independently through its own network, and the resulting predictions are merged using a fusion method. This approach offers a modular design although it limits interaction between image and metadata during representation learning.

3.3 Solutions from the ISIC 2024 Challenge

The ten top-ranking solutions of the ISIC 2024 Challenge show a strong convergence in their methodology. All winning teams adopted a similar late-fusion architecture that combines deep image encoders with tabular gradient boosting models.

Two-Stage Pipeline with GBDT Models. All top solutions followed a two-stage pipeline. In the first stage, image models based on transformer or hybrid architectures generated predictions or latent embeddings. In the second stage, these predictions were introduced as additional features into gradient boosting models such as LightGBM, XGBoost, or CatBoost.

Image Backbone Trends. The image backbones adopted by the top teams reveal a strong preference for modern transformer-based and hybrid architectures. Vision Transformers such as ViT, DeiT, EVA, and Swin, together with hybrid models like ConvNeXt and EdgeNeXt, were among the most frequent choices. All image models were fine-tuned from pretrained checkpoints, typically ImageNet-1k or ImageNet-21k. Convolutional networks, particularly EfficientNet and ResNet variants, also appeared in several high-ranking solutions. The tenth-place model, which relied solely on EfficientNet-B0, achieved a comparable leaderboard score and highlighted the ongoing relevance of CNN-based architectures.

Ensembling and Model Diversity. All leading teams built their final submission through ensembling. Multiple models were trained with different backbones, and their predictions were averaged or combined using weighted voting.

Data Augmentation and External Data. All top solutions relied on data augmentation to improve the robustness of image models. Most teams adopted augmentation strategies inspired by the winning pipeline of the ISIC 2020 Melanoma Classification challenge. In addition, several teams enriched their training data with images from previous ISIC editions.

Validation and Generalization Strategy. To prevent data leakage between multiple lesions from the same patient, top-performing solutions adopted a five-fold Stratified Group K-Fold validation based on patient identifiers. In addition, some competitors assessed generalization by simulating unseen hospitals in the test set, by splitting the data according to acquisition centers.

These results highlight a methodological convergence among top-performing teams. In the context of this competition, late fusion through gradient boosting models combined with pretrained vision encoders has proven to be an effective strategy for multimodal skin lesion analysis.

4 Convolutional Neural Network Models and Experimental Pipeline

4.1 Image Data Overview

The image data used in this study originate from the ISIC 2024 *Skin Cancer Detection with 3D-TBP* challenge hosted on Kaggle. The official training data for this competition correspond to the SLICE-3D dataset described by Kurtansky et al. (14), which consists of standardized lesion-centered crops obtained from full-body 3D Total Body Photography (3D-TBP). High-resolution 3D captures acquired with the Vectra WB360 system are processed by a software that detects skin lesions across the body surface and extracts each one as an individual image crop.

All images are provided in JPEG format and correspond to non-dermoscopic clinical photographs that resemble smartphone photos. The dataset comprises images collected over nearly a decade (2015–2024) across dermatology centers on three continents. Seven institutions contributed images to the SLICE-3D training dataset, while two additional sites contributed only to the reserved test set of the ISIC 2024 challenge.

In the Kaggle training split used in this work, each cropped image is associated with a binary label indicating whether the lesion is malignant or benign. According to the challenge description, labels originate from two labeling procedures. Strong labels correspond to lesions with histopathological confirmation, while weak labels correspond to lesions assessed as benign based on clinical evaluation without biopsy. This distinction reflects label reliability at the dataset level and is not available as an input feature in the released training data.

Table 4: Summary statistics of the ISIC 2024 SLICE-3D training dataset (Kaggle training split).

Total images	401 059
Non-malignant images	400 666
Malignant images	393
Number of patients	1 042
Collection period	2015–2024
Institutions (training set)	7 (across 3 continents)

4.2 Comparative Study of Image Backbone Architectures

A comparative analysis of multiple backbone networks was conducted to evaluate their performance on skin lesion images. The comparison focuses on convolutional, transformer-based, and hybrid architectures that are representative of the current state of the art in medical image analysis.

The initial selection of convolutional architectures was guided by the benchmarking study of Cassidy *et al.* (6), which provides an evaluation of baseline CNN models on the ISIC 2020 dataset under a consistent training protocol and without pretraining. Based on these findings, architectures such as EfficientNetB0, ResNet50, and DenseNet121 were selected as representative CNN baselines due to their favorable trade-off between performance and computational cost.

The comparison was then extended to transformer-based and hybrid backbones, including ViT, Swin Transformer, and EdgeNext. In this work, all models were evaluated both with and without pretrained initialization. Pretrained variants were initialized with weights obtained from pretraining on the ImageNet-1k dataset, provided by the `timm` library, before being fine-tuned on the dataset.

Computational cost is reported in terms of FLOPs, defined as the number of floating-point operations required for a single model inference. Model performance is evaluated using the partial area under the ROC curve (pAUC), computed over the region where the true positive rate is at least 80%, following the official ISIC 2024 evaluation protocol.

Model	**Params (M)**	**FLOPs**	**pAUC (No Pretraining)**	**pAUC (Pretrained)**
EfficientNet-B0	4.01	0.79 GFLOPs	0.139	0.158
DenseNet121	6.95	5.71 GFLOPs	0.128	0.152
ResNet50	23.51	8.21 GFLOPs	0.121	0.159
ViT-Tiny	5.52	2.16 GFLOPs	0.128	0.16
EdgeNeXt-Small	5.28	1.92 GFLOPs	0.122	0.159
Swin-Small	48.84	17.06 GFLOPs	0.133	0.165

Table 5: Model selection experiment using five-fold cross-validation with best-epoch selection, including model size and computational cost.

Table 5 reports the performance, model size, and computational cost for the architectures considered in this exploratory phase. Because the pAUC values correspond to the best validation epoch in each fold, they should be interpreted as peak performance rather than an estimate of expected generalization. Several observations can be drawn from this comparison:

- Vision Transformers and hybrid architectures outperform CNNs only when pretrained. Without pretraining, convolutional networks remain competitive, while transformer-based and hybrid models are more dependent on strong initialization to achieve their full performance potential.
- Among the convolutional architectures, EfficientNet-B0 offers the best balance between performance and cost, achieving competitive results with the smallest parameter counts and FLOP budgets.
- Pretrained transformer models such as ViT-Tiny and Swin-Small achieve the highest pAUC scores, but their performance advantage over CNNs remains modest relative to their substantially higher computational cost.

These considerations motivate the selection of EfficientNet-B0 as the main convolutional baseline for the subsequent experiments. Moreover, since the experimental setup involves architectural modifications that do not allow the reuse of pretrained weights, the backbone selection also accounts for performance when trained from scratch. In this context, EfficientNet-B0 provides the strongest baseline among the evaluated architectures.

4.3 Baseline Model: EfficientNet Architecture

Following the comparative analysis of backbone architectures, EfficientNetB0 was chosen as the baseline model for this study because it offers an excellent balance between accuracy and computational efficiency. Proposed by Tan and Le in 2019 (15), EfficientNet introduces the concept of compound scaling, a method that jointly adjusts network depth, width, and input resolution in a systematic way. This scaling strategy defines a family of models, from EfficientNet-B0 to EfficientNet-B7, that share the same structural design while differing in scale.

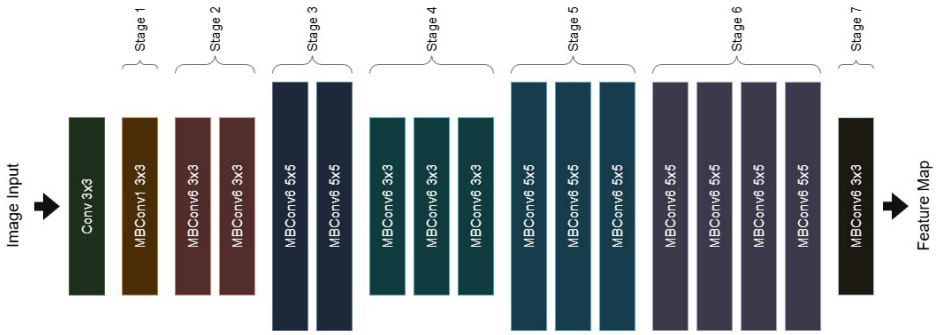


Figure 7: Architecture of EfficientNet-B0 using MBConv as basic building blocks.

EfficientNet builds upon the MobileNetV2 architecture (16) and integrates several mechanisms that improve efficiency. It relies on depthwise separable convolutions, linear bottlenecks, and squeeze-and-excitation (SE) blocks (17), which together reduce computational cost while maintaining strong feature extraction capabilities.

4.3.1 Compound Scaling Principle

The following discussion is based on the work of Tan and Le (2019) (15), who introduced a principled method for scaling convolutional neural networks. In previous approaches, model scaling typically involved modifying only one of three dimensions: network depth, width, or input resolution. Increasing any of these factors can improve accuracy, but scaling a single dimension often leads to excessive computational cost.

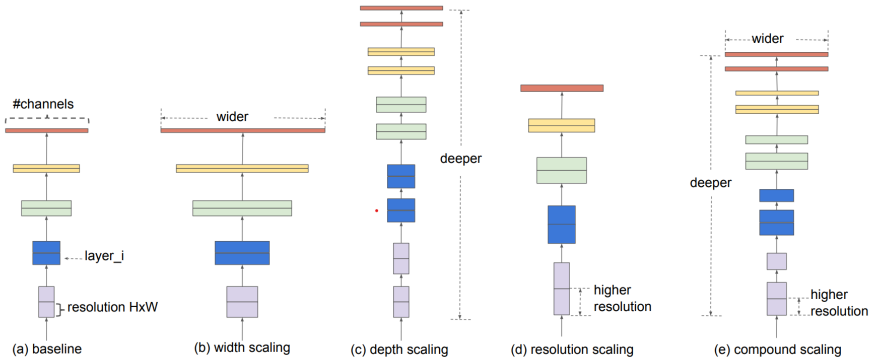


Figure 8: Comparison of scaling strategies for convolutional neural networks, figure from (15).

Intuitively, higher-resolution images naturally require deeper and wider networks in order to capture larger receptive fields and represent richer feature patterns. Determining how to adjust these architectural dimensions in a balanced and computationally efficient manner is however non-trivial. Tan and Le formalized this challenge as a constrained optimization problem in which the model $N(d, w, r)$, defined by its depth d, width w, and input resolution r, is optimized to maximize accuracy under fixed memory and FLOPs budgets:

$$\max_{d,w,r} \ \text{Accuracy}(N(d,w,r))$$

$$\text{Memory}(N) \leq \text{target memory}, \qquad \text{FLOPs}(N) \leq \text{target flops}.$$

The key difficulty is that the three dimensions are mutually dependent in achieving optimal accuracy. Increasing network depth scales computational cost approximately linearly, while increasing width or resolution scales it quadratically. Because modifying any one dimension affects the optimal choices of the others, the three must be tuned jointly. To address this issue, Tan and Le proposed a compound scaling method in which depth, width, and resolution are increased simultaneously according to fixed coefficients α, β, and γ, all controlled through a single compound coefficient ϕ:

$$\text{depth: } d = \alpha^\phi, \qquad \text{width: } w = \beta^\phi, \qquad \text{resolution: } r = \gamma^\phi,$$
$$\text{s.t. } \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2, \qquad \alpha \geq 1,\ \beta \geq 1,\ \gamma \geq 1.$$

The constraint $\alpha\beta^2\gamma^2 \approx 2$ ensures that increasing ϕ by one unit produces an architecture whose computational cost is approximately doubled. Each increment of ϕ increases depth, width, and resolution simultaneously while preserving the computational efficiency of the architecture.

