Comparative Analysis of Neural Network Architectures in Skin Lesion Classification

Vinay Jayanti

Received August 08, 2024
Accepted November 30, 2024
Electronic access December 15, 2024

Skin cancer is the most common type of cancer in the US, making efficient diagnosis of skin lesions key to saving lives. Artificial intelligence has proven to be a potential tool to improve the speed and accuracy of skin lesion diagnosis. This research paper analyzes the performance of several convolutional neural networks (CNNs) in skin lesion classification, namely ResNet50V2, ResNet101V2, ResNet152V2, InceptionV3, and InceptionResNetV2 architectures. More specifically, this study compares the performance of residual neural networks (ResNets) to traditional CNNs to determine the strengths and weaknesses of the architectural techniques in skin lesion classification. The models were trained on 10,940 images across fourteen classes sourced from the International Skin Imaging Collaboration 2019 Challenge dataset and the Mpox Skin Lesion Dataset v2.0. Metrics including loss trends, accuracy trends, training duration, precision, recall, and F1 scores were assessed. The ResNet101V2 architecture achieved the highest macro F1 score of 0.78 while the InceptionV3 architecture had the lowest macro F1 score of 0.70. The hybrid InceptionResNetV2 architecture had the fastest training time of 12100.1 seconds, a large improvement over the InceptionV3 architecture training time of 13909.6 seconds. The Inception-based models, however, did show fewer signs of overfitting compared to the ResNet models. The results indicate that hybrid models with residual connections have potential advantages in efficiency and accuracy compared to traditional CNNs in skin lesion classification.

Keywords: Convolutional neural network, residual neural network, skin lesion classification, transfer learning, hybrid model

Introduction

Background

Skin cancer is the most common type of cancer in the US, with over 3.3 million Americans diagnosed every single year[1]. According to the American Academy of Dermatology Association, approximately one in five Americans will develop skin cancer in their lifetime[2]. Early diagnosis of skin diseases directly impacts patient treatment and outcomes. Skin cancers become especially dangerous when allowed to metastasize to other regions of the body. For example, patients diagnosed with melanomas that metastasize to distant organs such as the lungs and liver have a five-year survival rate of only 35%[3]. Additionally, poor diagnostic accuracy of melanomas adds an estimated $673 million in disease management costs[4]. Accurate and early diagnoses allow for prompt and efficient intervention, allowing patients to return to their lives. Automated image classification tools developed with machine learning can improve diagnostic accuracy, reduce human error, and save time for dermatologists. Discerning between distinct types of skin lesions, ranging from carcinomas to viral infections, is important in improving patients' quality of life by increasing confidence and control over their health.

Artificial intelligence (AI) can play a significant role in medical diagnostics. Deep learning, a subset of AI, uses neural networks to find patterns in large amounts of data. For example, convolutional neural networks (CNNs) are widely used in processing images, also known as computer vision. CNN models have the potential to exceed human performance in finding patterns in images, sparking their implementation in analyzing CT scans, MRIs, and many more types of medical imaging.

Currently, dermatological condition classification requires the expertise of dermatologists, which leads to subjective interpretation. Manual categorization is time-consuming and leaves the door open to human error. While the implementation of machine learning tools in the medical field is not likely to fully replace human physicians, deep learning models have enormous potential as a tool to increase accuracy, efficiency, and confidence for both the patient and the provider.

Literature Review

Several prior studies analyzed deep-learning tools for skin lesion classification. Esteva et al. (2017) trained an InceptionV3 model loaded with ImageNet weights across more than seven hundred classes of benign and malignant skin diseases. They found that their Deep CNN was able to match the performance of twenty-

one dermatologists in keratinocyte, carcinoma, and melanoma classification[5]. Stofa et al. (2021) conducted a review of the performance of twelve CNN architectures in skin lesion classification, including ResNet, Inception, Xception, DenseNet, and EfficientNet architectures. They found the aforementioned models had excellent performance but required extensive computational resources and took a long time to optimally converge[6]. Lopez et al. (2017) explored three different training methods for a VGGNet architecture in skin lesion classification, with a focus on melanoma classification. The study found that the model with the strongest testing performance and the least overfitting included using transfer learning with ImageNet weights and unfreezing the highest layers[7]. AlSuwaidan (2023) studied the six CNN architectures of VGG16, EfficientNet, InceptionV3, MobileNet, NasNet, and ResNet50 for the classification of the three most common dermatological diseases in the Middle East. They found that the MobileNet architecture had the best performance in classifying atopic, eczema, and psoriasis[8].

Objective

This study is a comparative analysis between residual neural networks (ResNets) and traditional CNNs in classifying fourteen skin disorders, ranging from cancers to viral infections. The results of this study will provide insights to the advantages and disadvantages of ResNets and CNNs in skin lesion classification, which will contribute to future architectural development decisions when developing a model for clinical applications. This study builds on previous research by specifically comparing the techniques employed by ResNets rather than comparing a wide variety of models.

The primary usage of deep learning in medical image classification is to improve diagnostic accuracy and efficiency. Therefore, this comparative analysis will measure training duration alongside different accuracy metrics to analyze the strengths and weaknesses of each architecture. Trends across certain model architectures can help identify model architectures that have the strongest potential for clinical usage.

Scope and Limitations

This study evaluates the performance of models on the following fourteen classes: Actinic keratoses; basal cell carcinomas; benign keratosis-like lesions; chickenpox; cowpox; dermatofibromas; hand, foot, and mouth disease (HFMD); healthy; measles; melanocytic nevi; melanomas; monkeypox; squamous cell carcinomas; and vascular lesions. While this is a widespread class list, it is not all-encompassing, so the results of this study will produce general guidelines for skin lesion classification rather than a solution for all skin conditions. Additionally, because of GPU constraints, training was limited to sixty epochs.

Important Concepts

Transfer Learning

This study is centered around the usage of transfer learning. Transfer learning utilizes predefined weights and architectures to take advantage of proven techniques in areas such as feature extraction. This gives models a sort of head starts in training, especially on complex data, where it would be computationally inefficient for a model to learn basic parameters surrounding image processing. Transfer learning is particularly useful in medical image classification tasks, where image data is hard to collect due to privacy concerns, variations in imaging modality, and the need for expert annotations. Moreover, certain image classification tasks, such as this case of dermatological image classification, are highly specialized and do not have large datasets for models to learn from. The models in this study will use base models available through Keras[9], a high-level neural network application programming interface (API) designed to simplify the building and training of deep learning models. The study will compare the architectures of ResNet50V2, ResNet101V2, ResNet152V2, InceptionV3, and InceptionResNetV2. All models will also be loaded with ImageNet[10], a set of weights designed for the classification of 1000 image classes.

