Individual Final Report

Name: Rasika Nilatkar
Project Title: Classification of Skin Lesion Melanoma Detection
Course: Deep Learning
Group: Group 5

1. Introduction

The purpose of this project is to build a deep learning model to classify skin lesions into nine categories based on dermoscopic images. Early diagnosis of skin cancer is critical, and our project applies convolutional neural networks (CNNs), specifically ResNet18 in this project, to recognize melanoma alongside other diseases. HAM10000 (ISIC 2019) dataset containing over 25,000 images has been employed. This report reflects on my personal work, including data augmentation, configuration of the model, and development of the Streamlit-based interactive demo.

2. Work Description

Transforms and Data Augmentation -

To promote better generalization by the model and avoid overfitting, I built a sequence of transforms:

RandomResizedCrop to replicate changing sizes and zoom levels for images.
RandomHorizontalFlip and VerticalFlip for keeping positional versatility.
RandomRotation and ColorJitter for coping with differences in lighting and pose.

Normalization in terms of ImageNet statistics for adapting to the pre-trained assumptions of ResNet18.

Validation set was the only one undergoing resizing and normalization for ensuring assessment consistency.

Model Setup -

I applied the architecture with ResNet18 as a base and a transformed head:

Dropped the default classification head (nn.Identity())

Inserted a dense layer of 256 nodes with ReLU, Dropout(0.5), and a final Linear layer of 9 output nodes (for 9 classes)

Used a sigmoid activation to support multi-label classification

Loss function used: BCELossOptimizer: Adam with lr=1e-4Learning rate scheduler: ReduceLROnPlateau

3. Detailed Contribution

Data Preprocessing: Integrated data loading, label conversion to PyTorch tensors.

Transform Pipeline: Included higher-level image augmentations in train_transform and normalized images with ImageNet values.

Model Implementation: Implemented the model architecture using nn.Sequential layers.

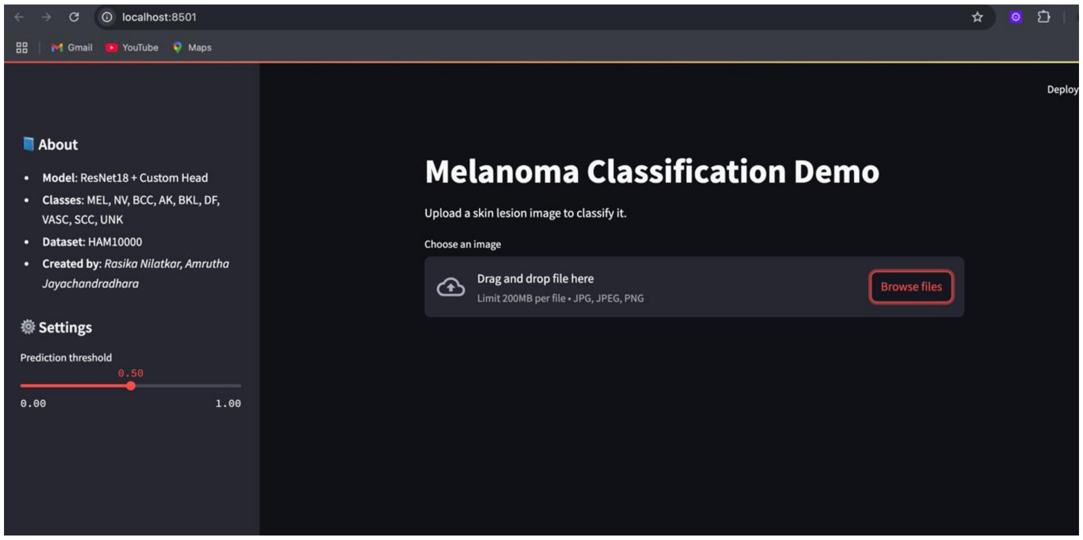
Streamlit App:

Developed an interactive frontend using Streamlit

Included upload feature, image preprocessing, model inference, and result visualization

Included entropy-based filtering and confidence thresholds to remove false positives on random images

Sample Outputs:



About

- **Model:** ResNet18 + Custom Head
- **Classes:** MEL, NV, BCC, AK, BKL, DF, VASC, SCC, UNK
- **Dataset:** HAM10000
- **Created by:** *Rasika Nilatkar, Amrutha Jayachandradhara*

Settings

Prediction threshold 0.50 (0.00 – 1.00)