Starting from the baseline EfficientNet-B0 model, Tan and Le determined the optimal scaling coefficients $\alpha = 1.2$, $\beta = 1.1$, and $\gamma = 1.15$ via a grid search. Applying the compound scaling rule with increasing values of ϕ then generates the EfficientNet family from B0 to B7. EfficientNet-B0 represents the base architecture with $\phi = 0$, and higher-index models such as B4 or B7 correspond to increasingly larger settings of ϕ.

4.3.2 MBConv Blocks and Their Internal Mechanisms

The main building block of EfficientNet's architecture is the Mobile Inverted Bottleneck Convolution module. This design is directly inspired by the inverted residual block introduced in MobileNetV2 (16), a lightweight architecture developed to achieve high accuracy under strict computational constraints. The MBConv block inherits the main principles of MobileNetV2, including the use of inverted residual connections and linear bottlenecks.

EfficientNet enhances this design by integrating a channel-wise attention mechanism through the inclusion of a *Squeeze-and-Excitation* (SE) block (17) within the MBConv structure.

Structural Composition of the MBConv Block

The MBConv block follows a structure consisting of an expansion layer, a depthwise convolution, a channel-wise recalibration through a Squeeze-and-Excitation

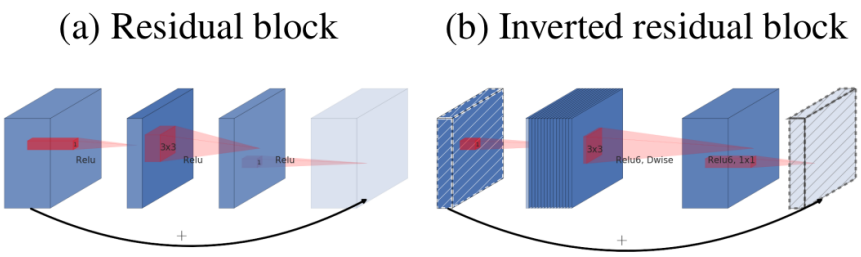


Figure 9: Comparison between (a) a standard residual block as used in ResNet, and (b) an inverted residual block as introduced in MobileNetV2 (16).

(SE) module, and a final projection layer.

Expansion phase (1×1 convolution). A pointwise convolution first expands the input channels by a factor t, projecting the representation into a higher-dimensional space.

Depthwise convolution ($k \times k$). Each channel is then convolved independently using a depthwise operation, which reduces the computational cost compared to a standard convolution. An illustration of the depthwise separable convolution process is shown in Figure 10.

Squeeze-and-Excitation integration. Following the depthwise stage, an *Squeeze-and-Excitation* (SE) module (17) applies a channel attention mechanism, revealing the most informative features. Its detailed mechanism is described in the next subsection.

Projection and residual connection. Finally, a 1×1 projection layer compresses the expanded features back to the original channel dimension without applying nonlinear activation, a design referred to as a *linear bottleneck*. This choice is motivated by the observation that when the number of channels is significantly reduced, nonlinear activations such as ReLU may suppress entire channels by setting all their activations to zero, causing irreversible information loss. Then a residual connection is added when the input and output shapes match, linking the bottleneck representations directly.

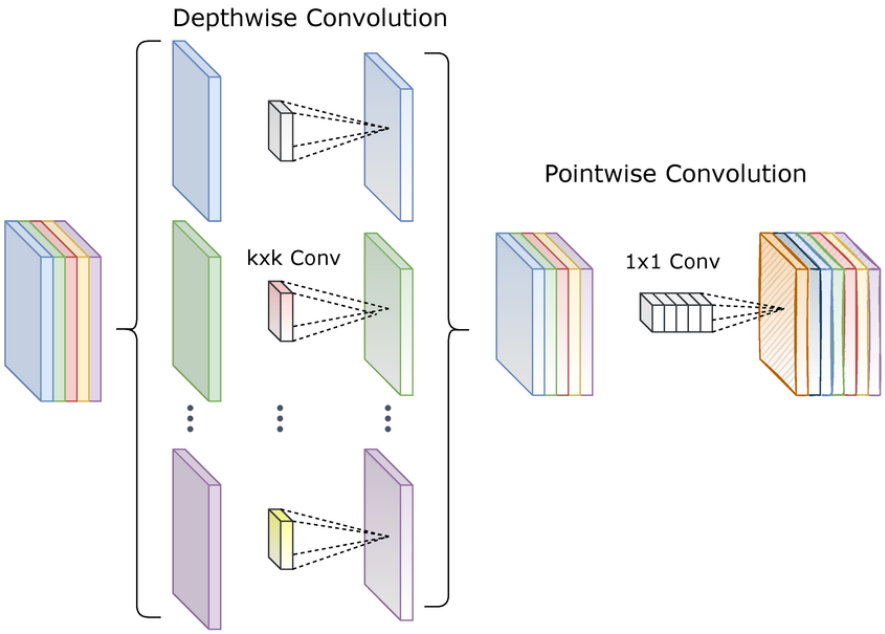


Figure 10: Illustration of the depthwise separable convolution (depthwise convolution followed by pointwise convolution), from (18).

4.3.3 Squeeze-and-Excitation (SE) Module

The *Squeeze-and-Excitation* (SE) module, introduced by Hu et al. (17), enhances the representational power of convolutional neural networks by modeling inter-channel dependencies. In the context of EfficientNet, each MBConv block integrates an SE module to reweight channels using global information.

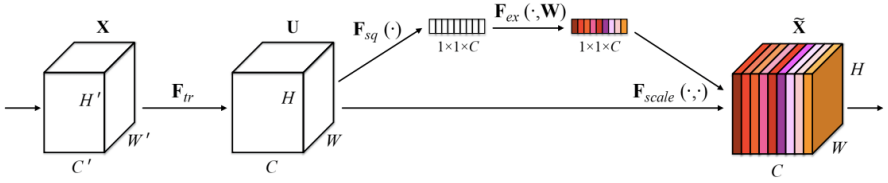


Figure 11: Overview of the Squeeze-and-Excitation (SE) mechanism, illustrating the squeeze, excitation, and recalibration stages. Figure adapted from (17).

Conceptually, the SE module operates through three sequential stages: *squeeze*, *excitation*, and *recalibration*.

Squeeze (Global information embedding). Given an input tensor of feature maps $\mathbf{U} \in \mathbb{R}^{H \times W \times C}$, the squeeze operation condenses spatial information using global average pooling, producing a compact channel descriptor $\mathbf{z} \in \mathbb{R}^C$. Each component of $\mathbf{z}$ represents the average activation of one channel:

$$z_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i, j), \quad c = 1, \ldots, C. \tag{1}$$

Excitation (Channel-wise dependency modeling). The vector $\mathbf{z}$ is then passed through a small fully connected network that captures inter-channel dependencies and generates channel-specific weights:

$$\mathbf{s} = \sigma\left(\mathbf{W}_2\, \delta(\mathbf{W}_1 \mathbf{z})\right), \tag{2}$$

where δ denotes the ReLU activation function, σ the sigmoid function, and $\mathbf{W}_1$, $\mathbf{W}_2$ are learned projection matrices that reduce and then restore the channel dimension.

Recalibration (Feature reweighting). Finally, the SE module applies channel-wise recalibration by rescaling each feature map u_c with its corresponding learned weight s_c:

$$\tilde{u}_c = s_c \cdot u_c. \tag{3}$$

This adaptive reweighting allows the network to enhance meaningful features while suppressing less informative ones.

4.4 Attention Mechanisms Explored

We explored architectural modifications focused exclusively on enhancing attention modeling. The goal of these experiments was to assess how the integration of different attention mechanisms influences accuracy and computational efficiency.

The attention-based extensions explored in this work include the Convolutional Block Attention Module and the Triplet Attention mechanism, both designed to refine feature selection by enhancing spatial and channel information. The following subsections describe these modules in detail.

4.4.1 CBAM Module

Since the SE block in EfficientNet captures only channel-level information, the CBAM module was introduced to incorporate complementary spatial attention while maintaining a negligible computational overhead. The Convolutional Block Attention Module (CBAM), proposed by Woo et al. (19), applies attention sequentially along the channel and spatial dimensions.

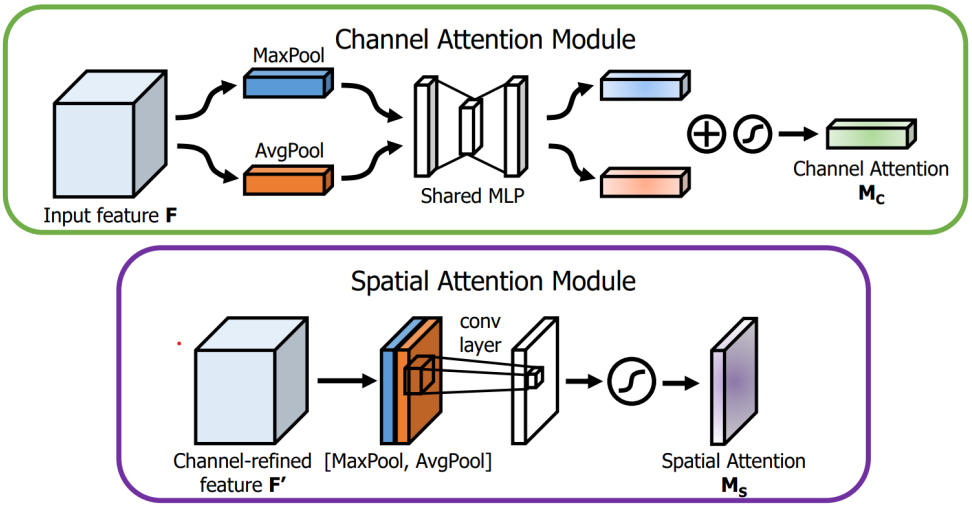


Figure 12: Structure of the CBAM block showing channel and spatial attention submodules. Figure from (19).

The channel attention submodule computes global average pooling and global max pooling descriptors from the input feature map F, and processes them through a shared MLP. The outputs are summed and passed through a sigmoid activation to produce a channel attention vector $M_c(F)$, which adaptively rescales each channel of the feature map:

$$M_c(F) = \sigma(\text{MLP}(\text{AvgPool}(F)) + \text{MLP}(\text{MaxPool}(F))) .$$

The refined feature map is obtained by element-wise multiplication:

$$F' = M_c(F) \odot F.$$

The spatial attention module operates on the channel-refined feature map F'. It first summarizes spatial information by applying average pooling and max pooling across the channel dimension of F', producing two single-channel spatial descriptors. These are concatenated and passed through a convolution with a 7×7 kernel to generate a spatial attention map $M_s(F')$:

$$M_s(F') = \sigma(\text{Conv}_{7\times7}([\text{AvgPool}(F'); \text{MaxPool}(F')])) .$$

The final attention-refined output is obtained as:

$$F'' = M_s(F') \odot F'.$$

4.4.2 Triplet Attention Module

Although CBAM enhances feature refinement by sequentially applying channel and spatial attention, it remains limited in its ability to capture cross-dimensional interactions. In CBAM, the two attention mechanisms are computed independently, meaning that any relationship between channel-wise and spatial patterns is not explicitly modeled. Furthermore, the channel-attention MLP introduces additional parameters that increase the computational cost of the module.

Triplet Attention, proposed by Misra et al. (20), addresses these limitations by introducing an attention mechanism that captures dependencies across three orthogonal views of the input tensor. Triplet Attention models interactions across the (H, W), (C, H), and (C, W) planes by applying rotation operations to the input feature map and computing attention in each transformed orientation.