Residual Neural Networks Versus Traditional CNNs

Traditional CNNs are types of deep learning algorithms designed to process images. Figure 1 shows an example of a CNN. The neural network is organized into layers, with each layer serving a different purpose. Images are converted into an array of pixel values, which serve as input data. Convolutional layers apply specific filters (kernels) that run over the input data using matrix multiplication. Figure 2 shows an example of a kernel. Certain filters are specialized in various types of feature extraction, such as locating vertical or horizontal edges. Pooling layers, such as average or maximum pooling, are applied to decrease the amount of input data by outputting the average or highest values, respectively, from specific regions of the image. The convolutional and pooling layers are known as feature extraction layers[11]. The output of a layer serves as the input for the proceeding layer, and these inputs and outputs are referred to as feature maps. Ultimately, the original feature map passes through layers of convolutional and pooling filters until it reaches the fully connected (FC) or dense layers. The feature map is flattened to one dimension by multiplying its height with its width. It is also possible that its depth is multiplied if multiple feature maps were created by multiple kernels in a preceding layer. The dense layers are the final stages of neurons that classify the image. Each neuron has weight W and bias b. The output, z, of a neuron, is derived from the equation $z = Wx + b$, where x is the input. Activation functions are applied to the outputs of both convolutional and dense layers, which will be

elaborated on when discussing architectural development.

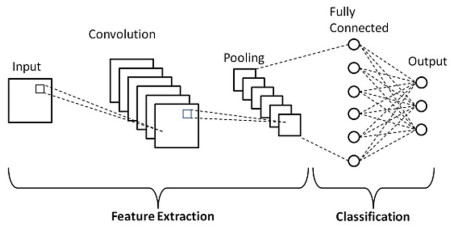

Fig. 1 Example Convolutional Neural Network [11]

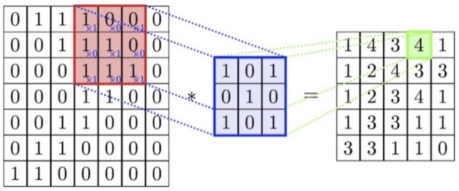

Fig. 2 Example of a kernel [12]. The input (left) is multiplied by the kernel (center) to output a feature map (right)

Residual neural networks (ResNets) are a variation of traditional CNNs that address the problems of training deep (many-layered) neural networks. In CNNs, a loss function determines the accuracy of a model's outputs in relation to the true class labels after a forward pass. Minimizing the loss is the key to higher model performance. Based on the loss function, a gradient is calculated, which determines the extent to which a model's weights are adjusted via backpropagation. In a CNN with many layers, gradients can compound during backpropagation, making the changes to weights too extreme (exploding gradient) or negligible (vanishing gradient). ResNets solve this issue by employing residual or skip connections [13]. Figure 3 compares a traditional CNN to a ResNet. The output of an early convolutional layer is added to the input of a future convolutional layer, the residual connection skipping layers in between. Instead of having the model learn how to go from the input to the output, the model learns the difference (the residual) between the input and output. This prevents exploding and vanishing gradients because the gradient can follow the residual connections during backpropagation rather than going through all the layers. The skip connections are more computationally efficient and can lead to better convergence.

Methodology Overview

The dataset used [14] was adapted from a Kaggle dataset [15] which was originally sourced from the International Skin Imaging Collaboration's 2019 Challenge dataset [16] and the Mpox Skin Lesion Dataset v2.0 [17]. Data augmentation was applied to the images to create a more robust model. The fully connected

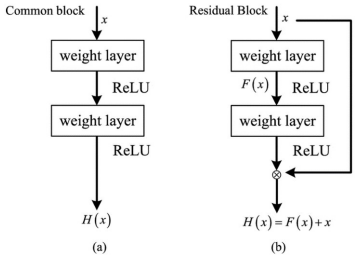

Fig. 3 Traditional CNN (left) and residual (skip) connection (right) [13]

layers (top) for each model were replaced with new dense layers. Finally, the models were trained for sixty epochs with a learning rate of 0.001, a batch size of thirty-two, a class weights balancing function, and an early stopping callback. The metrics used to evaluate each model are accuracy, F1 score, precision, recall, and training duration.

Results

Metrics

The metrics analyzed are training and validation loss over time, training and validation accuracy over time, training duration, and testing accuracy. The macro and weighted averages for F1 score, precision, and recall are also computed.

$$\text{Testing Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Number of Total Predictions}} \quad (1)$$

Testing accuracy is a straightforward metric, but not the best tool for evaluating an unbalanced dataset like the one used in this study, as the metric overlooks poor performance in minority classes [18].

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \quad (2)$$

Precision is used to calculate how often a model correctly predicts a positive class [18]. This metric is useful when evaluating false positives, which is important in the medical setting as this study intends.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \quad (3)$$

Recall is similar to precision, except it focuses on evaluating false negatives [18]. A high recall score is critical in the medical setting as a false negative can result in the missed diagnosis of a potentially dangerous disease.

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

The F1 Score is the harmonic mean of precision and recall [18], making it an effective tool in the medical setting for evaluating both false positives and false negatives.

The macro average version of a score indicates that each class was given the same weight, regardless of its number of appearances in the dataset. The weighted average accounts for class imbalance, giving larger classes more weight to the score [18].

Loss and Accuracy Plots

The figures below show training and validation loss and accuracy plots. The outputs shown below are from the author's code repositories, which are referenced in each figure.

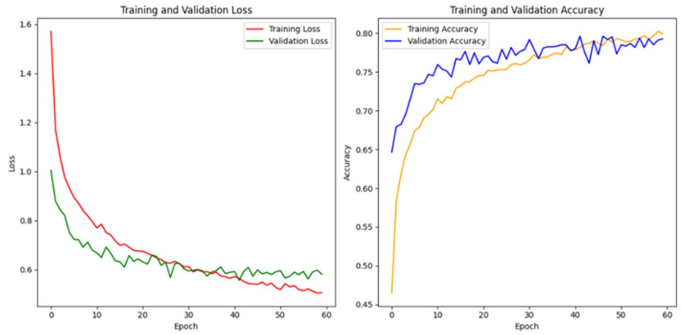

Fig. 4 ResNet50V2 Plot. Training duration of 13945.9 seconds [19]

In Figure 4, we see that the ResNet50V2 model experienced moderate overfitting, as around epoch 35 the training and validation loss lines intersected then diverged. The validation loss and accuracy plateaued after epoch 40.