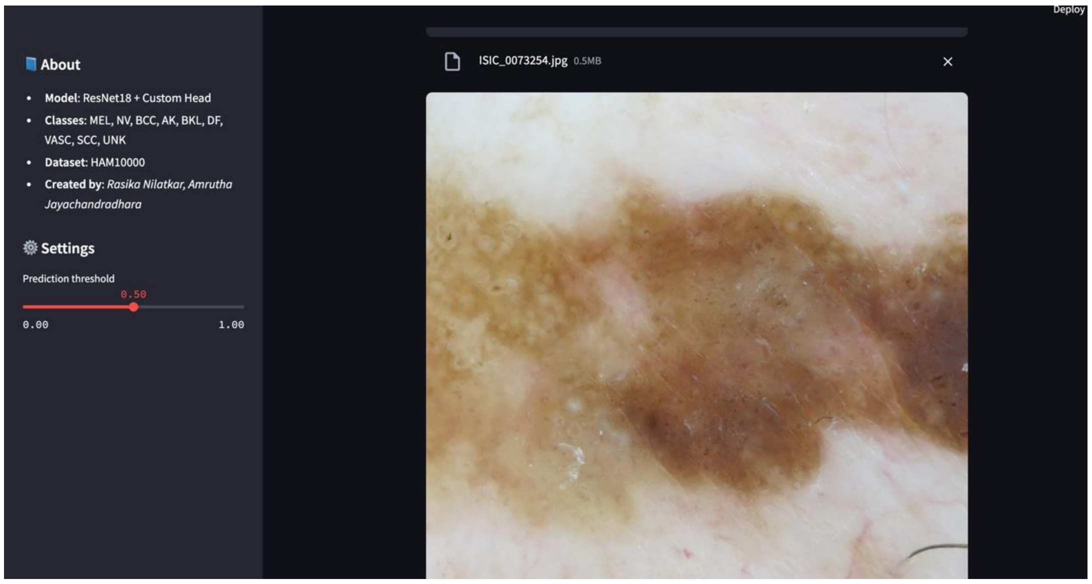
Melanoma Classification Demo

Upload a skin lesion image to classify it.

Choose an image

Drag and drop file here

Limit 200MB per file • JPG, JPEG, PNG


Deploy

About

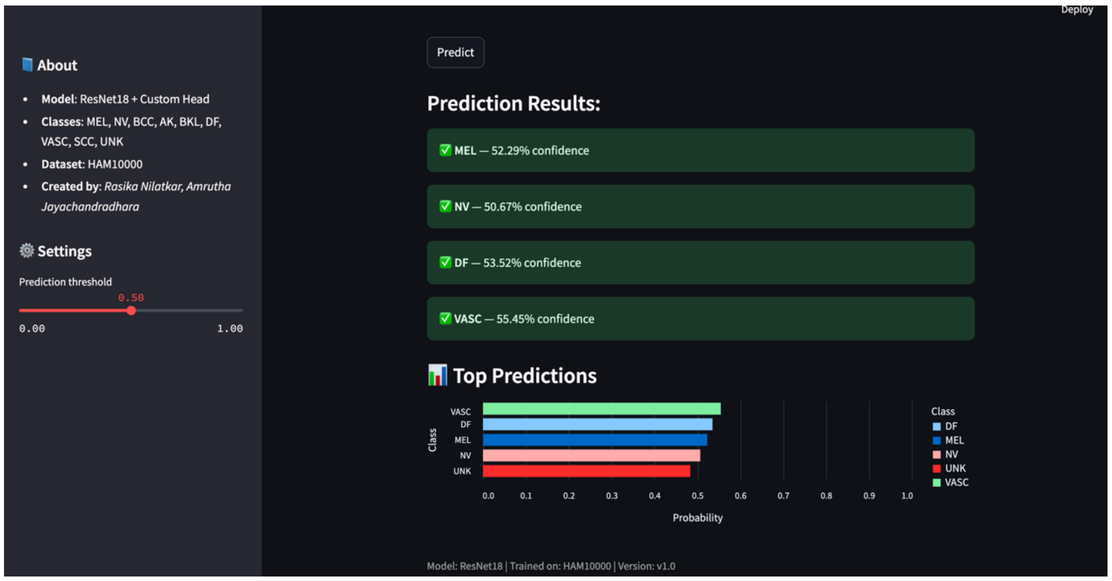
- **Model:** ResNet18 + Custom Head
- **Classes:** MEL, NV, BCC, AK, BKL, DF, VASC, SCC, UNK
- **Dataset:** HAM10000
- **Created by:** *Rasika Nilatkar, Amrutha Jayachandradhara*

Settings

Prediction threshold 0.50 (0.00 – 1.00)

ISIC_0073254.jpg 0.5MB

The Streamlit demo also helps users interpret the model's top 5 predictions through a bar chart, enhancing transparency.