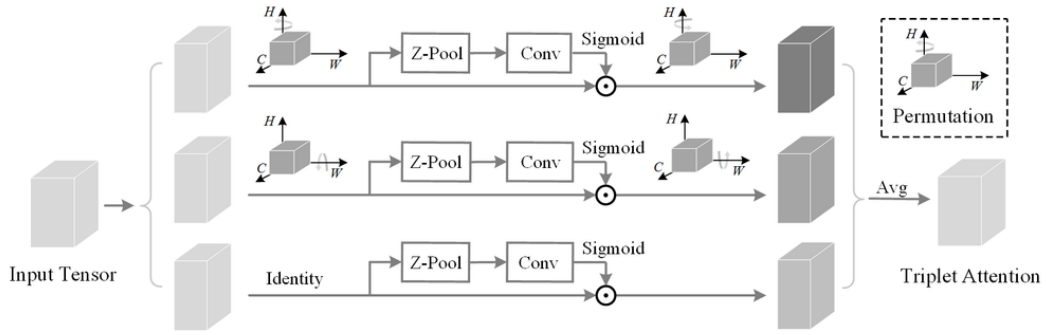


Figure 13: Architecture of the Triplet Attention module, from (20).

Computation in Each Branch. Given an input tensor $\chi \in \mathbb{R}^{C \times H \times W}$, Triplet Attention generates three rotated versions of the input:

$$\hat{\chi}_1,\ \hat{\chi}_2,\ \hat{\chi}_3,$$

each corresponding to one of the three orthogonal interaction planes.

Each rotated tensor is passed through the Z-pool operation, which summarizes information by applying max pooling and average pooling along the appropriate tensor axis, depending on the rotation applied.

$$\hat{\chi}_i^* = \text{ZPool}(\hat{\chi}_i) = [\text{MaxPool}(\hat{\chi}_i);\ \text{AvgPool}(\hat{\chi}_i)]\,,$$

resulting in a tensor of shape $2 \times D_1 \times D_2$, with (D_1, D_2) denoting the pair of dimensions processed in the current branch (one of (H, W), (C, H), or (C, W)).

The pooled representation is then passed through a convolution layer ψ_i followed by a sigmoid activation:

$$A_i = \sigma(\psi_i(\hat{\chi}_i^*)),$$

producing an attention map A_i that is subsequently applied to the corresponding rotated tensor:

$$\hat{\chi}_i' = A_i \odot \hat{\chi}_i.$$

Final Aggregation. After each branch computes its attention-based output, the output is obtained by rotating each tensor back to the original orientation and averaging the three branches:

$$y = \frac{1}{3}\left(\hat{\chi}_1 \odot \sigma(\psi_1(\hat{\chi}_1^*)) + \hat{\chi}_2 \odot \sigma(\psi_2(\hat{\chi}_2^*)) + \hat{\chi}_3 \odot \sigma(\psi_3(\hat{\chi}_3^*))\right), \tag{6}$$

Compared to CBAM, Triplet Attention provides two important benefits. First, it captures cross-dimensional interactions that CBAM does not model. Second, it removes the MLP used in channel attention, which reduces the number of learnable parameters.

4.5 Modified EfficientNetB0 Architecture

We explored the replacement of the standard Squeeze-and-Excitation (SE) blocks with alternative attention mechanisms. Each SE module within an MBConv block can be substituted with either a CBAM module or a Triplet Attention module.

Figure 14 illustrates this substitution scheme. The baseline MBConv block includes an SE module, whereas the modified version replaces this SE component with a Triplet Attention block.

Both CBAM and Triplet Attention include specific architectural hyperparameters. In CBAM, the channel-attention MLP uses a reduction ratio of 16, and the spatial-attention module applies a 7×7 convolution. Triplet Attention also applies a 7×7 convolution in each of its branches. In all experiments, these hyperparameters were kept at the default values specified by the original authors. The implementations of CBAM and Triplet Attention used in this work follow the official GitHub reference code released by the authors.

Several variants of EfficientNetB0 were evaluated by replacing SE modules with either CBAM or Triplet Attention. Each attention mechanism was tested independently as a replacement for the SE block. Among all configurations, the

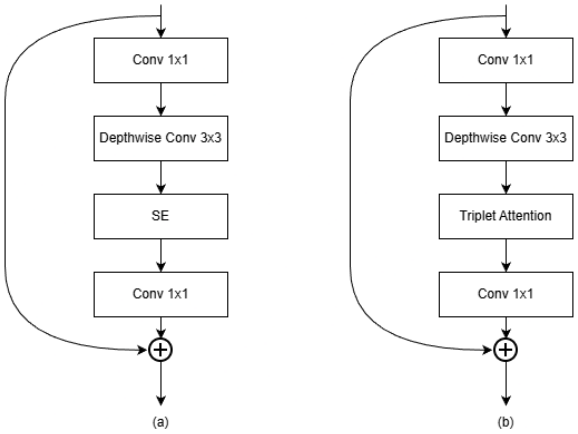


Figure 14: Comparison between (a) the standard MBConv block with SE and (b) the modified MBConv block where SE is replaced by an attention module.

best-performing configuration, identified through cross-validation, used Triplet Attention in the first six MBConv stages while retaining the original SE block in the final stage. This final architecture is presented in Figure 15.

The classification head remained unchanged across all variants. Following the final convolutional stage, global average pooling is applied, followed by a fully connected output layer producing the final classification scores.

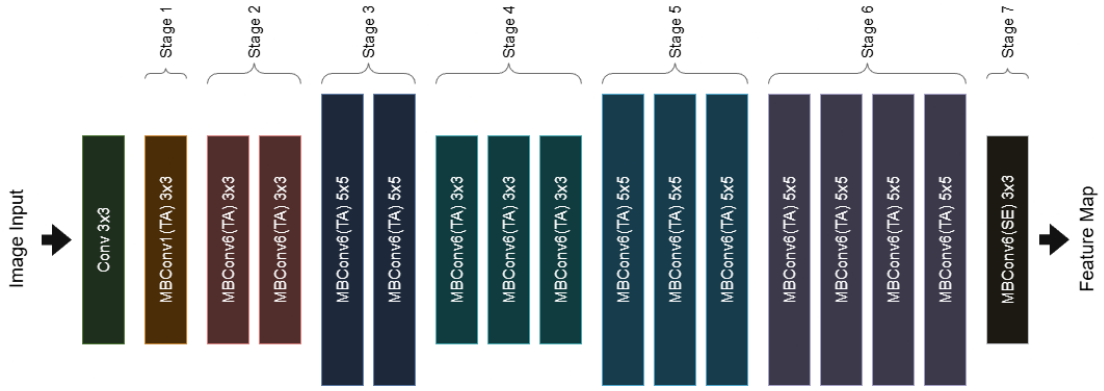


Figure 15: Modified EfficientNetB0 architecture using Triplet Attention (TA) in MBConv stages 1 to 6, with the SE block preserved in the final stage.

4.6 Self-Supervised Representation Learning

In skin cancer detection, obtaining expert-annotated data is both costly and time-consuming, as it requires dermatological expertise and clinical validation. Although the ISIC 2024 dataset provides a large collection of labeled non-dermoscopic images, the broader challenge of limited annotations remains significant in medical imaging. Self-supervised learning provides a way to learn visual representations from unlabeled data. In this work, we use Bootstrap Your Own Latent (BYOL) (21) as the self-supervised pretraining method.

Among existing self-supervised approaches, SimCLR (22) was a popular choice. However, its dependence on a large number of negative samples makes it computationally expensive, as effective training typically requires very large batch sizes to ensure sufficient sample diversity. In this setting, negative samples refer to all other images in the batch that the contrastive loss forces to treat as different from the reference image.

The ISIC 2024 Challenge provides an unprecedented dataset of over 400,000 non-dermoscopic images, offering an opportunity to exploit self-supervised pretraining within the target medical domain. This strategy allows the model to learn domain-specific features before supervised fine-tuning, enhancing generalization and stability in the supervised fine-tuning stage.

Principle of BYOL

Bootstrap Your Own Latent (BYOL) (21) is a self-supervised learning approach that does not rely on negative samples. The method is based on two networks with identical backbone architectures, referred to as the *online* and *target* networks. Both networks process two different augmented views of the same image, obtained through a stochastic data augmentation operator t.

Each branch consists of an encoder f, implemented in our case as an EfficientNet-B0 backbone, followed by a projection head g that maps the encoder output to a projected feature space. In addition, the online network includes a prediction head q, whose objective is to predict the representation produced by the target network.

The online network parameters are updated through gradient descent, while the target network parameters are updated as an exponential moving average of the online network weights, ensuring that the target representations evolve smoothly during training. The training objective minimizes a similarity loss between the

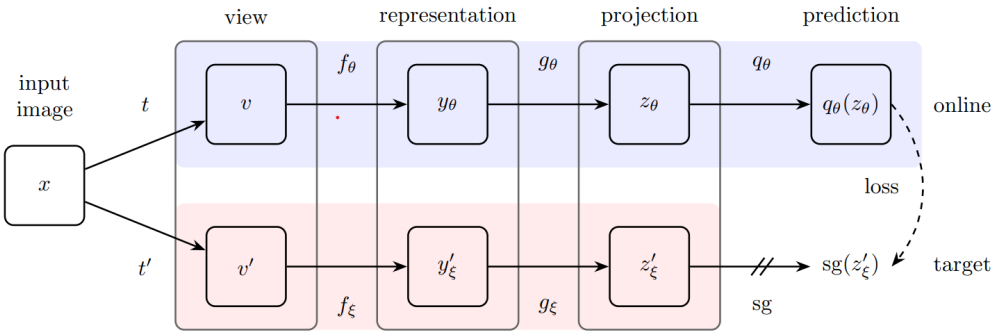


Figure 16: Schematic overview of the BYOL framework. Two augmented views are processed by an *online* and a *target* network. Figure adapted from (21).

online network prediction and the stop-gradient version of the target network projection computed from the alternate augmented view.

This asymmetry in the training procedure addresses the issue of representation collapse, a common problem in self-supervised contrastive methods where the network converges to a trivial solution, mapping all inputs to identical representations that minimize the loss without learning meaningful features.

Integration into the Training Pipeline

In this study, the BYOL framework was integrated into a two-stage training pipeline. In the first stage, the network was pretrained on large collections of unlabeled dermatologic images using self-supervised learning. In the second stage, the pretrained backbone was fine-tuned on the ISIC 2024 Skin Cancer Detection dataset.

Self-supervised pretraining aims to learn general visual representations from unlabeled data, independently of the target classification task. These representations are then adapted to the binary classification objective during supervised fine-tuning. Applying self-supervised learning after supervised training would not further support the target task, as self-supervised objectives do not make use of the target class labels.

Implementation details. EfficientNetB0 enhanced with a Triplet Attention module was used as the backbone encoder within the BYOL framework. The framework was implemented in PyTorch using Lightly's BYOL, with the proposed attention-enhanced backbone integrated.

Pretraining dataset. To maximize data diversity, the pretraining dataset was constructed by aggregating multiple publicly available dermoscopic and non-dermoscopic image collections from the official ISIC repository (23), including the ISIC Challenge datasets (2016–2024), HAM10000, BCN20000, and the "Melanoma and Nevus Dermoscopy" dataset. This aggregation resulted in approximately 480,000 images, representing about a 20% increase compared to the original ISIC 2024 Skin Cancer Detection dataset, which already constitutes the largest publicly available dermatological image collection to date and thus limits further expansion.

Data augmentations. During the self-supervised pretraining phase, we used the default data augmentation pipeline provided by the Lightly Python package, which reproduces the transformations described in the BYOL framework (21). In accordance with the BYOL paper, these augmentations are directly adopted from SimCLR (22), including random cropping and resizing, color jittering, Gaussian blur, and horizontal flipping.

Training parameters. The main hyperparameters used during BYOL pretraining are summarized in Table 6. The hyperparameters closely follow the values reported in the original BYOL paper, except for the batch size, which was reduced due to hardware limitations. Consequently, the LARS optimizer was not employed, as it was originally introduced to stabilize training with very large batch sizes.

At each epoch, the model was trained on only 10% of the training data, randomly sampled from the full dataset. This strategy reduces the computational cost of each epoch while ensuring that, over 1000 epochs, the pretraining procedure still covers the full dataset.