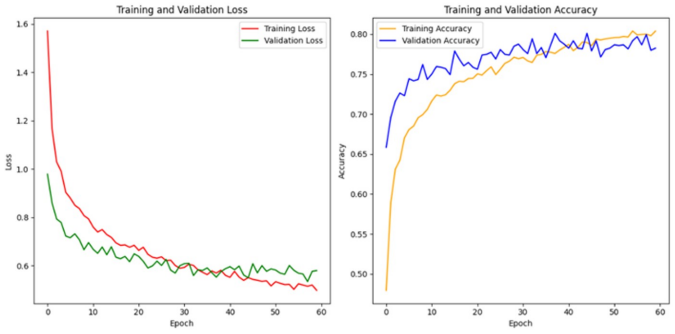

Fig. 5 ResNet101V2 Plot. Training duration of 12781.2 seconds [20]

In Figure 5, we see that the ResNet101V2 model experienced moderate overfitting, as around epoch 35 the training and validation loss lines intersected then diverged. The validation loss and accuracy plateaued after epoch 40.

In Figure 6, we see that the ResNet152V2 model experienced mild overfitting, as around epoch 35 the training and validation loss lines intersected and diverged around epoch 55. There was a consistent decrease in validation and training loss and a consistent increase in validation and training accuracy until the final 5 epochs.

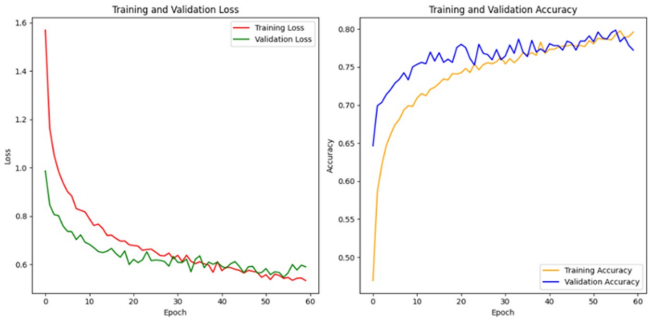

Fig. 6 ResNet152V2 Plot. Training duration of 12319.4 seconds [21]

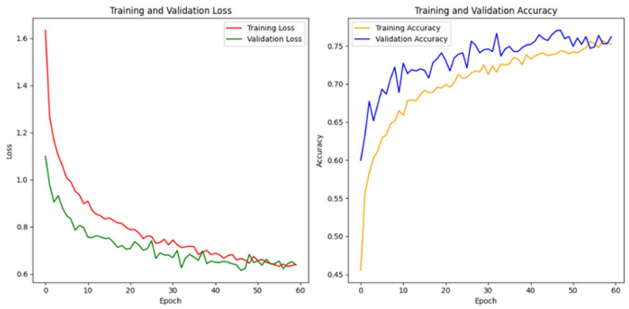

Fig. 7 InceptionV3 Plot. Training duration of 13909.6 seconds [22]

In Figure 7, we see that the InceptionV3 model experienced no noticeable overfitting, as around epoch 50 the training and validation loss lines intersected but never diverged. There was a consistent decrease in validation and training loss and a consistent increase in validation and training accuracy for the entire training duration.

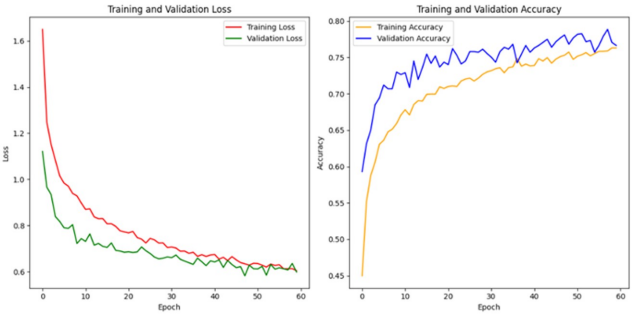

Fig. 8 InceptionResNetV2 Plot. Training duration of 12100.1 seconds [23]

In Figure 8, we see that the InceptionResNetV2 model experienced no noticeable overfitting, as around epoch 55 the training and validation loss lines intersected but never diverged. There was a consistent decrease in validation and training loss and a

consistent increase in validation and training accuracy for the entire training duration.

Classification Reports

The following figures show classification reports. The outputs shown below are from the author's code repositories, which are referenced in each figure.

	precision	recall	f1-score
Actinic keratoses	0.65	0.68	0.67
Basal cell carcinoma	0.65	0.53	0.59
Benign keratosis-like lesions	0.53	0.59	0.56
Chickenpox	0.97	0.93	0.95
Cowpox	0.99	0.97	0.98
Dermatofibroma	0.36	0.68	0.47
HFMD	0.93	0.93	0.93
Healthy	0.90	0.93	0.92
Measles	0.90	0.94	0.92
Melanocytic nevi	0.71	0.67	0.69
Melanoma	0.61	0.61	0.61
Monkeypox	0.96	0.92	0.94
Squamous cell carcinoma	0.74	0.53	0.62
Vascular lesions	0.65	0.85	0.73
accuracy			0.77
macro avg	0.75	0.77	0.75
weighted avg	0.78	0.77	0.78

Fig. 9 ResNet50V2 Report [19]

In Figure 9, we see that the ResNet50V2 model had the lowest performance in classifying dermatofibromas and benign keratosis-like lesions and the highest performance in classifying chickenpox, cowpox, and monkeypox.

Actinic keratoses	0.74	0.70	0.72
Basal cell carcinoma	0.68	0.53	0.60
Benign keratosis-like lesions	0.62	0.66	0.64
Chickenpox	0.93	0.94	0.94
Cowpox	0.97	0.95	0.96
Dermatofibroma	0.58	0.60	0.59
HFMD	0.96	0.93	0.94
Healthy	0.88	0.97	0.92
Measles	0.95	0.87	0.91
Melanocytic nevi	0.72	0.84	0.77
Melanoma	0.63	0.57	0.60
Monkeypox	0.94	0.94	0.94
Squamous cell carcinoma	0.55	0.64	0.59
Vascular lesions	0.77	0.77	0.77
accuracy			0.79
macro avg	0.78	0.78	0.78
weighted avg	0.80	0.79	0.79

Fig. 10 ResNet101V2 Report [20]

In Figure 10, we see that the ResNet101V2 model had the lowest performance in classifying dermatofibromas and squamous cell carcinomas and the highest performance in classifying chickenpox, cowpox, monkeypox, HFMD, and measles.

In Figure 11, we see that the ResNet152V2 model had the lowest performance in classifying benign keratosis-like lesions, squamous cell carcinomas, and melanomas and the highest performance in classifying cowpox, monkeypox, and measles.

