Neural Network-Based Classification of Skin Lesions: Differentiating Malignant and Benign Lesions

Morgan Burch[1], Karam Khan[2], Brian Ortiz[3], Dhruvin Patel[4], Tyler Rosario[5], Manel Sadok[6], and Abdul Rahman Younis[7]

[1–7]COMP 379, Loyola University of Chicago, supervised by Dr. Dligach

Abstract

Skin cancer is the most common form of cancer in the United States, with 1 in 5 Americans developing it during their lifetime. Despite this prevalence, early detection significantly improves survival rates. In this paper, we present a convolutional neural network (CNN)-based approach to differentiate malignant and benign lesions using 3D Total Body Photography (TBP) images. The data set was pre-processed to address class imbalance, and advanced augmentation techniques were applied. The methodology leveraged EfficientNetV2 to achieve a classification accuracy of 91% and an F1 score of 66%. Challenges, including data imbalance, and limitations of the approach are discussed.

Contents

1 Introduction

Skin cancer is the most common form of cancer in the United States, with 1 in 5 Americans developing it in their lifetime (**cdc**). Early detection significantly improves survival rates, highlighting the importance of automated machine learning algorithms for diagnosis. In this project, we develop a convolutional neural network (CNN) to classify malignant and benign skin lesions, emphasizing its utility for underserved communities.

2 Dataset Description

The dataset consists of 3D Total Body Photography (TBP) images of lesions cropped to 15x15mm. Collected across 9 institutions on 3 continents between 2015 and 2024, the dataset includes approximately 500,000 images with a 99:1 ratio of benign to cancerous lesions. For efficient training, we used a subset of 4,400 images with a 10:1 benign-to-cancerous ratio.

3 Baseline Approach

The dataset had over 400,000 images with less than 0.1% being cancerous. To not bias the model, we pared the overall dataset to just over 4,400 images, improving the ratio of benign to cancerous lesions to 10:1. We used a pretrained image classifier from Kaggle and added that to our model. To improve model robustness and prevent over- and underfitting, we employed k-fold cross-validation.

To establish a foundation for the ISIC (International Skin Imaging Collaboration) 2024 challenge, a baseline approach was implemented. The goal was to set up a simple yet functional pipeline for image classification while maintaining computational efficiency. The key steps involved in this approach are:

Dataset Preparation

The training metadata was processed to create folds using the GroupKFold method. Patients were grouped to ensure no overlap between training and validation sets across folds. For baseline training, only a fraction (1%) of negative cases was retained while keeping all positive cases to address class imbalance.

Image Augmentation and Preprocessing

Minimal augmentations were applied using Albumentations to ensure data consistency. Images were resized to 224x224 resolution, normalized to match ImageNet statistics, and converted into tensors.

Model Setup

The baseline used a pretrained EfficientNetV2 (B1 variant) without fine-tuning its base parameters (freeze_base_model=True). A custom classifier layer was added to map the feature space to two classes (positive and negative).

Training Configuration

The baseline was optimized for a single epoch to quickly establish results. A small subset of data (50,000 records) was used for efficiency. Training focused on frozen base layers to isolate classifier performance.

Performance Evaluation

The evaluation metric was primarily the positive-to-negative ratio and initial Hamming Loss on a held-out validation set. This baseline served as a starting point for developing a more sophisticated approach, enabling rapid iteration and debugging while preserving GPU quota.

4 Methodology

Building upon the baseline, the full methodology expanded both data utilization and model complexity to optimize classification performance. The following enhancements were made:

Data Utilization and Balancing

The full dataset, including all positive cases and a controlled sample of 1% negatives, was used to ensure a comprehensive representation of the data distribution. A 5-fold cross-validation strategy (GroupKFold) was employed to evaluate generalizability and robustness.

Advanced Augmentation

A robust augmentation pipeline using Albumentations was applied to increase variability and improve model generalization. This included:

- Random rotations and flips.
- Brightness and contrast adjustments.
- Resizing to 224x224 followed by normalization using ImageNet statistics.

Model Architecture and Training

- **Model**: A pretrained EfficientNetV2 (B1 variant) from timm was used. Unlike the baseline, all model parameters were trainable (freeze_base_model=False), enabling end-to-end optimization.
- **Loss Function**: Binary Cross-Entropy Loss for the binary classification task.
- **Optimizer**: Adam optimizer with a learning rate scheduler for adaptive learning.
- **Training**: Multiple epochs to ensure convergence, leveraging the full dataset split into training and validation folds.

Evaluation and Metrics

The model was evaluated using Hamming Loss, F1-Score, and AUC-ROC on validation folds. Predictions were binarized to rigorously assess classification accuracy. This evaluation approach allowed us to measure the trade-offs between precision and recall across different classes.

Augmentation effectiveness was validated by visualizing multiple augmented versions of positive samples, confirming the diversity in the training data. Furthermore, predictions on the test set were generated using an ensemble of models trained across all folds to ensure robust performance.

Table 1: 1 Epoch Metrics (Normalized Data Set)

Class	Precision (%)	Recall (%)	F1-Score (%)
Non-Cancerous	93	97	95
Cancerous	50	30	38

Table 1 shows the evaluation metrics after 1 epoch using the normalized dataset, demonstrating early-stage performance.