An important observation from the original BYOL study is that self-supervised performance still shows a noticeable dependence on the pretraining batch size. Although BYOL does not rely on negative samples, its representation quality still improves substantially with larger batches. Figure 17 indicates that the decrease in accuracy relative to each method's baseline is consistently smaller for BYOL at small and medium batch sizes. At the same time, BYOL still exhibits a noticeable performance improvement when the batch size increases from 128 to 512.

Table 6: Main training parameters for BYOL pretraining.

Parameter	Value
Pretraining images	484 319
Epochs	1000
Batch size	128
Learning rate	0.2
Weight decay	1×10^{-4}
EMA coefficient	Cosine schedule from 0.996 to 1
Image size	224×224
Optimizer	Stochastic Gradient Descent
Train batches ratio	0.1

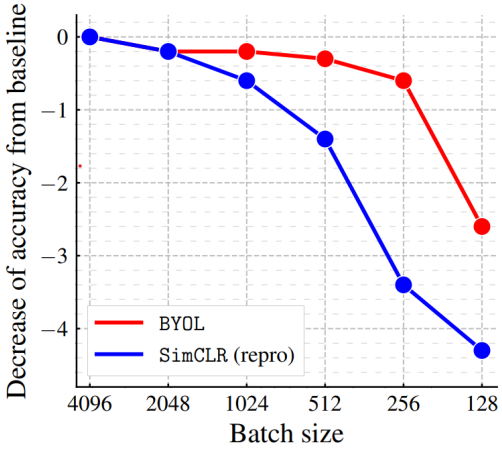


Figure 17: Performance of BYOL and SimCLR across different batch sizes. The y-axis shows the decrease in accuracy relative to each method's own baseline. Figure from (21).

This observation is relevant for this study. Due to hardware limitations, BYOL pretraining was performed with a batch size of 128, which is far smaller than the configuration used in the original paper. Such a reduction limits the effectiveness of the method.

4.7 CNN Preprocessing and Data Augmentation

This section describes the preprocessing steps and data augmentation strategies applied during the training of convolutional models.

Image preprocessing. All images were resized to 224×224 pixels and normalized using the standard ImageNet mean and standard deviation. This resolution was chosen to ensure compatibility with the pretrained EfficientNet-B0 backbone used in this study, which is designed for a 224×224 input size. The same ImageNet normalization was also applied to models trained from scratch to maintain a consistent preprocessing pipeline.

As shown in Figure 18, most lesion-centered crops in the ISIC 2024 dataset range between 110 and 160 pixels. Because this native resolution is below 224 pixels, resizing to 224×224 mainly results in upsampling and does not remove visual information. This supports the choice of a 224×224 input size.

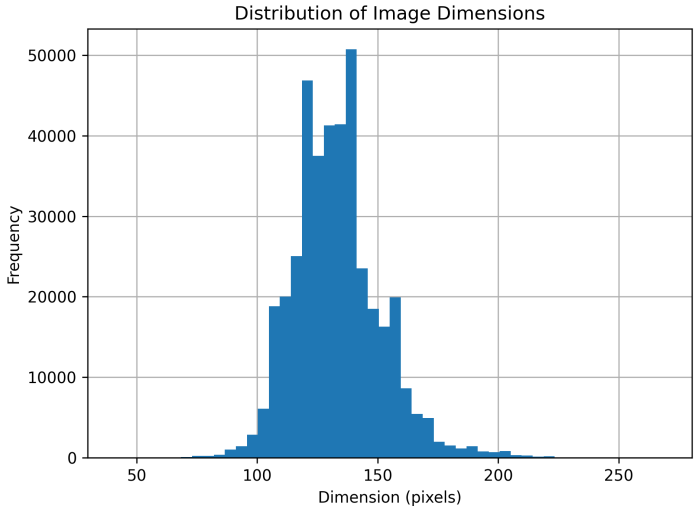


Figure 18: Distribution of image dimensions in the ISIC 2024 dataset.

Data augmentation. The augmentation pipeline was adopted from the winning solution of the ISIC 2020 Challenge proposed by Ha et al. (24) and was implemented using the Albumentations library. It combined geometric and photometric transformations such as random flips, rotations, brightness and contrast adjustments, blurring and elastic distortions, each applied probabilistically.

4.8 Training and Optimization Setup

All convolutional models were trained using the AdamW optimizer and the BCEWithLogitsLoss objective. The learning rate for supervised training was set

to 10^{-3} and scheduled using a cosine annealing strategy defined over 100 epochs, with a minimum value of 10^{-6}. A fixed random seed was used for all experiments unless otherwise specified. For all experiments, training was stopped at a fixed number of epoch determined empirically during preliminary experiments, and this stopping point was applied consistently across all cross-validation folds. The full set of training hyperparameters is summarized in Table 7.

All experiments were implemented using the PyTorch deep learning framework (version 2.8.0, CUDA 12.8). Training and inference were performed with GPU acceleration using an NVIDIA GeForce RTX 5090 GPU. Experiments were conducted on rented cloud-based GPU resources using the vast.ai platform.

Table 7: Training hyperparameters common to all experiments.

Optimizer	AdamW
Loss function	BCEWithLogitsLoss
Epochs	100
Learning rate	1×10^{-3}
Minimum learning rate	1×10^{-6}
Scheduler	CosineAnnealingLR
Weight decay	1×10^{-5}
Training batch size	256
Validation batch size	512
Default seed	42

4.9 Evaluation Metrics and Experimental Protocol

Model performance was assessed using the Partial Area Under the ROC Curve (pAUC), the official evaluation metric of the ISIC 2024 Skin Cancer Detection challenge. The ROC curve summarizes the trade-off between true positive and false positive rates, and the pAUC is computed over the region where the true positive rate is at least 80%, as specified in the competition's evaluation protocol.

To efficiently evaluate the impact of architectural and training pipeline choices, exploratory experiments were conducted on a reduced subset of the training data. Running full evaluations on more than 400,000 images would have been computationally infeasible. The benign class was therefore undersampled to obtain an approximate 1:20 malignant-to-benign ratio.

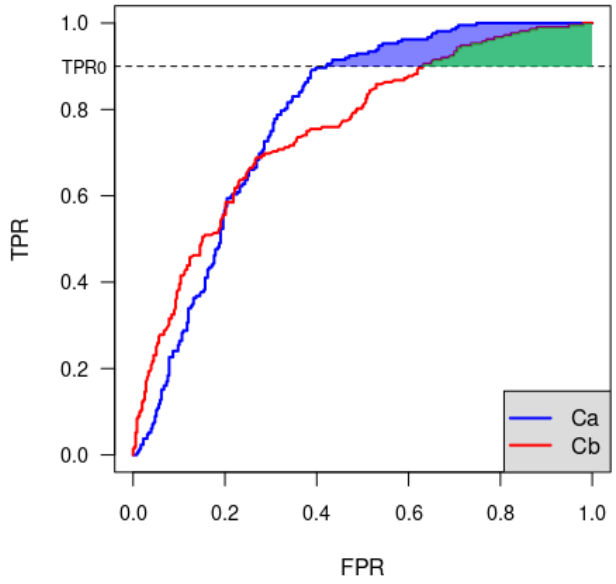


Figure 19: Illustration of pAUC defined by constraining the TPR threshold. Image by ProfGigio (CC-BY-SA-4.0), available at Wikipedia.

All convolutional models were evaluated using a five-fold cross-validation protocol. A stratified group split was applied, ensuring that the class distribution was preserved across folds while preventing patient-level data leakage by keeping all images from the same patient within a single fold. To account for training stochasticity, each experiment was repeated five times using different random seeds (42–46). Models were trained on the same five cross-validation folds for each seed. Reported performance metrics correspond to the mean and standard deviation across these runs, using a fixed number of training epochs determined empirically during preliminary experiments.

4.10 Effects of Class Imbalance

The ISIC 2024 dataset presents an extreme class imbalance, with only 393 malignant cases among 401,059 training images (approximately 1:1,000). To keep experimentation feasible, the main CNN models were trained on a subsampled set with a 1:20 malignant-to-benign ratio. A second series of experiments used a larger subsample with a 1:200 ratio. This setting allows us to examine how severe imbalance affects model behaviour when scaling toward the full dataset.

Batch-Level Class-Balanced Sampling under High Imbalance

In the 1:200 subsampled setting, the composition of training batches is dominated by non-malignant samples, which limits the effective contribution of malignant cases during optimisation. A batch-level class-balanced sampling was evaluated, in which mini-batches are stochastically constructed to approximate a predefined malignant-to-benign ratio. This approach increases the relative frequency with which malignant samples are presented during training while preserving the diversity of the majority class.

Because modifying the class balance increases the risk of overfitting, the evaluation becomes highly sensitive to the choice of training duration. To avoid bias introduced by a fixed stopping point, all results reported in Table 8 correspond to the best validation pAUC observed over the course of training within each fold. This best-epoch criterion provides a more stable and comparable measure of performance across the different imbalance settings and sampling strategies.

Table 8 summarises the best-epoch pAUC obtained under different class imbalance conditions, both with and without batch-level class-balanced sampling. These results were obtained using a single five-fold cross-validation setup. Training on a highly imbalanced dataset-level regime (malignant-to-benign ratio of 1:200) without sampling leads to a decrease in performance compared to training on a dataset-level subsampled 1:20 condition. When batch-level class-balanced sampling is applied to the 1:200 dataset-level setting, the resulting pAUC is restored to a level comparable to the initial 1:20 dataset-level configuration.

Table 8: CNN performance under different class imbalance ratios, with and without batch-level class-balanced sampling. When applied, class-balanced sampling enforces a malignant-to-benign ratio of 1:20 within each batch. All results correspond to best-epoch performance using the EfficientNetB0 backbone with Triplet Attention.

Imbalance Ratio	**Class-Balanced Sampling**	**pAUC (Mean ± Std)**
1:20	No	0.156 ± 0.0085
1:200	No	0.143 ± 0.011
1:200	Yes	0.155 ± 0.010

Training Dynamics Across Epochs

Although class-balanced sampling recovers equivalent best-epoch performance, its effect on training dynamics reveals important limitations. Figures 20 and 21 illustrate the evolution of training and validation pAUC across epochs, without and with class-balanced sampling, respectively. Both curves are obtained from the same cross-validation fold, ensuring that the training and validation data are identical in both settings.

Without class-balanced sampling. As shown in Figure 20, training without sampling shows a gradual and stable increase in both training and validation pAUC over epochs. While validation performance remains lower than that achieved with balanced sampling.

Figure 20: Evolution of training and validation pAUC across epochs under high class imbalance without batch-level class-balanced sampling.

With class-balanced sampling. In contrast, Figure 21 shows that batch-level class-balanced sampling leads to a rapid increase in validation pAUC during the early stages of training. This initial rise is followed by a progressive decline in validation performance as training proceeds, while training pAUC quickly saturates. This divergence between training and validation curves indicates strong overfitting of the minority class.

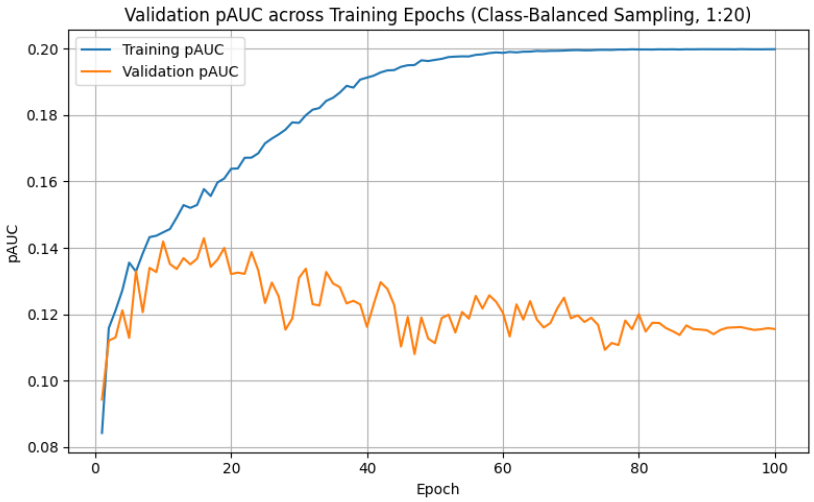


Figure 21: Evolution of training and validation pAUC across epochs with batch-level class-balanced sampling under high class imbalance.