In Figure 12, we see that the InceptionV3 model had the lowest performance in classifying dermatofibromas, squamous cell carcinomas, benign keratosis-like lesions and melanomas, and the highest performance in classifying cowpox.

	precision	recall	f1-score
Actinic keratoses	0.69	0.67	0.68
Basal cell carcinoma	0.67	0.50	0.57
Benign keratosis-like lesions	0.54	0.61	0.57
Chickenpox	0.91	0.96	0.94
Cowpox	0.98	0.97	0.97
Dermatofibroma	0.79	0.60	0.68
HFMD	0.92	0.96	0.94
Healthy	0.91	0.93	0.92
Measles	0.96	0.92	0.94
Melanocytic nevi	0.68	0.71	0.69
Melanoma	0.58	0.60	0.59
Monkeypox	0.96	0.90	0.93
Squamous cell carcinoma	0.55	0.62	0.58
Vascular lesions	0.63	0.73	0.68
accuracy			0.78
macro avg	0.77	0.76	0.76
weighted avg	0.78	0.78	0.78

Fig. 11 ResNet152V2 Report [21]

	precision	recall	f1-score
Actinic keratoses	0.74	0.56	0.64
Basal cell carcinoma	0.61	0.71	0.65
Benign keratosis-like lesions	0.49	0.41	0.45
Chickenpox	0.89	0.89	0.89
Cowpox	0.96	0.88	0.92
Dermatofibroma	0.26	0.48	0.34
HFMD	0.88	0.93	0.90
Healthy	0.85	0.88	0.87
Measles	0.89	0.90	0.90
Melanocytic nevi	0.63	0.66	0.65
Melanoma	0.49	0.49	0.49
Monkeypox	0.90	0.87	0.88
Squamous cell carcinoma	0.52	0.42	0.47
Vascular lesions	0.62	0.88	0.73
accuracy			0.72
macro avg	0.70	0.71	0.70
weighted avg	0.73	0.72	0.72

Fig. 12 InceptionV3 Report [22]

	precision	recall	f1-score
Actinic keratoses	0.70	0.65	0.67
Basal cell carcinoma	0.66	0.57	0.61
Benign keratosis-like lesions	0.56	0.59	0.57
Chickenpox	0.94	0.93	0.93
Cowpox	0.99	0.94	0.96
Dermatofibroma	0.39	0.60	0.48
HFMD	0.91	0.92	0.92
Healthy	0.91	0.95	0.93
Measles	0.88	0.92	0.90
Melanocytic nevi	0.70	0.66	0.68
Melanoma	0.48	0.55	0.51
Monkeypox	0.93	0.91	0.92
Squamous cell carcinoma	0.52	0.42	0.47
Vascular lesions	0.76	0.85	0.80
accuracy			0.76
macro avg	0.74	0.75	0.74
weighted avg	0.76	0.76	0.76

Fig. 13 InceptionResNetV2 Report [23]

In Figure 13, we see that the InceptionResNetV2 model had the lowest performance in classifying dermatofibromas, squamous cell carcinomas, melanomas, and benign keratosis-like lesions and the highest performance in classifying chickenpox, cowpox, and monkeypox.

Confusion Matrices

Predictions are on the X-axis and true labels are on the Y-axis. The following lesions were used in this study and the numbers below correspond to the class labels.

1. Actinic keratoses
2. Basal cell carcinoma
3. Benign keratosis-like lesions
4. Chickenpox
5. Cowpox
6. Dermatofibroma
7. Hand, Foot, and Mouth Disease (HFMD)
8. Healthy
9. Measles
10. Melanocytic nevi
11. Melanoma
12. Monkeypox
13. Squamous cell carcinoma
14. Vascular lesions

The outputs shown below are from the author's code repositories, which are referenced in each figure.

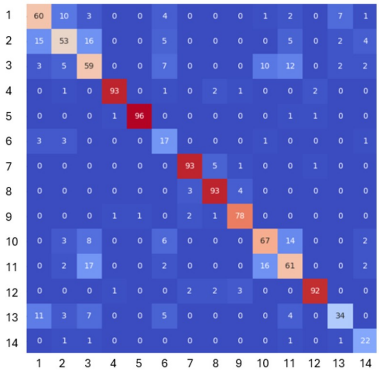

Fig. 14 ResNet50V2 Matrix [19]

In Figure 14, we see that the ResNet50V2 model most easily classified chickenpox, cowpox, HFMD, healthy skin, measles, vascular lesions, and monkeypox. The model struggled to differentiate between melanocytic nevi and melanomas (classes 10 and 11), actinic keratoses and basal cell carcinomas (classes 1 and 2), actinic keratoses and squamous cell carcinomas (classes 1 and 13), basal cell carcinomas and benign keratosis-like lesions (classes 2 and 3), and benign keratosis-like lesions and melanomas (classes 3 and 11) given by the hotspots away from the correct classification axis.

In Figure 15, we see that the ResNet101V2 model most easily classified chickenpox, cowpox, HFMD, healthy skin, measles, vascular lesions, and monkeypox. The model struggled to differentiate between melanocytic nevi and melanomas (classes 10 and 11), basal cell carcinomas and squamous cell carcinomas (classes 2 and 13), actinic keratoses and squamous cell carcinomas (classes 1 and 13), and benign keratosis-like lesions and melanomas (classes 3 and 11) given by the hotspots away from the correct classification axis.

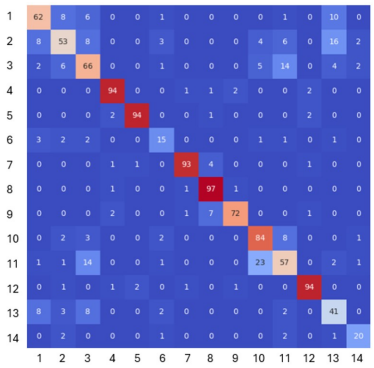

Fig. 15 ResNet101V2 Matrix [20]

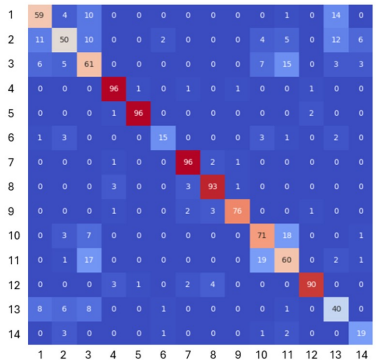

Fig. 16 ResNet152V2 Matrix [21]

In Figure 16, we see that the ResNet152V2 model most easily classified chickenpox, cowpox, HFMD, healthy skin, measles, and monkeypox. The model struggled to differentiate between melanocytic nevi and melanomas (classes 10 and 11), basal cell carcinomas and squamous cell carcinomas (classes 2 and 13), actinic keratoses and squamous cell carcinomas (classes 1 and 13), basal cell carcinomas and vascular lesions (classes 2 and 14), and benign keratosis-like lesions and melanomas (classes 3 and 11) given by the hotspots away from the correct classification axis.