Table 2: 20 Epoch Metrics (Full Data Set)

Class	Precision (%)	Recall (%)	F1-Score (%)
Non-Cancerous	100	100	100
Cancerous	0	0	0

Table 3: 50 Epoch Metrics (Full Data Set)

Class	Precision (%)	Recall (%)	F1-Score (%)
Non-Cancerous	100	100	100
Cancerous	0.31	0.03	0.05

Tables 2 and 3 highlight the progression of model performance using the full dataset over 20 and 50 epochs, respectively.

Table 4: 50 Epoch Metrics (Full Normalized Data Set)

Class	Precision (%)	Recall (%)	F1-Score (%)
Non-Cancerous	93	98	95
Cancerous	56	20	30

Table 5: 100 Epoch Metrics (Full Normalized Data Set)

Class	Precision (%)	Recall (%)	F1-Score (%)
Non-Cancerous	93	98	95
Cancerous	54	21	31

Tables 4 and 5 showcase the model's improved performance after normalization, with significant gains in recall and F1-score for the cancerous class.

Table 6: Out-of-Fold Predictions on Training Data (20 Epoch Full Data Set)

isic_id	Target	Original Target	Fold	OOF Prediction	Model Filename
ISIC_0015670	0	0	3	0.045678	model_fold_3_epoch_20.pth
ISIC_0015845	0	0	1	0.312456	model_fold_1_epoch_20.pth
ISIC_0015864	0	0	4	0.002345	model_fold_4_epoch_20.pth
ISIC_0015902	0	0	1	0.000678	model_fold_1_epoch_20.pth
ISIC_0024200	0	0	0	0.041256	model_fold_0_epoch_20.pth

Table 6 provides detailed out-of-fold predictions, illustrating how the model performed on unseen validation folds during training.

Table 7: Predictions on Test Data Set (All Epochs)

isic_id	Target	1E	20E	50E	100E
ISIC_0015657	0	0.057628	0.508706	0.491470	0.500368
ISIC_0015729	0	0.003957	0.486076	0.513111	0.502515
ISIC_0015740	0	0.097699	0.488023	0.487390	0.501127

Finally, Table 7 summarizes predictions on the test set across all epochs, highlighting performance improvements over time.

5 Model Comparison

To compare, a K-Nearest Neighbors (KNN) model was implemented with K values of {5, 10, 15, 20}. The best performance for malignant lesions was observed at K=5.

Table 8: KNN F1-Scores by K Values

K Value	Class 1 F1-Score
5	0.2418
10	0.2222
15	0.1839
20	0.1628

As the output shows, the larger the K value, the worse the performance of the model with respect to Class 1 (the cancerous class). This is likely due to the data imbalance: as the K value increases, it considers more neighbors which likely represent more samples of the majority class (Class 0), leading to more Class 0 classifications. The best performance of the model with respect to the cancerous class was with a K value of 5.

At first glance, it seems the KNN model performed well, with the macro-average being 0.96. However, when looking at the performance with regard to the cancerous (and most important) class, we can see that the KNN model performed much worse than the more complex CNN model. The recall of the CNN model was double that of the KNN, and the overall F1 score was also much higher.

Table 9: Classification Metrics for Best KNN Model (K=5)

Metric	Class 0	Class 1	Accuracy	Macro Avg
Precision	1.00	0.92	1.00	0.96
Recall	1.00	0.14		0.57
F1-Score	1.00	0.24		0.62
Support	80133	79	80212	80212

In a real-world scenario for cancer detection, missing a cancerous case (a false negative) would be the most severe outcome. The CNN model's higher recall means it is far less likely to miss cancerous samples, making it a more reliable choice for this task.

6 Discussion

Our model achieved an overall accuracy of 91%, precision of 71%, recall of 64%, and an F1 score of 66%. While the accuracy is impressive, the rest of the metrics are middling. When we break the metrics down by class, however, we can see where the model truly falls short.

Class 0 (benign) had a precision of 93%, a recall of 97%, and an F1 score of 95%. The model reliably labeled benign lesions without too much issue. However, the model struggled to identify Class 1 (malignant) lesions. Class 1 had a precision of 50%, recall of 30%, and an F1 score of 38%.

Several factors contributed to these results. While there were thousands of images of benign lesions, the number of malignant lesions was just shy of 400. This data imbalance

was something we were aware of and actively sought to combat by reducing the total number of benign lesions in preprocessing. Still, this discrepancy may have biased the dataset in a way that is difficult to stop.

Another concern is that reducing the dataset size may have decreased the model's performance due to a lack of training samples. Image classifiers work best when given very large datasets. Our small sample may have had a negative impact.

7 Conclusion

This project was an eye-opening experience working on convolutional neural networks. This real-world competition dataset allowed us to apply machine learning principles and techniques in a realistic manner, further reinforcing course learning objectives.

In the future, if we were to perform this experiment again with enough time, we would train the model with a full balanced dataset (all 500,000 training samples and an equal number of testing samples) on multiple GPUs. Professor Dligach gave our group access to two GPU servers to train our model. However, when we tried accessing them, both servers were occupied.

Appendix

Team Member	Contributions
Morgan Burch	KNN Model and Analysis
Karam Khan	Data cleanup, normalization, literature review, and compilation
Brian Ortiz	Model testing, report, PowerPoint
Dhruvin Patel	Model creation and testing
Tyler Rosario	Helped write report, reviewed model and PowerPoint
Manel Sadok	PowerPoint, model review, writing report
Abdul Rahman Younis	PowerPoint, model review, writing report