Implications for the Final Training Regime

These results indicate that class-balanced sampling under extreme imbalance can partially compensate in terms of best-epoch performance, but at the cost of unstable training dynamics and accelerated overfitting. In this regime, the combination of a substantially larger number of mini-batches per epoch and batch-level class-balanced sampling leads to frequent reuse of the same malignant samples within a single epoch. This repeated exposure makes training more sensitive to hyperparameter choices and complicates the selection of a stable stopping point.

Given the substantial computational demands required to stabilise class balanced sampling under extreme imbalance, and since this strategy did not produce an improvement over the controlled 1:20 setting, all subsequent experiments involving CNN models trained solely on image data were therefore conducted using the subsampled 1:20 dataset.

4.11 Experimental Results

This section presents the evaluation of the proposed architectural modifications and pretraining strategies. The analysis is divided into two parts. The first part examines the effect of integrating attention mechanisms into the EfficientNetB0 architecture by comparing the baseline model with its CBAM and Triplet Attention variants. The second part evaluates the impact of self-supervised BYOL pretraining in comparison with the supervised ImageNet-1k initialization.

Architectural Variants: Effect of Attention Mechanisms

To analyze the impact of the proposed attention mechanisms, three variants of the EfficientNetB0 architecture were evaluated: the baseline model, a version incorporating CBAM, and a version integrating Triplet Attention. For a fair comparison, both attention modules were inserted following the same architectural strategy. Their performance, parameter count, and computational cost were compared. The results are summarized in Table 9.

Table 9: Comparison of EfficientNetB0 variants (pAUC reported as mean ± std over five repeated five-fold cross-validations).

Model	**Parameters**	**FLOPs**	**pAUC (Mean ± Std)**
EfficientNetB0 (baseline)	4.01M	788.78M	0.13 ± 0.0025
EfficientNetB0 + CBAM	4.26M	799.60M	0.126 ± 0.0028
EfficientNetB0 + TA	3.49M	831.25M	0.134 ± 0.0015

The results indicate that introducing attention mechanisms does not lead to a substantial change in performance relative to the baseline. The CBAM variant slightly underperforms despite a modest increase in model size. The Triplet Attention variant achieves a slightly higher mean pAUC than the baseline. Notably, this comparable level of performance is achieved with a reduced number of trainable parameters, making Triplet Attention a lightweight alternative to the baseline architecture.

To assess whether the observed difference between the Triplet Attention variant and the baseline is statistically meaningful, a paired Student's t-test was performed using the 25 paired pAUC scores obtained from the repeated five-fold cross-validation runs. The resulting p-value of 0.0333 falls below the conventional significance threshold of 0.05, indicating a statistically significant difference in the training from scratch setting.

ImageNet Pretraining vs. BYOL

This experiment compares two pretraining strategies to evaluate their impact on classification performance. The first uses the supervised ImageNet-1k pretrained weights provided by the `timm` library, while the second relies on BYOL pretraining performed on in-domain skin lesion images. The resulting pAUC scores for both the baseline EfficientNetB0 and its Triplet Attention variant are reported in Table 10.

Table 10: Effect of BYOL pretraining compared to the ImageNet-1k initialization provided by the `timm` library.

Model	**pAUC (Mean ± Std)**
EfficientNetB0 (ImageNet-1k pretrained)	0.148 ± 0.0024
EfficientNetB0 (BYOL pretrained)	0.148 ± 0.0040
EfficientNetB0 + TA (BYOL pretrained)	0.149 ± 0.0058

As shown in Table 10, BYOL pretraining yields performance comparable to supervised ImageNet-1k initialization. This result suggests that in-domain self-supervised learning may represent a viable alternative to large-scale supervised pretraining. This observation is particularly relevant in the context of skin cancer detection, where the acquisition of labeled data is costly.

However, the performance advantage observed for the Triplet Attention variant when training from scratch does not persist after pretraining. Once pretrained initialization is introduced, both the baseline and the modified architectures converge to similar performance levels, suggesting that the choice of pretraining strategy plays a more critical role than minor architectural modifications in determining final performance.

Limitations and Potential Improvements

The performance of BYOL needs to be interpreted with respect to the training constraints of this study. The original BYOL paper shows that representation quality improves substantially with large batch sizes. In our experiments, pretraining was limited to a batch size of 128 due to hardware constraints. In addition, only one tenth of the available pretraining dataset was processed at each epoch.

For reference, the supervised ImageNet-1k pretraining used to initialize EfficientNet models relies on a larger optimization budget. As summarized in Table 11, ImageNet-1k pretraining involves more than one million labeled images and very large batch sizes. Despite these different training conditions, the BYOL-pretrained models achieve performance comparable to the ImageNet-1k initialization.

Under the current experimental setting, the performance differences observed between ImageNet-1k and BYOL pretraining remain negligible. Under more favorable training conditions, such as larger batch sizes and full passes over the entire dataset, it is plausible that BYOL pretraining could surpass the performance achieved with ImageNet-1k initialization. However, validating this hypothesis

Table 11: Summary of key training parameters used for supervised ImageNet-1k pretraining of EfficientNet models, based on the official TensorFlow implementation.

Parameter	**Value**
Training images	1,281,167
Validation images	50,000
Test images	100,000
Training epochs	$\sim$350
Batch size	2048
Training paradigm	Supervised classification
Pretraining dataset	ImageNet-1k

would require substantially greater computational resources and more extensive pretraining.

5 Multimodal Integration of Image and Tabular Data

5.1 Metadata Description

The ISIC 2024 Skin Cancer Detection dataset provides a diverse set of metadata associated with each image. These variables describe demographic information, anatomical characteristics of the lesion, and properties related to the image acquisition process. Since the objective of this work is to evaluate multimodal fusion rather than to optimize a tabular model on its own, all available metadata were incorporated without performing feature selection or feature engineering.

Metadata Available at Train and Test Time

The metadata fields used in this study include demographic attributes such as the approximate age of the patient and the recorded sex. Anatomical descriptors are also provided, including the body region where the lesion is located. Several variables characterize the acquisition process, for example the type of lighting used, the acquisition center, and the origin of the image. The dataset also contains a large number of numerical descriptors derived from the 3D Total Body Photography pipeline. These descriptors quantify aspects of lesion geometry, pigmentation, colour variation, and spatial location. Table 12 lists the metadata fields included in the multimodal model.

Table 12: Metadata fields included in the multimodal model.

Field	Description
`age_approx`	Approximate patient age
`sex`	Patient sex recorded in the metadata
`anatom_site_general`	General anatomical region of the lesion
`clin_size_long_diam_mm`	Longest diameter of the lesion in millimetres
`image_type`	Image type field from the ISIC Archive
`tbp_tile_type`	Lighting modality used in the TBP system
`attribution`	Field indicating the image source
`tbp_lv_*`	Set of numerical descriptors extracted from the TBP system

Excluded Metadata Fields

Some metadata fields appear only in the training file and contain diagnostic information or biopsy-derived measurements, including diagnostic labels at dif-

ferent levels of detail and melanoma thickness values. These variables are not available in the test set and are directly linked to the target label. For this reason, all diagnosis-related and biopsy-based fields were excluded from the multimodal pipeline. Table 13 lists the excluded metadata fields that appear exclusively in the training file.

Table 13: Metadata fields present only in the training file and excluded from the multimodal model.

Field	**Description**
`lesion_id`	Identifier for lesions manually tagged as lesions of interest
`iddx_1` to `iddx_5`	Diagnostic labels defined at multiple levels of detail
`iddx_full`	Complete diagnostic label
`mel_thick_mm`	Measured melanoma thickness
`mel_mitotic_index`	Melanoma mitotic index
`tbp_lv_dnn_lesion_confidence`	Confidence score from an external lesion classifier

5.2 Preprocessing and Encoding of Tabular Features

The metadata associated with the ISIC 2024 Skin Cancer Detection dataset contain very few missing values. Only three fields present missing entries in the training set. Table 14 reports the proportion of missing values expressed as percentages.

Table 14: Proportion of missing values in the metadata (training set).

Field	**Missing values (%)**
`sex`	2.84
`anatom_site_general`	1.10
`age_approx`	0.68

For numerical metadata, missing values were imputed using a K-Nearest Neighbours (KNN) imputer, which estimates each missing entry from the values of similar samples. For categorical metadata, missing entries were mapped to an additional category labelled *"missing"*.

Following imputation, categorical variables were encoded with one-hot encoding. This transformation expanded the metadata from the 40 selected fields to a total of 75 features.

5.3 Fusion Strategy

Multimodal integration can be implemented in different ways depending on how image information and metadata interact within the pipeline. In this work, both intermediate fusion and late fusion strategies are explored. This choice aligns with the objective of preserving the pretrained convolutional encoder obtained from BYOL.

Three fusion strategies are explored. The first two approaches implement intermediate fusion by combining image features and metadata within a single neural network. The third strategy implements late fusion, following the approach adopted by many participants in the ISIC 2024 challenge, where the CNN prediction is treated as an additional feature and combined with metadata in a separate gradient boosting model.

5.3.1 Fusion by Concatenation

The first method relies on a concatenation of feature vectors. The metadata are processed with a multilayer perceptron that produces an embedding. Let $\mathbf{f}_{\text{img}} \in \mathbb{R}^d$ be the feature vector extracted by the CNN and $\mathbf{f}_{\text{tab}} \in \mathbb{R}^k$ be the output of the MLP. The fused representation is defined by:

$$\mathbf{f}_{\text{fusion}} = [\mathbf{f}_{\text{img}} \parallel \mathbf{f}_{\text{tab}}]$$

This approach is commonly used in multimodal neural networks. The CNN operates mostly independently from the metadata, and the classification head learns how both embeddings should be combined for prediction. The design adopted here follows the multimodal integration strategy described by Ningrum et al. (25), where image features and patient metadata are concatenated directly within the final classification head.

5.3.2 Fusion by Hadamard Modulation

The second approach combines the two modalities through a multiplicative interaction. The metadata MLP generates a gating vector $\mathbf{g} \in \mathbb{R}^d$, aligned in dimension with the CNN feature vector. The fused vector is obtained through the element-wise product

$$\mathbf{f}_{\text{fusion}} = \mathbf{f}_{\text{img}} \odot \mathbf{g}.$$

This mechanism allows the metadata to adjust the visual features by increasing or decreasing the contribution of each dimension, thereby altering the visual representation learned by the network.