In Figure 17, we see that the InceptionV3 model most easily classified chickenpox, cowpox, HFMD, healthy skin, measles, vascular lesions, and monkeypox. The model struggled to differentiate between melanocytic nevi and melanomas (classes 10 and 11), basal cell carcinomas and squamous cell carcinomas (classes 2 and 13), actinic keratoses and squamous cell

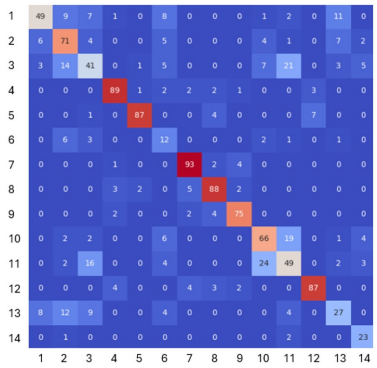

Fig. 17 InceptionV3 Matrix [22]

carcinomas (classes 1 and 13), benign keratosis-like lesions and melanomas (classes 3 and 11), and basal cell carcinomas and dermatofibromas (classes 2 and 6) given by the hotspots away from the correct classification axis.

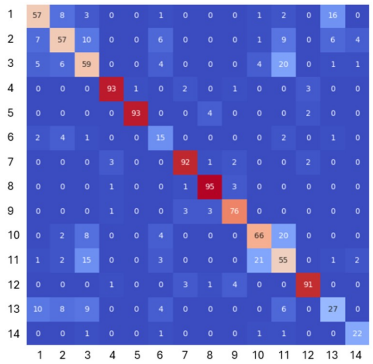

Fig. 18 InceptionResNetV2 Matrix [23]

In Figure 18, we see that the InceptionResNetV2 model most easily classified chickenpox, cowpox, HFMD, healthy skin, measles, vascular lesions, and monkeypox. The model struggled to differentiate between melanocytic nevi and melanomas (classes 10 and 11), basal cell carcinomas and squamous cell carcinomas (classes 2 and 13), actinic keratoses and squamous cell carcinomas (classes 1 and 13), benign keratosis-like lesions and melanomas (classes 3 and 11), and basal cell carcinomas and dermatofibromas (classes 2 and 6) given by the hotspots away from the correct classification axis.

Trends

From the loss and accuracy plots, the InceptionV3 and InceptionResNetV2 architectures were less prone to overfitting, as seen through the convergence of the loss plots as well as the high validation accuracy relative to training accuracy.

The ResNet50V2-based model took the longest duration to train, while the Inception-ResNet hybrid model took the least time to train. Counterintuitively, in the ResNet architectures, adding more layers resulted in a shorter training duration.

Regarding performance metrics, the models based on residual network architectures outperform the Inception-based models. The InceptionV3-based model had the lowest performance with a macro average F1 score of 0.70, precision of 0.70, accuracy of 0.72, and recall of 0.71. The ResNet101V2 model had the strongest performance with an accuracy of 0.79 and macro average F1, precision, and recall scores of 0.78. The residual network architectures were better at minimizing false negatives and false positives, as seen by the generally stronger recall and precision scores. The InceptionResNetV2 hybrid model saw significant improvements over the traditional InceptionV3 model, with a precision of 0.74, accuracy of 0.76, recall of 0.75, and F1 score of 0.74.

All of the models had the strongest performance in classifying infections such as monkeypox, cowpox, HFMD, and measles. The models struggled the most in classifying the carcinomas, melanomas, keratoses, and dermatofibromas. Although the ResNets typically had the highest classification metrics, the Inception models had notably higher recall scores of 0.71 and 0.57 for basal cell carcinomas compared to scores of 0.53 or lower for the other models.

In the confusion matrices, the models usually struggled to differentiate between melanocytic nevi and melanomas (classes 10 and 11), actinic keratoses and basal cell carcinomas (classes 1 and 2), actinic keratoses and squamous cell carcinomas (classes 1 and 13), basal cell carcinomas and benign keratosis-like lesions (classes 2 and 3), and benign keratosis-like lesions and melanomas (classes 3 and 11).

Discussion

Key Findings

The models that employed residual connections significantly outperformed the traditional convolutional neural networks in training efficiency, the most dramatic change of which was seen when the InceptionResNetV2 hybrid model trained 1800 seconds faster than the InceptionV3 model. We can attribute this to residual networks' specialty in improving gradient flow in deeper models by employing skip connections. Skip connections connect earlier layers directly to later ones, allowing for faster convergence. Additionally, skip connections create alternate pathways for information to flow, ensuring all layers contribute efficiently to the model. The shorter training duration of the InceptionResNetV2 compared to the other ResNet models is attributed to the use of residual connections across modules, or groups of convolutional layers, rather than just singular convolutional layers in the ResNetV2s[9]. The use of skip connections over larger portions of layers further increases the efficiency of gradient flow.

The ResNet152V2 ran at roughly 550 ms/step[21], the ResNet101V2 ran at roughly 600 ms/step[20], and the ResNet50V2 ran at roughly 650 ms/step[19]. The counterintuitive result of more layers leading to faster training in the ResNets can be explained by the GPU resource utilization of each of the models. GPUs are designed to handle many operations simultaneously, and models with more layers can sometimes optimize the use of parallel processing. The ResNet models with more layers have more residual connections and more instances of matrix multiplication, which can be performed more efficiently in parallel than in shallower networks.

Inception models had less overfitting than the ResNets because of their use of multiple filter sizes within each layer, including 1x1, 3x3, and 5x5 filters. Using multiple filter sizes allows the network to capture features at both the local and global level. The model can learn a wider range of feature patterns, allowing it to generalize to new data better. This is because the model does not focus on memorizing a single type of feature that could have just been noise in the training data, which prevents overfitting.

The models with residual connections, namely the ResNets and the hybrid model, posted significantly better performance metrics than the InceptionV3. This can be attributed to the residual connections mitigating the vanishing and exploding gradient problem, allowing for more accurate updating of weights during backpropagation. The ResNets performed better than the hybrid model regarding performance metrics as well. This can be explained by the fact that the hybrid architecture is much more complex than the standard ResNet architecture. The higher number of pathways and operations likely made it difficult for the model to effectively converge as the model learned too many parameters. The InceptionV3, lacking residual connections entirely, struggled to converge with its deep network.

Although for basal cell carcinoma classification the Inception models had significantly higher recall scores, or fewer false negatives, this does not indicate a strength of the Inception architecture. In examining the confusion matrices, we see that the InceptionV3 and InceptionResNetV2 predicted basal cell carcinoma (class 2) 116 and 87 times overall, respectively. In contrast, the greatest number of times a standard ResNet model predicted basal cell carcinomas is the ResNet50V2 with 81 times. This tendency of the Inception models to over-predict class 2 is supported by the similar class 2 precision scores across all models and generally lower F1 scores for other classes. This suggests that the higher recall in Inception models may stem from a bias toward class 2, leading to imbalanced performance across other classes.

