

Plagiarism Detection Report by SmallSEOTOOLS



● Plagiarism	0%	● Partial Match	0%
● Exact Match	0%	● Unique	100%

Scan details

Total Words	Total Characters	Plagiarized Sentences	Unique Sentences
1000	7301	0	45 (100%)

#1 100% Unique

Multi-Label Classification of Skin Diseases Using Deep Learning

Description

This project tackles the problem of automated skin lesion classification using deep learning. The goal is to build and evaluate a model that can accurately classify an image of a skin lesion into one of nine distinct disease categories:

Actinic keratosis

Atopic Dermatitis

Benign keratosis

Dermatofibroma

Melanocytic nevus

Melanoma

Squamous cell carcinoma

Tinea Ringworm Candidiasis

Vascular lesion

Automating this diagnosis is important as it can assist dermatologists, reduce diagnostic time, and provide an accessible, low-cost preliminary screening tool, which is especially critical for early detection of malignant conditions like Melanoma.

This project explores three different Convolutional Neural Network (CNN) architectures, compares their performance, and identifies key challenges such as model overfitting.

Dataset Source

Source: The data used is a subset of the ISIC (International Skin Imaging Collaboration) 2019 Training Input dataset. The notebook specifically uses a small, pre-split version of the data, referred to as Split_smol.

You can download the dataset from <https://www.kaggle.com/datasets/nodoubtome/skin-cancer9-classesisic>

Data Size: The exact number of images is not specified, but the training logs for the main model show 18 batches, and the test logs show 6 batches, indicating a relatively small dataset.

Preprocessing: To ensure uniformity and compatibility with the deep learning models, all images in the dataset underwent the following preprocessing steps:

Images were converted to the 'RGB' color format.

All images were resized to a uniform 240x240 pixels.

Pixel values were normalized to a range of [0, 1] by dividing by 255.0.

The string labels (e.g., 'Melanoma') were mapped to integer indices (e.g., 5) and then one-hot encoded using keras.utils.to_categorical for use with the categorical_crossentropy loss function.

Methods

The project compares three different deep learning approaches to identify the most effective one for this task. All models were compiled with the Adam optimizer, categorical_crossentropy loss, and tracked accuracy as the primary metric.

1. DenseNet121 (Transfer Learning)

Approach: This model uses the DenseNet121 architecture, pre-trained on the ImageNet dataset, as a feature extraction base. Transfer learning is a powerful technique that leverages knowledge from a massive dataset (ImageNet) for a new, specific task (skin lesion classification).

Architecture:

The pre-trained DenseNet121 base (with top layers excluded).

A GlobalAveragePooling2D layer to reduce spatial dimensions.

A Dropout(0.5) layer to prevent overfitting.

A final Dense layer with 9 outputs and a softmax activation function for multi-class classification.

Rationale: DenseNets are known for their parameter efficiency and strong feature propagation, making them a popular choice for image classification.

2. MobileNet (Transfer Learning)

Approach: This model uses the MobileNet architecture, also pre-trained on ImageNet.

Architecture: The structure is identical to the DenseNet model, with the DenseNet121 base replaced by the MobileNet base.

Rationale: MobileNet is a lightweight, computationally efficient architecture. It serves as a good alternative to see if a less complex model can achieve comparable results, which would be ideal for deployment on mobile or low-power devices.

3. Custom CNN (Baseline)

Approach: A simple Convolutional Neural Network was built from scratch to serve as a baseline.

Architecture: The model consists of several Conv2D and MaxPooling2D layers, followed by Dropout, a Flatten layer, and two Dense layers, with the final layer having 9 outputs and softmax activation.

Rationale: This baseline helps to quantify the performance gain provided by the more complex transfer learning models.

Steps to Run the Code

Clone the repository:

```
git clone https://github.com/PravegChikte/ML-Project.git
```

```
cd ML-Project
```

Install Dependencies:

```
pip install tensorflow keras numpy pandas matplotlib seaborn scikit-learn scikit-plot opencv-python pillow streamlit
```

Get the Data:

Download the ISIC 2019 dataset.

Organize the data into the structure expected by the notebook:

```
.../Split_smol/
```

```
train/
```

```
Actinic keratosis/
```

```
images/
```

Atopic Dermatitis/

(images...)

... (and 7 other classes)

test/

Actinic keratosis/

(images...)

... (and 8 other classes)

Update the IMG_SAVE_PATH and IMG_SAVE_PATH_TESTING variables in the ISIC Skin disease v5.3.ipynb notebook to point to your local train and test directories.

Run the Notebook:

Open and run the ISIC Skin disease v5.3.ipynb notebook in Jupyter Lab or Jupyter Notebook to train the models and see the evaluation.

Run the Streamlit Web App (Optional):

First, save your trained model from the notebook:

```
# Add this to your notebook after training  
  
model.save('skin_disease_model.h5')
```

Create an app.py file (as provided in the previous answer) in the same directory.

Run the app from your terminal:

```
streamlit run app.py
```

Experiments/Results Summary

The three models were trained for 5 epochs and evaluated on the held-out test set.

Model Performance Comparison:

Model Test Loss Test Accuracy

MobileNet ~1.00 ~1.11 110%

DenseNet121 ~0.58 ~35.36%

Custom CNN ~2.21 ~24.31%

Based on the notebook's test evaluation cells.

Overfitting Analysis: The training and validation plots for all models show a clear and significant overfitting problem. The training accuracy climbs rapidly (e.g., ~74% for DenseNet) while the validation accuracy remains low and flat (e.g., ~20% for DenseNet). This indicates the models are memorizing the small training dataset and are not generalizing to new, unseen images.

Accuracy and Loss curves for the DenseNet121 model, showing a large gap between training (blue/orange) and validation (green/red) metrics.

Prediction Analysis: A confusion matrix was generated for the DenseNet model's predictions on the test set. This matrix helps visualize which classes are most often confused (e.g., if 'Melanoma' is frequently misclassified as 'Benign keratosis').

Conclusion

This project successfully implemented and compared three deep learning models for skin lesion classification.

Key Results: The lightweight MobileNet model achieved the highest test accuracy (41.44%) after 5 epochs, outperforming the deeper DenseNet121 (35.36%) and the baseline custom CNN (24.31%).

Key Learning: The most critical takeaway is that all models severely overfit the training data. The small dataset size (Split_smol) is insufficient for these large models to learn generalizable features without memorization.

Future work should focus entirely on combating overfitting by:

Using a much larger dataset.

Implementing aggressive data augmentation (flips, rotations, zooms, color shifts).

Increasing regularization (e.g., increasing Dropout rates, adding L2 regularization).

Experimenting with fine-tuning strategies (e.g., unfreezing fewer layers of the pre-trained base).

Using techniques like Early Stopping to halt training before the model overfits.

References

Dataset: The International Skin Imaging Collaboration (ISIC) 2019. (<https://challenge.isic-archive.com/2019/>)

DenseNet: Densely Connected Convolutional Networks.