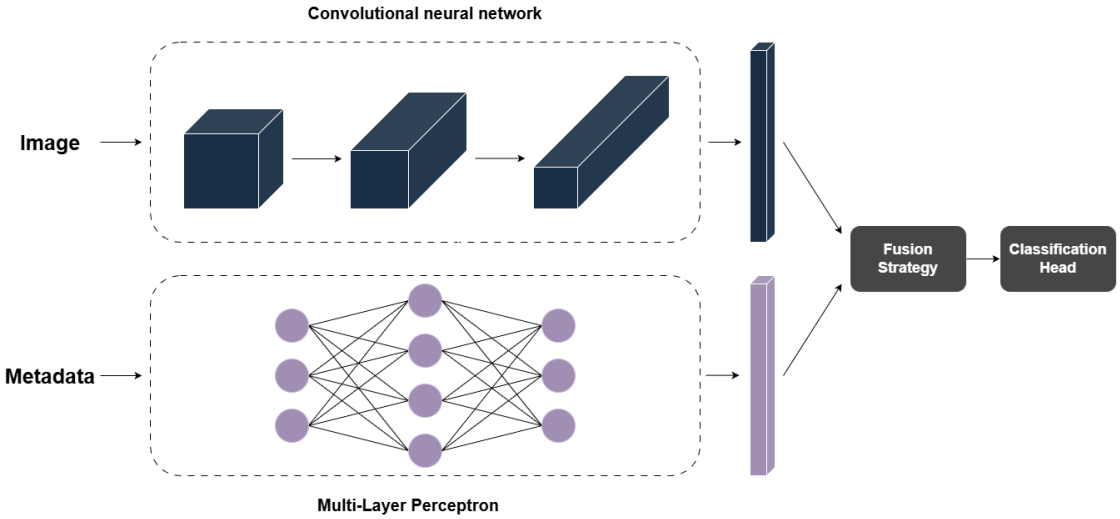


Figure 22: Neural multimodal integration through concatenation or Hadamard modulation.

5.3.3 Fusion through Gradient Boosting

A third strategy is also examined. In this configuration, the CNN produces a prediction probability, which serves as an additional feature. This information is then combined with the metadata and provided as input to a gradient boosting model.

This approach follows the methodology adopted by many teams in the ISIC 2024 challenge, where models such as LightGBM, XGBoost and CatBoost were commonly used for metadata. The CNN and the gradient boosting model operate independently, and the fusion takes place when the CNN-derived feature is provided as an additional input alongside the metadata.

The following paragraphs summarise the key principles of gradient boosting, as described in (26). noting that models such as LightGBM, XGBoost and CatBoost are optimised variants of this method, the focus here is limited to its general principles.

Principles of Gradient Boosting Gradient boosting constructs a model by iteratively adding weak learners. The ensemble takes the form

$$F_M(x) = F_0(x) + \sum_{m=1}^{M} \rho_m h_m(x),$$

where each learner h_m is trained to reduce the residual errors of the current model estimate, and ρ_m denotes a scaling coefficient that controls how much the m-th learner contributes to the updated model. The initial term $F_0(x)$ is a constant prediction, chosen as the value minimizing the loss. At each boosting step, a new learner is fitted to approximate the remaining prediction error, and its weight ρ_m is then chosen to minimize the loss.

Regularization and Model Control Two mechanisms play a central role in controlling model complexity. The first is the learning rate ν which scales the contribution of each boosting step

$$F_m(x) = F_{m-1}(x) + \nu\, \rho_m h_m(x),$$

The second is the number of boosting iterations M, which limits the overall expressiveness of the ensemble and acts as a form of regularization. Tree-based learners may further incorporate constraints such as depth limits or minimum samples per leaf.

The three fusion strategies described above offer complementary perspectives on multimodal integration.

5.4 Experimental Setup for Multimodal Models

The multimodal models were evaluated using the same protocol as the convolutional baselines. All experiments relied on a 5-fold stratified group cross validation ensuring that images from the same patient were kept within a single fold. In all multimodal configurations, image features were extracted using the EfficientNetB0 backbone enhanced with Triplet Attention, which served as the reference model for concatenation, Hadamard modulation, and to generate the CNN prediction used by the gradient boosting models.

For the gradient boosting models, the training hyperparameters were determined through a dedicated optimisation procedure using Optuna, which relies on Bayesian optimisation to efficiently explore the hyperparameter space.

5.5 Multimodal Results

This section summarizes the performance of the evaluated multimodal integration strategies compared to the image-only baseline. The results are reported in Table 15.

Table 15: Performance comparison between unimodal image-only baseline and multimodal integration strategies (mean ± standard deviation over 5 repeated 5-fold cross-validation).

Model	**pAUC (Mean ± Std)**
Unimodal image-only baseline	
EfficientNetB0 + TA (BYOL pretrained)	0.149 ± 0.0058
Multimodal integration	
EfficientNetB0 + TA + Hadamard integration	0.15 ± 0.0031
EfficientNetB0 + TA + Concatenation integration	0.148 ± 0.0015
XGBoost (metadata + CNN prediction)	0.164 ± 0.0022
LightGBM (metadata + CNN prediction)	0.165 ± 0.0011
CatBoost (metadata + CNN prediction)	$\mathbf{0.168 \pm 0.0015}$

Two main observations arise from this evaluation. First, integrating metadata through concatenation or Hadamard modulation does not improve performance over the image-only CNN baseline. This suggests that the network does not effectively exploit the metadata embedding when fusion occurs through a late MLP-based interaction, despite the metadata being highly informative on their own.

Second, gradient boosting models clearly outperform all neural multimodal architectures. CatBoost achieves the highest pAUC, followed by LightGBM and XGBoost, demonstrating the strong suitability of tree-based methods. This behaviour reflects the trends observed in the ISIC 2024 challenge, where top-performing solutions rely almost exclusively on gradient boosting for multimodal fusion rather than CNN-based integration.

5.6 Kaggle Leaderboard Evaluation

The Kaggle leaderboard evaluation was conducted using the best performing multimodal configuration identified through cross validation. This configuration relies on a CatBoost classifier combining metadata features with image based predictions. The image predictions were obtained from a modified EfficientNetB0 backbone incorporating Triplet Attention and pretrained in domain using BYOL before fine tuning.

The obtained leaderboard scores are summarized below:

- Public leaderboard pAUC: 0.16964
- Private leaderboard pAUC: 0.15217

These results are consistent with the cross validation estimate reported in Table 15, where the CatBoost multimodal model achieved a pAUC of 0.168 ± 0.0015. This confirms that the cross validation protocol provides a reliable estimate of public leaderboard performance.

A decrease in performance is observed between the public and private leaderboard scores, a behaviour observed across the majority of participants. Figure 23 shows the cumulative distribution functions of public and private scores for all submissions. The private score distribution is consistently shifted toward lower values, which highlights the increased difficulty requirements of the private test set.

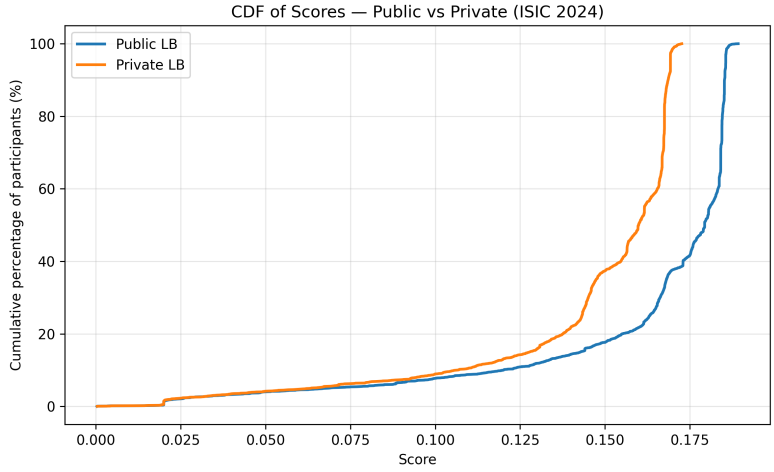


Figure 23: Cumulative distribution functions of public and private leaderboard scores for the ISIC 2024 challenge.

This submission does not correspond to an official competition entry, as the ISIC 2024 challenge was already closed at the time of evaluation. The reported private score would correspond to a ranking of 1687 out of 2739 submissions. Within the context of this master's thesis, the objective was not to achieve a competitive leaderboard position. As a result, the obtained score is relatively low but remains consistent with the proposed methodology. The approach relies on a single convolutional neural network backbone and a single gradient boosting

model pretrained in domain using BYOL, without relying on large scale public datasets for robust initialization. In contrast, many top ranking solutions relied on strategies that combined a large ensemble of image models, together with multiple gradient boosting models.

6 Explainability Analysis using Grad-CAM

While pAUC is used throughout this work to evaluate performance, it does not provide explanations about how predictions are formed. In medical image analysis, strong performance values alone are insufficient if the predicted malignancy probability is influenced by visual artefacts or irrelevant image information.

To address this limitation, this study employs Gradient-weighted Class Activation Mapping, to analyse the image regions that most strongly influence model predictions. By comparing saliency maps across architectures and pretraining strategies, the analysis examines whether improvements in pAUC are reflected in more consistent and lesion-focused attention patterns.

Before presenting the qualitative results, the next subsection summarizes the Grad-CAM methodology.

6.1 Grad-CAM Methodology

Overview of the Grad-CAM method. Gradient-weighted Class Activation Mapping (Grad-CAM), introduced by Selvaraju *et al.* (27), is used to analyse which image regions most strongly influence the model output. The method generates a saliency map based on the gradients of the model output with respect to convolutional feature maps.

Here, y^c refers to the score associated with class c, and A^k denotes the k-th feature map of the selected convolutional layer. For each feature map, an importance weight is computed by globally averaging the gradients of y^c with respect to the spatial activations:

$$\alpha_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k},$$

where Z is the number of spatial locations in the feature map.

The Grad-CAM saliency map is then obtained as a weighted linear combination of the feature maps followed by a ReLu:

$$L_{\text{Grad-CAM}}^c = \text{ReLU}\left(\sum_k \alpha_k^c A^k\right).$$

The resulting heatmap is upsampled to the input image resolution and overlaid on the original image for visual inspection.

Application of Grad-CAM in this study. This study employs the Grad-CAM++ variant, which provides more accurate localization of the regions relevant to the model's prediction. Grad-CAM++ is applied to feature maps extracted from the fifth MBConv block of the EfficientNetB0 backbone, which has a spatial resolution of 14×14. This layer represents a compromise between spatial detail and the representation of features that are most relevant to the model's prediction and interpretation.

In all Grad-CAM visualisations presented in this work, warmer colors (red) indicate image regions that contribute more to the model's prediction, whereas cooler colors (blue) correspond to regions with little or no influence.

Explainability analysis is carried out exclusively on malignant samples. In the context of skin cancer detection, malignant cases are clinically critical and represent the cases of greatest diagnostic importance. They are also heavily underrepresented in the dataset, yet they contribute most to the pAUC score. For all visualisations, Grad-CAM++ is therefore computed with respect to the malignant class logit.

6.2 Effect of Triplet Attention

This subsection analyses the effect of integrating Triplet Attention into the EfficientNetB0 architecture under in-domain self-supervised pretraining.

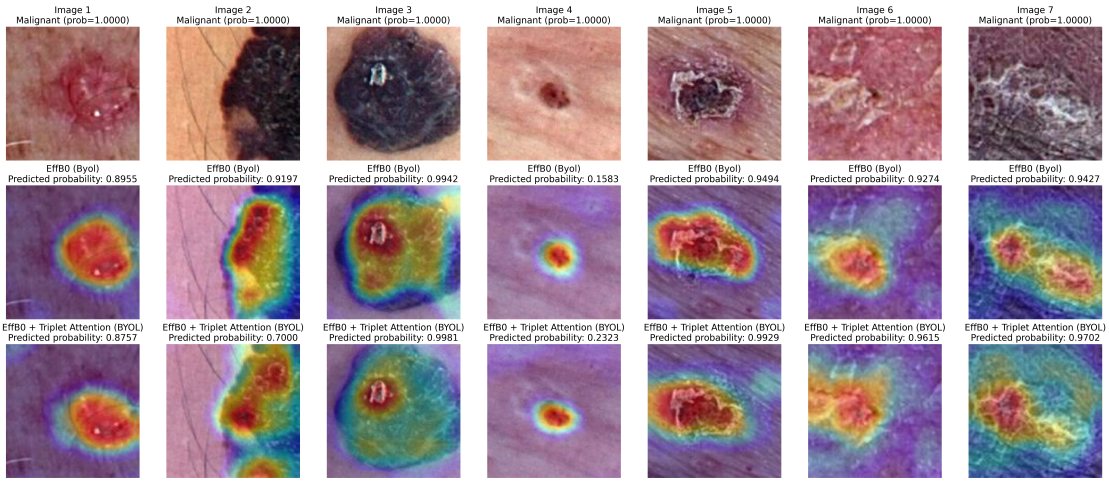


Figure 24: Grad-CAM visualisations for the baseline EfficientNetB0 and its Triplet Attention variant, both pretrained using BYOL on in-domain dermatological data. For each example, the predicted malignancy probability is reported for reference.