Regarding class wise performance, all models were more successful in classifying diseases rather than tumors. The images of diseases such as chickenpox had more pronounced textures compared to the tumors, which provides clearer patterns for feature extraction and classification. Images like vascular lesions and healthy skin were easier to classify because they had either very bright or dull colors, respectively. On the other hand, many of the tumors were very visually similar with no visible textures and similar colors. For example, the struggle to differentiate melanocytic nevi and melanomas could be explained by the fact that both nevi and melanomas originate from melanocyte overproduction[24], and the colors and patterns in the images were difficult for the models to distinguish. Moreover, many of the images of keratoses and carcinomas had similar shapes and distributions of colors, making differentiation difficult. Finally, the poor classification of dermatofibromas can be explained by the relative lack of dermatofibroma images in the dataset, leaving a smaller variety of training data for the model to learn features from.

The results from this study are consistent with other studies comparing residual network architectures to Inception architectures. AlSuwaidan (2023) studied the six CNN architectures of VGG16, EfficientNet, InceptionV3, MobileNet, NasNet, and ResNet50 for the classification of the three most common dermatological diseases in the Middle East[8]. In all six of the training iterations, the ResNet50 model outperformed the InceptionV3 model in accuracy, precision, F1 score, sensitivity, and Matthew's Correlation Coefficient (MCC). Stofa et al. (2021) conducted a review of the performance of twelve CNN architectures in skin lesion classification, including ResNet, Inception, Xception, DenseNet, and EfficientNet architectures6. In one of the studies they reviewed, they found that the best accuracy for an individual model was obtained by ResNeXt101 and the best accuracy for an ensemble model was obtained by the combined network of InceptionResNetV2 and ResNetXt101, both of which use skip connections. In another article in the review of Stofa et al. (2021), it was found that the InceptionResNet architecture outperformed the standard Inception architecture in both the segmentation and classification of skin lesions.

Implications

Based on the findings, for skin lesion classification, we can conclude that the use of residual connections has the potential to improve training efficiency and convergence, especially when used in a hybrid approach with the Inception architecture, as the Inception models showed fewer signs of overfitting. The findings from this study suggest that models with residual connections could significantly impact clinical practice in dermatology by reducing diagnostic error, efficiency, and confidence. For example, models used to distinguish benign and malignant tumors could allow for early detection and treatment, improving patient outcomes. However, there are numerous challenges in applying these models in the real world. Many clinical settings globally may not have access to the infrastructure or funding to support complex machine learning technology. Additionally, machine learning models require an immense amount of data

to train, which is one of the most severe bottlenecks for their implementation. This is especially true in medical imaging, where patient data is protected by strict regulations, such as the Health Insurance Portability and Accountability Act (HIPAA), which requires patient consent to release personal health information. With limited data, it is important to make sure datasets are representative of all people and are not biased towards certain populations, such as those with varying skin tones in the case of skin lesion classification. Moreover, implementing artificial intelligence requires changing healthcare practices to merge the skills of physicians and models which will require mass experimentation, education, and approval. Addressing these challenges with foresight, diligence, and patience is crucial for AI to serve as a complement, rather than a crutch, in patient care as a whole.

Limitations

A possible limitation includes a GPU usage constraint, which prevented the models from running past sixty epochs. It is possible that we would see varying convergence and the activation of early stopping if the models were allowed to train until a definite plateau. Secondly, this study only evaluated performance across fourteen classes, while a model in the clinical setting would need to be evaluated on dozens if not hundreds of different skin disorders.

0.1 Recommendations

Future studies in skin lesion classification should experiment with a variety of hybrid models that employ residual connections. These studies should focus on changes in training efficiency and performance with varying degrees of model complexity. There should also be increased focus on hyperparameter tuning for the classification of tumors as opposed to infections, as the models already demonstrated strong generalization in detecting infections.

The metrics of the ResNet models reinforce the idea that ResNets offer superior performance in skin lesion image classification tasks, which is important to note in the development of dermatological image classification models. However, the Inception-based models were less prone to overfitting, and with proper hyperparameter tuning, may lead to more accurate results in a trade-off for more computational resources. The hybrid model demonstrated high efficiency while retaining a relatively high accuracy, making it a compelling option for future experimentation.

Methods

Dataset

Sourcing: The dataset used for developing these models contains 10,940 training samples, 1,177 validation samples, and 1,185 testing samples14. This is roughly an 80% training, 10% validation, and 10% testing data split. The data is split into the following fourteen classes: Actinic keratoses; basal cell carcinomas; benign keratosis-like lesions; chickenpox; cowpox; dermatofibromas; hand, foot, and mouth disease (HFMD); healthy; measles; melanocytic nevi; melanomas; monkeypox; squamous cell carcinomas; and vascular lesions. All data was sourced from the International Skin Imaging Collaboration 2019 Challenge Dataset (ISIC 2019)[16] and the Mpox Skin Lesion Dataset Version 2.0 (MSLD v2.0)[17]. The two datasets were merged to obtain the fourteen classes. The ISIC 2019 dataset was compiled by collecting data from various international clinical centers. The MSLD v2.0 dataset was compiled via manual web-scraping[17].

The dataset used[14] was adapted from a Kaggle dataset[15], but certain images were removed because of a large class imbalance. A class imbalance occurs when certain classes have far more data than other classes used for training the model. Class imbalances lead to bias in the model towards the majority class, as the model encounters more of its data. Consequently, the model will have poor performance in detecting minority classes, skewed performance metrics due to imbalanced testing data, and trouble generalizing to new data that the imbalanced classes do not accurately represent. In the original dataset, there were extreme class imbalances, the most prominent of which was in the training data where there were 10,300 images of melanocytic nevi and only 191 images of dermatofibroma, a roughly 54:1 ratio. Using a systematic sample of 1% of images every 10% of the original data, images were selected to be included in an adapted dataset to create a milder class imbalance and to ensure accurate representation of the original data. The milder class imbalance will be addressed in the Training Parameters and Procedures section. In the adapted dataset that we used for the models, the most prominent class imbalance was in the training data where the maximum number of images per class was 1,000 and the minimum was the 191 images of dermatofibroma, a roughly 5:1 ratio.

A detailed view of the sources and distribution of the classes is available in Table 1.