Deploy

About

- **Model:** ResNet18 + Custom Head
- **Classes:** MEL, NV, BCC, AK, BKL, DF, VASC, SCC, UNK
- **Dataset:** HAM10000
- **Created by:** *Rasika Nilatkar, Amrutha Jayachandradhara*

Settings

Prediction threshold 0.50 (0.00 – 1.00)

Predict

Prediction Results:

✅ **MEL** — 52.29% confidence

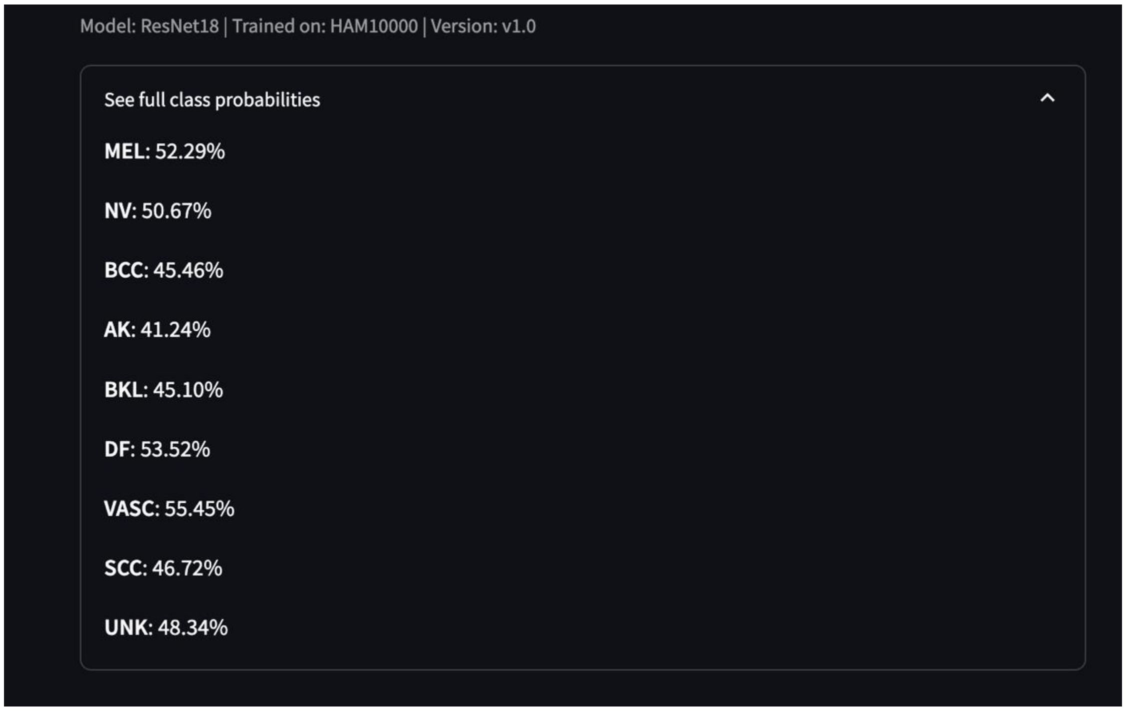
✅ **NV** — 50.67% confidence

✅ **DF** — 53.52% confidence

✅ **VASC** — 55.45% confidence

Top Predictions

Model: ResNet18 | Trained on: HAM10000 | Version: v1.0


Model: ResNet18 | Trained on: HAM10000 | Version: v1.0

See full class probabilities

MEL: 52.29%

NV: 50.67%

BCC: 45.46%

AK: 41.24%

BKL: 45.10%

DF: 53.52%

VASC: 55.45%

SCC: 46.72%

UNK: 48.34%

4. Results

Our model verified at ~85% accuracy.

Major observations:

Training Loss Curve showed consistent convergence

Streamlit Output: Correctly classifies known lesion types with confidence and discards non-lesion images based on heuristics

5. Summary and Conclusion

I assisted in the preprocessing, architecture design, and frontend integration of the project.

Takeaways:

Data augmentation significantly enhances model robustness
Custom heads rather than pretrained models allow flexibility
Streamlit allows simple and quick deployment of ML demos

Future improvements can be:

Training on full 25k dataset
Adding Grad-CAM for visual explanation
Testing on real clinical data

6. Code Reuse Calculation

Total lines reused code: 60
Lines modified: 20
New lines written by me: 40
Percentage of code reuse: (60-20)/(60+40) x 100 = 40%

7. References

1. https://ieeexplore.ieee.org/document/10094404
2. https://pytorch.org/vision/stable/models.html
3. https://www.kaggle.com/datasets/kmader/skin-cancer-mnist-ham10000
4. https://docs.streamlit.io