Figure 24 shows that the baseline EfficientNetB0 and its Triplet Attention variant produce comparable saliency maps. In both cases, attention is primarily concentrated on the lesion. The introduction of Triplet Attention does not result in a noticeably more precise or more localized attention pattern, as both architectures exhibit similar spatial coverage of the lesion. Overall, it does not appear to meaningfully alter how the model allocates attention within the image.

6.3 Effect of Pretraining

This subsection analyses the effect of different pretraining strategies on Grad-CAM saliency maps for the baseline EfficientNetB0 architecture. Specifically, models trained from scratch, pretrained on ImageNet-1k, and pretrained using BYOL on in-domain dermatological data are compared in order to assess how pretraining influences spatial attention patterns.

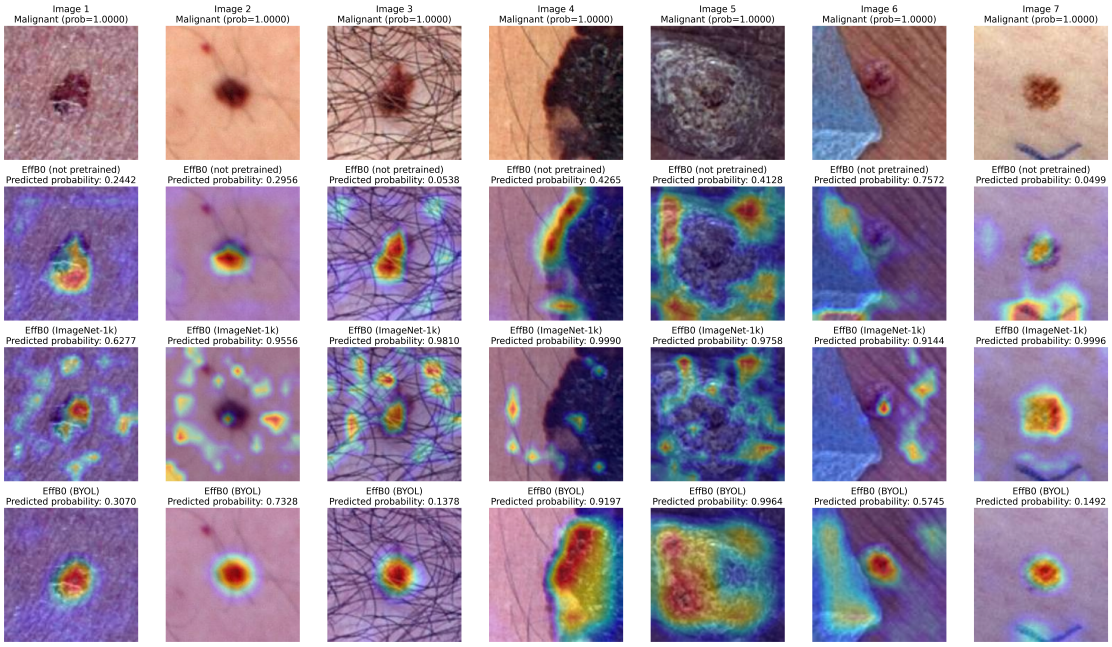


Figure 25: Grad-CAM visualisations for the baseline EfficientNetB0 trained from scratch, pretrained on ImageNet-1k, and pretrained using BYOL on in-domain dermatological data.

Impact of In-Domain Self-Supervised Pretraining (BYOL) Figure 25 shows that in-domain BYOL pretraining leads to more consistent and lesion-centred attention patterns compared to training from scratch. This effect is particularly

pronounced in visually extreme lesions, whose appearance differs substantially from benign patterns such as moles. In these cases, BYOL-pretrained models show clearer focus on the lesion, suggesting improved robustness to variations within the malignant class.

In addition, BYOL pretraining appears to reduce the model's reliance on certain visual artefacts. As illustrated in images 6 and 7, the attention assigned to distracting elements such as a blue cloth or marker traces on the skin is noticeably reduced compared to the scratch model.

Comparison with ImageNet-1k Pretraining In contrast, the ImageNet-1k–pretrained model exhibits more heterogeneous saliency patterns. While certain cases show activation on specific local structures within the lesion, other examples display highly dispersed attention across the image. This behaviour suggests a greater reliance on non lesion-centred visual cues, possibly influenced by surrounding skin texture or artefacts such as hair.

Overall, these observations indicate that in-domain pretraining encourages more lesion-centred attention, which aligns better with the lesion-focused visual inspection used in dermatological practice. Although ImageNet-1k pretraining achieves performance comparable to in-domain self-supervised pretraining, this in-domain approach provides explanations that better align with clinically visual reasoning.

While these visualisations are informative, Grad-CAM has inherent limitations. The method produces low resolution saliency maps that highlight broad regions of interest rather than pixel-level contributions. As a result, although it is possible to identify which areas of an image are important for the prediction, Grad-CAM does not reveal which specific textures or visual cues within these regions are actually driving the model's decision.

7 External Validation on a Dermoscopic Dataset

7.1 Dataset Description

External validation was conducted using the DERM12345 dataset (28), a large dermoscopic image collection comprising 12,345 lesion images acquired from 1,627 patients in Türkiye. The dataset consists of dermoscopic images, which differ substantially from the non-dermoscopic clinical photographs used in the ISIC 2024 SLICE-3D dataset.

Only lesions with reliable diagnostic confirmation were included. According to the dataset authors, each case was validated through a consensus diagnosis by two expert dermatologists, supported when possible by patient follow-up or by histopathological confirmation.

For the purposes of this study, the binary classification labels were taken directly from the `diagnosis_1` field provided in the DERM12345 dataset. This field assigns each lesion to one of three categories: benign, malignant, or indeterminate. Only benign and malignant samples were retained for external validation. The resulting class distribution is shown in Table 16.

Table 16: Class distribution of the DERM12345 external validation dataset.

Class	**Number of Images**
Benign	11,120
Malignant	1,167
Indeterminate	58

The composition of the malignant class in DERM12345 differs from the one used in the ISIC 2024 SLICE-3D dataset. While both datasets group clinically serious lesions under a single malignant category, the exact composition of this category is not identical, as each dataset includes a slightly different set of diagnoses. External evaluation therefore does not replicate the original ISIC 2024 prediction task exactly. This provides an assessment of robustness across datasets that differ in both imaging modality and diagnostic categorisation.

7.2 Evaluation Setup

External validation is limited to the image-only CNN models because no publicly available dermoscopic dataset provides metadata compatible with those used in the ISIC 2024 Skin Cancer Detection challenge. As a result, multimodal architectures cannot be evaluated in this setting.

The external evaluation follows the same preprocessing and inference pipeline as the ISIC 2024 internal validation procedure. All images were resized to 224×224 pixels and normalized using the ImageNet statistics, consistent with the preprocessing used during model training. No fine-tuning or parameter adjustment was performed on the DERM12345 dataset.

Performance was assessed using AUROC and pAUC, computed with the same implementation as in the ISIC 2024 challenge. No additional model adjustment or re-training was performed on the external data.

7.3 Validation Results

Table 17 reports the performance of the image-only models under internal cross-validation on the ISIC 2024 dataset and external validation on the DERM12345 dataset. For each architecture, five independent fine-tuning runs were performed on the ISIC 2024 training data using different random seeds, and the reported AUROC and pAUC values correspond to the mean and standard deviation over these runs.

The comparison includes a baseline EfficientNetB0 pretrained on ImageNet-1k, the same architecture pretrained using BYOL on in-domain dermatologic images, and the proposed EfficientNetB0 enhanced with Triplet Attention and pretrained in-domain using BYOL. The internal cross-validation pAUC scores reported here are taken from Table 10, with the addition of AUROC to enable comparison with the external validation results.

Table 17: Internal cross-validation and external validation performance for image-only models. Results for external validation are reported as mean ± standard deviation over five independent runs.

Model	**AUROC**	**pAUC**
Internal cross-validation (ISIC 2024)		
EfficientNetB0 (ImageNet-1k)	0.928 ± 0.0022	0.148 ± 0.0024
EfficientNetB0 (BYOL, in-domain)	0.927 ± 0.0056	0.148 ± 0.004
EfficientNetB0 + TA (BYOL, in-domain)	0.929 ± 0.0075	0.149 ± 0.0058
External validation (DERM12345)		
EfficientNetB0 (ImageNet-1k)	0.904 ± 0.02	0.137 ± 0.013
EfficientNetB0 (BYOL, in-domain)	0.841 ± 0.037	0.102 ± 0.017
EfficientNetB0 + TA (BYOL, in-domain)	0.886 ± 0.013	0.124 ± 0.0045

Overall, the ImageNet pretrained EfficientNetB0 achieves the strongest performance on both datasets. The BYOL-pretrained model enhanced with Triplet Attention obtains slightly lower scores but remains competitive. Under internal cross-validation, the three architectures exhibit comparable performance. However, in external validation, the EfficientNetB0 enhanced with Triplet Attention consistently outperforms the BYOL-pretrained baseline. This observation suggests that incorporating attention mechanisms that jointly model spatial and channel relationships may contribute to improved generalization.

Limitations related to pretraining data composition. One limitation of this evaluation concerns the composition of the data used for in-domain pretraining. Although the BYOL pretraining dataset includes multiple public dermatological collections, a substantial proportion of the images originate from the ISIC 2024 challenge and consist of non-dermoscopic clinical photographs. This data composition may introduce a bias toward non-dermoscopic image characteristics and could partly account for the lower performance observed during external validation on the dermoscopic DERM12345 dataset.

7.4 Grad-CAM Analysis on External Dermoscopic Images

Before analysing model behaviour on DERM12345, it is important to note that this dataset contains several dermoscopy-specific artefacts. Many images exhibit dark circular borders from the dermatoscope lens and shadowed corners, which may influence the model's attention.

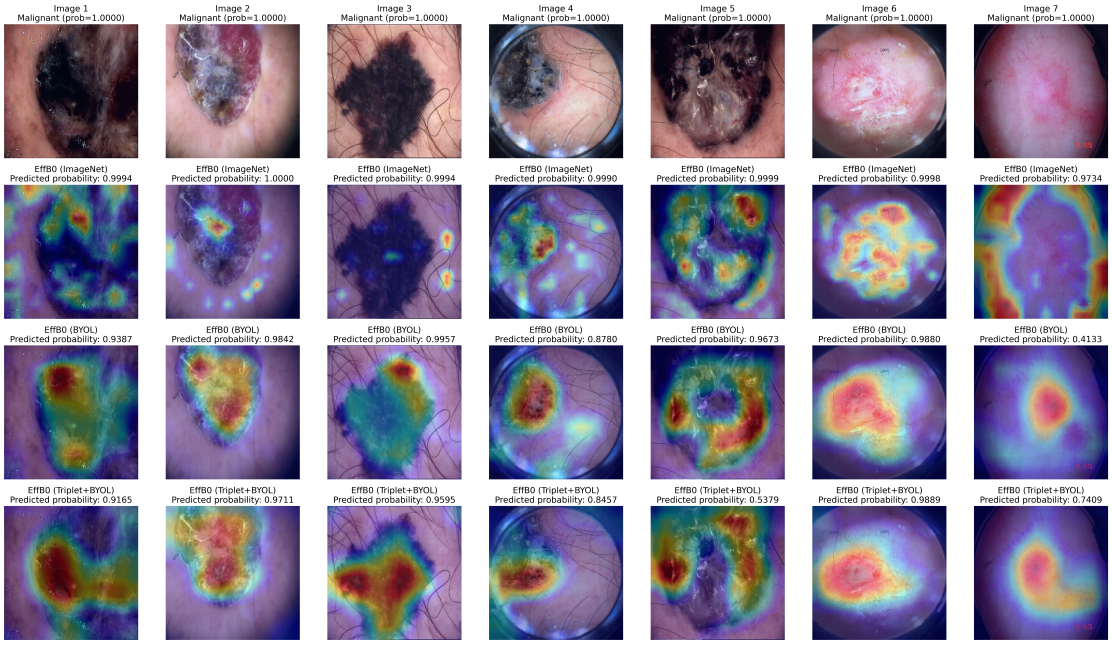


Figure 26: Grad-CAM visualisations on DERM12345 for three models: ImageNet-pretrained EfficientNetB0, BYOL-pretrained EfficientNetB0, and the BYOL-pretrained Triplet Attention variant.