Decreasing the amount of data leads to a decrease in the number of computational resources that the model requires but can have negative effects on a model's ability to generalize to new data. The highest accuracy achieved by any of the models was 0.79, indicating that a lack of data limited the models' learning capabilities. However, from the loss and accuracy plots, we see that none of the models experienced extreme overfitting,

Class	Training Image Count	Validation Image Count	Testing Image Count	Sourced from MSLD v2.0	Sourced from ISIC
Actinic Keratoses	693	86	88		Yes
Basal Cell Carcinomas	1000	100	100		Yes
Benign Keratosis-like Lesions	1000	100	100		Yes
Chickenpox	900	100	100	Yes	
Cowpox	792	99	99	Yes	
Dermatofibromas	191	23	25		Yes
Healthy	1000	100	100	Yes	
Hand, Foot, and Mouth Disease	1000	100	100	Yes	
Measles	660	82	83	Yes	
Melanocytic Nevi	1000	100	100		Yes
Melanomas	1000	100	100		Yes
Monkeypox	1000	100	100	Yes	
Squamous Cell Carcinomas	502	62	64		Yes
Vascular Lesions	202	25	26		Yes

Table 1 Class distribution and sourcing [14]

indicating that the models' abilities to generalize to new data was not heavily impacted.

Preprocessing

Data augmentation was applied using Image Data Generator [25] to the training data so our model would be more robust in handling variation. The training images were randomly vertically and horizontally flipped and rotated. The images were also randomly shifted in all directions up to 10%. Shearing (angling) and zooming were applied up to 10%. Finally, brightness and RGB channels were modified up to 10%. After data augmentation, a preprocessing function specific to each model was run. This ensured that the image sizes were valid inputs, and all pixel values were rescaled to the range (0, 1). Figure 19 shows some sample images before and after data augmentation.

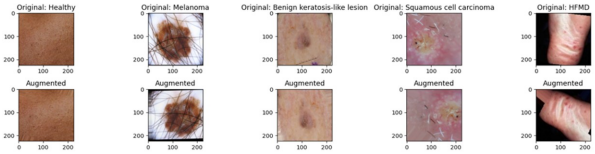

Fig. 19 Images before and after data augmentation [22]

Architectural Development

Independent Variables: In developing each model architecture, certain techniques were held constant for accurate comparison. After loading the base model architecture, all the layers were frozen, and the tops (fully connected or dense layers) were removed and replaced with new dense layers, as shown in Figure 20. This is necessary because the fully connected layers in pre-trained models are optimized for the original training dataset, in this case, ImageNet, and might not generalize accurately to skin lesion data. By replacing the dense layers, the model can learn features relevant to skin lesion classification, such as distinguishing between tumors and diseases.

global_average_pooling2d (GlobalAveragePooling2D)	(None, 2048)	0	post_relu[0][0]
batch_normalization (BatchNormalization)	(None, 2048)	8,192	global_average_pooling…
dense (Dense)	(None, 256)	524,544	batch_normalization[0]…
dropout (Dropout)	(None, 256)	0	dense[0][0]
batch_normalization_1 (BatchNormalization)	(None, 256)	1,024	dropout[0][0]
dense_1 (Dense)	(None, 64)	16,448	batch_normalization_1[…
dropout_1 (Dropout)	(None, 64)	0	dense_1[0][0]
batch_normalization_2 (BatchNormalization)	(None, 64)	256	dropout_1[0][0]
dense_2 (Dense)	(None, 14)	910	batch_normalization_2[…

Fig. 20 New dense layers [22]

A global average pooling layer was included to decrease the number of trainable parameters and flatten the feature maps before entering the dense layers. This aids in decreasing the computational resources needed and preventing overfitting from learning too many parameters. There were three dense layers that were added to the model. The first dense layer had 256 neurons, the second dense layer had 64 neurons, and the third dense layer had 14 neurons. The number of neurons was determined through empirical testing from running previous iterations of the models. Batch normalization layers adjust a layer's outputs to have a mean of zero and a standard deviation of one, which aids in gradient descent and speeds up training [26]. Additionally, batch normalization introduces noise which helps prevent overfitting, or memorizing training data, which would lead to poor testing results. Finally, dropout layers were implemented to also aid in preventing overfitting. Dropout layers randomly set certain neurons' outputs to zero, as shown in Figure 21. Our dropout value of 0.2 means this occurs in 20% of neurons, so they will not participate in the forward pass or backpropagation to update weights. This ensures the model does not become overly reliant on any one neuron.

The fully connected layers use the Rectified Linear Unit

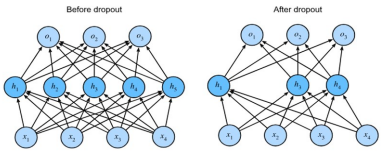

Fig. 21 Layers of neurons before and after dropout[27]

(ReLU) and Softmax activation functions. Activation functions introduce nonlinearity to the data, building on the z=Wx+b function explained earlier. The ReLU function is given by f(x)=max (0, x)[28], meaning that all negative outputs will be set to 0 and all positive outputs will be retained. This promotes sparsity, meaning many neurons' outputs are set to 0 so there are fewer features to compute, improving efficiency and potentially reducing overfitting. The ReLU function is used in the first two fully connected layers. However, the Softmax activation function is used for the last dense layer, or the classification layer. Softmax transforms outputs into probabilities for each class on a scale from zero to one that all sum to one[29]. This helps us understand the confidence of a model's prediction. Therefore, the last dense layer has 14 neurons to classify an image into one of the 14 classes of the dataset.

Base Models

ResNetV2s: All ResNet architectures use residual connections. ResNetV2 architectures differ from ResNetV1 architectures in that the ReLU activation and batch normalization occur before the convolutional layers instead of after[9]. ResNet50V2, ResNet101V2, and ResNet152V2 have 50, 101, and 152 convolutional layers, respectively. Each model has 546,638 trainable parameters. The three models also have twenty-four, forty-three, and fifty-eight million total parameters, respectively. ResNetV2s were chosen for their ability to mitigate the vanishing and exploding gradient issues that are common in very deep networks. These deeper networks can capture many levels of features, which is essential for complex medical diagnosis tasks. ResNetV2s of varying layer counts were chosen to experiment with how model depth influences performance metrics and computational efficiency, specifically when ResNet architectures have been employed.

InceptionV3: Inception models are traditional convolutional neural networks. InceptionV3 has eleven modules of convolutional layers. InceptionV3 builds off the original Inception model by using a variety of filter sizes to aid in accuracy and efficiency[9]. The InceptionV3 model has 546,638 trainable parameters and twenty-two million total parameters. Inception models are one of the most commonly used image classification models. The varying filter sizes allow them to capture details at both a local and global level, which is important in skin lesion classification for generally differentiating between infections and tumors, or nuanced differentiation between different types of tumors themselves.

InceptionResNetV2: InceptionResNets are a hybrid of Inception architectures, which specialize in feature extraction, and ResNets, which specialize in gradient flow. By using a hybrid approach, InceptionResNetV2 can use the strengths of Inception models with a lower risk of the vanishing or exploding gradient issue[9]. Our model has 414,542 trainable parameters and fifty-five million total parameters. InceptionResNetV2 was chosen to evaluate the combination of the computational efficiency of ResNets with the extraction techniques of the Inception modules. The ResNetV2s and InceptionV3 models serve as a baseline for comparison with the InceptionResNetV2 model.

Dependent Variables

The dependent variables in this study are the performance metrics of accuracy, F1 score, precision, and recall. These metrics evaluate the performance of each neural network architecture on the 14 classes of skin lesions, both within each class and overall. Testing accuracy gives us a general view of the strongest architectures. Precision allows us to evaluate which model minimizes false positives, while recall allows us to evaluate which model minimizes false negatives. The F1 score takes into account both precision and recall. In the medical setting, recall and the F1 score are especially important as false negatives can lead to missed diagnoses and adverse health outcomes. Additionally, overfitting was considered, which is the gap between the training and validation accuracies, as well as training duration to measure model efficiency.

Training Parameters and Procedures

The NVIDIA Tesla P100 Graphics Processing Unit (GPU) was used to accelerate training, as GPUs can process complex data for machine learning faster than Central Processing Units (CPUs). As the dataset was moderately imbalanced, the compute class weights function was called. The compute class weights function assigns higher weights to minority classes while training to prevent bias towards the majority classes. The Adam (Adaptive Movement Estimation) optimizer was used, which adjusts the learning rate for each parameter automatically and utilizes momentum to accelerate gradients when in the correct direction[30]. The loss function used was categorical cross-entropy. The models were trained for sixty epochs, with a batch size of thirty-two and a learning rate of 0.001. Finally, an early stopping callback was implemented, but this callback was never activated except to restore a model's best weights after training.

Conclusion

This study analyzed the performance of ResNet50V2, ResNet101V2, ResNet152V2, InceptionV3, and InceptionRes-

NetV2 architectures skin lesion image classification. The ultimate objective was to determine which model architectures show promising results for future clinical use, specifically comparing residual neural networks to traditional convolutional neural networks. The results indicate that the ResNets and hybrid models were generally both more accurate and computationally efficient than traditional CNNs, but the Inception and hybrid models were less prone to overfitting. Further experimentation regarding residual connections and hybrid models will provide more insights into the best tools for dermatological image classification and medical image classification as a whole.

Acknowledgments

The author would like to thank Dr. Lakshman Tamil, Yuchen Cai, and Lena Duraisamy Swamikannan from the University of Texas at Dallas for their guidance.

References

1 *Basal Squamous Cell Skin Cancer Statistics*, www.cancer.org/cancer/types/basal-and-squamous-cell-skin-cancer/about/key-statistics.

2 *American Academy of Dermatology Association. "Skin Cancer*.

3 A. C. Society, *Melanoma Survival Rates — Melanoma Survival Statistics*, www.cancer.org/cancer/types/melanoma-skin-cancer/detection-diagnosis-staging/survival-rates-for-melanoma-skin-cancer-by-stage.html.

4 A. Bhattacharya, A. Young, A. Wong, S. Stalling, M. Wei and D. Hadley, *Precision Diagnosis of Melanoma And Other Skin Lesions From Digital Images*.

5 A. Esteva, B. Kuprel, R. Novoa, J. Ko, S. Swetter, H. Blau and S. Thrun, *Dermatologist-level classification of skin cancer with deep neural networks*, https://doi.org/10.1038/nature21056.

6 M. Stofa, M. Zulkifley and M. Zainuri, *Skin lesions classification and segmentation: A review*.

7 A. Lopez, X. Giró-i Nieto, J. Burdick and O. Marques, *Skin Lesion Classification from dermoscopic images using deep learning techniques*, https://doi.org/10.2316/P.2017.852-053.

8 L. AlSuwaidan, *Deep learning based classification of dermatological disorders*, https://doi.org/10.1177/11795972221138470.

9 K. Team, *Keras Documentation: Backbones*, Keras.io, keras.io/api/keras_cv/models/backbones/.

10 *ImageNet*, Www.image-Net.org,www.image-net.org/about.php.

11 V. Phung and W. Rhee, *A High-Accuracy Model Average Ensemble of Convolutional Neural Networks for Classification of Cloud Image Patches on Small Datasets*, www.researchgate.net/publication/336805909_A_High-Accuracy_Model_Average_Ensemble_of_Convolutional_Neural_Networks_for_Classification_of_Cloud_Image_Patches_on_Small_Datasets ResearchGate, MDPI,.

12 X. Sun, J. Peng, Y. Shen and H. Kang, *Tobacco Plant Detection in RGB Aerial Images*, https://doi.org/10.3390/agriculture10030057.

13 L. Cheng, Y. Ji, C. Li and X. Liu, *Improved SSD Network for Fast Concealed Object Detection and Recognition in Passive Terahertz Security Images*, https://doi.org/10.1038/s41598-022-16208-0.

14 V. Jayanti, *Skin Lesions*, www.kaggle.com/datasets/vinayjayanti/skin-lesion-image-classification.

15 A. AL-Rufai, *Skin Lesions Classification Dataset*, www.kaggle.com/datasets/ahmedxc4/skin-ds, Kaggle.com,.

16 A. Maranhão, *Skin Lesion Images for Melanoma Classification*.

17 J. Paul, *Mpox Skin Lesion Dataset Version 2.0 (MSLD V2.0*.

18 C. Prasanna, *Classification Report Explained — Precision, Recall, Accuracy, Macro Average, and Weighted Average*.

19 V. Jayanti, *Skin Lesion Classification ResNet50V2*.

20 V. Jayanti, *Skin Lesion Classification ResNet101V2*.

21 V. Jayanti, *Skin Lesion Classification ResNet152V2*.

22 V. Jayanti, *Skin Lesion Classification InceptionV3*.

23 V. Jayanti, *Skin Lesion Classification InceptionResNetV2*.

24 C. Chang and W. Sung, *Tzu Chi Medical Journal*, **34**, 1,.

25 *ISIC Challenge*, Challenge.isic-Archive.com.

26 Tensorflow, Tf.keras.preprocessing.image.ImageDataGenerator.

27 K. Doshi, *Batch Norm Explained Visually — How It Works, and Why Neural Networks Need It*.

28 A. Zhang, Z. Lipton, M. Li and A. Smola, *4.6. Dropout — Dive into Deep Learning 0.17.6 Documentation*.

29 *ReLU Activation Function — Dremio*, www.dremio.com/wiki/relu-activation-function/.

30 S. Saxena, *Softmax — What Is Softmax Activation Function — Introduction to Softmax*, www.analyticsvidhya.com/blog/2021/04/introduction-to-softmax-for-neural-network/.