Comparison of attention patterns across pretraining strategies. Despite this change in imaging modality, the two models pretrained in-domain with BYOL generally direct most of their attention toward the lesion. Across many examples, the Grad-CAM heatmaps remain centred on the lesion region, which suggests that the representations learned through in-domain self-supervised pretraining transfer well to dermoscopic images. More broadly, all three evaluated models show robustness to artefacts characteristic of the DERM12345 dataset, such as dark corner regions and the circular lens border.

In contrast, the EfficientNetB0 model pretrained on ImageNet-1k displays a different attention pattern. The saliency maps often show a diffuse spread across the lesion and surrounding skin, a behaviour consistent with what was previ-

ously observed on the ISIC 2024 non-dermoscopic images. This suggests that the ImageNet-pretrained model may rely on more general texture information rather than specific lesion regions. In several examples, such as Images 5 and 6, the model appears to respond to texture patterns within the lesion. In other cases, illustrated by Image 7, the attention spreads around the edge of the lesion while avoiding corner artefacts.

Role of surrounding context in model attention. A relevant observation provided by Bissoto *et al.* (29) showed that deep models can achieve relatively high AUROC scores even when the lesion itself is masked. Their results show that networks are able to exploit information present outside the lesion region. This phenomenon offers a plausible interpretation for the diffuse attention observed in the ImageNet-pretrained model, which often focuses on areas beyond the lesion

Overall, the external Grad-CAM analysis highlights qualitative differences in attention patterns across pretraining strategies. In-domain BYOL pretraining is associated with more lesion-centred saliency maps, whereas ImageNet-1k pretraining tends to produce more diffuse attention beyond the lesion. This indicates that the image domain used during pretraining influences how the CNN exploits visual information. However, these differences in visual explanations are not reflected in predictive performance, as the ImageNet-1k–pretrained EfficientNetB0 remains the strongest model on the external dataset.

8 Future Work

An important direction for future work concerns the re-evaluation of in-domain self-supervised learning by performing BYOL pretraining under its standard training conditions, including larger batch sizes and full passes over the complete dataset. This evaluation would provide a more reliable estimate of the potential of self-supervised learning in skin lesion imaging. Another possible direction involves examining alternative families of deep learning models, such as hybrid CNN–ViT architectures or pure transformer models. These represent a different paradigm from the convolutional architectures evaluated in this thesis.

In the context of multimodal integration, this thesis focused exclusively on intermediate and late-fusion strategies. Early fusion approaches represent another direction for additional research. These alternatives may exploit the clinical metadata more effectively than the neural fusion strategies tested in this work.

A further direction for future work concerns the interpretation of the attention patterns produced by the models. While lesion-centred saliency maps are generally more intuitive to interpret, more diffuse attention patterns may still capture clinically relevant contextual information. Future work could therefore benefit from collaboration with medical experts to better assess how such patterns should be interpreted, particularly when model attention spreads beyond the lesion region. In addition, the use of more advanced saliency and explainability methods could provide higher-resolution attribution maps. These may offer finer insight into the visual cues driving model predictions, particularly when interpreted by medical experts.

9 Conclusion

This thesis investigated the problem of skin cancer detection in the context of the ISIC 2024 challenge, examining both architectural modifications to the EfficientNetB0 backbone and the influence of different pretraining strategies on performance, robustness, and interpretability.

The experimental results showed that integrating attention mechanisms into EfficientNetB0 did not consistently improve performance. CBAM reduced the pAUC, while Triplet Attention yielded only modest gains when training from scratch, and did not provide a clear advantage once pretrained initialization was used. However, Triplet Attention remained the most favourable option among the tested attention mechanisms by offering competitive performance with fewer parameters. These findings clarify how the attention mechanisms evaluated in this study affect the performance of EfficientNetB0 under different training conditions.

The comparison between supervised ImageNet-1k initialization and in-domain BYOL pretraining shows that both strategies lead to comparable predictive performance under the experimental conditions considered in this study. Despite more restrictive computational constraints, BYOL achieves performance comparable to ImageNet pretraining, indicating that in-domain self-supervised learning can serve as an alternative when large-scale labelled data are limited. These results further indicate that pretraining choices influence the visual representations and attention patterns learned by the model, even when performance remains comparable.

Several limitations must be acknowledged. The experiments were conducted on subsampled versions of the ISIC dataset due to the extreme class imbalance and the computational cost. This choice also affects how BYOL's contribution should be interpreted. The observed improvement over training from scratch was measured only on the reduced dataset. No comparison was made with a scratch baseline trained on the full dataset. As a result, the performance improvement should be considered valid only within the limited data setting used in this study.

Despite these limitations, this thesis provides a detailed assessment of architectural modifications, pretraining strategies, and multimodal integration for skin cancer detection. The findings contribute to a clearer understanding of how modern deep learning methods behave in challenging medical imaging settings and offer a foundation for future research.

Bibliography

[1] OpenAI, "Chatgpt," https://chat.openai.com, 2025.

[2] Z. Apalla, D. Nashan, R. B. Weller, and X. Castellsagué, "Skin cancer: Epidemiology, disease burden, pathophysiology, diagnosis, and therapeutic approaches," *Dermatol Ther (Heidelb)*, vol. 7, no. Suppl 1, pp. 5–19, 2017.

[3] Wikipedia contributors, "Skin cancer — Wikipedia, the free encyclopedia," 2026, [Online; accessed 1-January-2026]. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Skin_cancer&oldid=1330540449

[4] M. E. Celebi, H. A. Kingravi, B. Uddin, H. Iyatomi, Y. A. Aslandogan, W. V. Stoecker, and R. H. Moss, "A methodological approach to the classification of dermoscopy images," *Computerized Medical Imaging and Graphics*, vol. 31, no. 6, pp. 362–373, Sep. 2007.

[5] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, pp. 115–118, 2017.

[6] B. Cassidy, C. Kendrick, A. Brodzicki, J. Jaworek-Korjakowska, and M. H. Yap, "Analysis of the isic image datasets: Usage, benchmarks and recommendations," *Medical Image Analysis*, vol. 75, p. 102305, 2022.

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," *CoRR*, vol. abs/2010.11929, 2020. [Online]. Available: https://arxiv.org/abs/2010.11929

[8] A. Steiner, A. Kolesnikov, X. Zhai, R. Wightman, J. Uszkoreit, and L. Beyer, "How to train your vit? data, augmentation, and regularization in vision transformers," 2022. [Online]. Available: https://arxiv.org/abs/2106.10270

[9] G. H. Dagnaw, M. E. Mouhtadi, and M. Mustapha, "Skin cancer classification using vision transformers and explainable artificial intelligence," *Journal of Medical Artificial Intelligence*, vol. 7, no. 0, 2024. [Online]. Available: https://jmai.amegroups.org/article/view/8962

[10] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of big data*, vol. 6, no. 1, pp. 1–48, 2019.

[11] A. Mahbod, P. Tschandl, G. Langs, R. Ecker, and I. Ellinger, "The effects of skin lesion segmentation on the performance of dermatoscopic image classification," *Computer Methods and Programs in Biomedicine*, vol. 197, p. 105725, Dec. 2020. [Online]. Available: http://dx.doi.org/10.1016/j.cmpb.2020.105725

[12] A. Quishpe-Usca, S. Cuenca-Dominguez, A. Arias-Viñansaca, K. Bosmediano-Angos, F. Villalba-Meneses, L. Ramírez-Cando, A. Tirado-Espín, C. Cadena-Morejón, D. Almeida-Galárraga, and C. Guevara, "The effect of hair removal and filtering on melanoma detection: a comparative deep learning study with alexnet cnn," *PeerJ Computer Science*, vol. 10, p. e1953, 2024.

[13] S.-C. Huang, A. Pareek, S. Seyyedi, I. Banerjee, and M. Lungren, "Fusion of medical imaging and electronic health records using deep learning: a systematic review and implementation guidelines," *npj Digital Medicine*, vol. 3, 12 2020.

[14] N. R. Kurtansky, B. M. D'Alessandro, M. C. Gillis, B. Betz-Stablein, S. E. Cerminara, R. Garcia, M. A. Girundi, E. V. Goessinger, P. Gottfrois, P. Guitera *et al.*, "The slice-3d dataset: 400,000 skin lesion image crops extracted from 3d tbp for skin cancer detection," *Scientific Data*, vol. 11, no. 1, p. 884, 2024.

[15] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning*, 2019.

[16] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4510–4520.

[17] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-excitation networks," 2019. [Online]. Available: https://arxiv.org/abs/1709.01507

[18] F. Sultonov, J.-H. Park, S. Yun, D.-W. Lim, and J.-M. Kang, "Mixer u-net: An improved automatic road extraction from uav imagery," *Applied Sciences*, vol. 12, p. 1953, 02 2022.

[19] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," 2018. [Online]. Available: https://arxiv.org/abs/1807.06521

[20] D. Misra, T. Nalamada, A. U. Arasanipalai, and Q. Hou, "Rotate to attend: Convolutional triplet attention module," 2020. [Online]. Available: https://arxiv.org/abs/2010.03045

[21] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar, B. Piot, K. Kavukcuoglu, R. Munos, and M. Valko, "Bootstrap your own latent: A new approach to self-supervised learning," 2020. [Online]. Available: https://arxiv.org/abs/2006.07733

[22] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," 2020. [Online]. Available: https://arxiv.org/abs/2002.05709

[23] International Skin Imaging Collaboration, "Isic archive," https://www.isic-archive.com/, accessed: 2025-08-01.

[24] Q. Ha, B. Liu, and F. Liu, "Identifying melanoma images using efficientnet ensemble: Winning solution to the siim-isic melanoma classification challenge," 2020. [Online]. Available: https://arxiv.org/abs/2010.05351

[25] D. N. A. Ningrum, S.-P. Yuan, W.-M. Kung, C.-C. Wu, I.-S. Tzeng, C.-Y. Huang, J. Y.-C. Li, and Y.-C. Wang, "Deep learning classifier with patient's metadata of dermoscopic images in malignant melanoma detection," *Journal of multidisciplinary healthcare*, pp. 877–885, 2021.

[26] C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz, "A comparative analysis of gradient boosting algorithms," *Artificial Intelligence Review*, vol. 54, no. 3, pp. 1937–1967, 2021.

[27] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," *International Journal of Computer Vision*, vol. 128, no. 2, p. 336–359, Oct. 2019. [Online]. Available: http://dx.doi.org/10.1007/s11263-019-01228-7

[28] A. Yilmaz, S. P. Yasar, G. Gencoglan, and B. Temelkuran, "Derm12345: A large, multisource dermatoscopic skin lesion dataset with 38 subclasses," 2024. [Online]. Available: https://arxiv.org/abs/2406.07426

[29] A. Bissoto, M. Fornaciali, E. Valle, and S. Avila, "(de) constructing bias on skin lesion datasets," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**