A picture containing symbol, text, circle, font

Description automatically generated

The Egyptian E-Learning University

Faculty of Computers and Information Technology

Skin Diseases Detection

Graduation Project Documentation

**Supervised by:**

Dr. Yasser Abd El Hamid

Eng. Yasmine Mahmoud

**Team members:**

Raed Mohamed Badr 20-00898

Banoub Nagy Edwar 20-01025

Sherif Mohamed Waheed 20-01153

Mohamed Kamal Abdelaziz 20-01080

Rawda Abdelhadi Alkashef 20-00774

Rowan Abdelkader Mohamed 20-01472

Maryam Mohamed Abdelsamad 20-00120

Acknowledgement

We extend our heartfelt appreciation to Dr. Yasser Abd El Hamid. His guidance and mentorship have been instrumental in the successful completion of this graduation project. His expertise, constructive feedback, and dedication to fostering academic growth have greatly influenced the development of this work. We are grateful for the opportunities to learn and collaborate under his insightful supervision.

We would also like to express our gratitude to Eng. Yasmine Mahmoud, who served as a co-supervisor for this project. Eng. Yasmine's technical acumen and collaborative spirit have significantly enriched the project's content and outcomes. Her support and encouragement have been invaluable, contributing to the overall success of this endeavor.

We extend our sincere thanks to both Dr. Yasser Abd El Hamid and Eng. Yasmine Mahmoud for their crucial roles in guiding and shaping this graduation project.

Abstract

Skin diseases have a profound impact on individuals' well-being and health, being pervasive across all age groups. As some of the most prevalent health issues globally, they pose a substantial burden, encompassing psychological, social, and economic challenges for patients, their families, and society at large. This intricate concept of dermatological burden underscores the urgency for innovative approaches in disease identification and management.

By harnessing the power of deep learning, which excels at capturing complex patterns, this paper discusses two algorithms ResNet-152 and DenseNet-121. The chosen algorithm operates on ISIC 2019 dataset to detect a nuanced spectrum of skin diseases. The investigation aims to contribute to a more precise and efficient means of identification, potentially alleviating the multifaceted challenges posed by dermatological conditions. The study's scope encompasses the exploration of eight distinct skin conditions, reflecting a comprehensive analysis within the domain of dermatology.

Table of Contents

List of Abbreviations6

List of Figures7

List of Tables8

1. Introduction9
   1. Problem overview9
   2. Aim and goal9
   3. Documentation overview10
2. Datasets12
   1. Introduction12
   2. HAM1000013
   3. ISIC 201914
   4. PAD-UFES-2016
   5. BCN2000018
   6. Derm7pt20
   7. ISIC 201722
   8. DermNet23
   9. PH225
   10. Dermofit26
   11. Comparison table 28
   12. Conclusion 31
3. Related work and algorithms32
   1. Introduction32
   2. ResNet-15233
   3. ResNet-10136
   4. Inception-v338
   5. DenseNet-12139
   6. DenseNet-20141
   7. MobileNet-v242
   8. MobileNet-v2 & LSTM44
   9. VGGNet47
   10. VGG1649
   11. Comparison table53
   12. Conclusion56
4. Proposed model57
   1. Introduction57
   2. Dataset57
   3. Model Architecture59
   4. Training Procedure61
5. Experimental results63
   1. Main experiment63
   2. Experiments comparisons66
6. Web development67
   1. Front-End development67
   2. Back-End development69
   3. Deployment70
7. Conclusion and Future work72
8. References76
9. Appendices79
   1. Code79
      1. Front-End code79
      2. Back-End code97
      3. AI code98
   2. Images108

**List of abbreviations**

|  |  |
| --- | --- |
| **Abbreviations** | **Meaning** |
| **HAM 10000** | Human Against Machine with 10000 |
| **ISIC 2019** | International Skin Imaging Collaboration 2019 |
| **PAD-UFES-20** | Patient and Dermoscopic Images -  Universidade Federal do Espírito Santo - 2020 |
| **BCN 20000** | Barcelona 20,000 dermoscopic images of skin lesions |
| **Derm7pt** | Dermatology 7-point checklist |
| **ISIC 2017** | International Skin Imaging Collaboration 2017 |
| **DermNet** | Dermatology Network or Dermatology on the Net |
| **PH2** | Pedro Hispano Hospital |
| **AI** | Artificial Intelligence |
| **DL** | Deep Learning |
| **CNN** | Convolutional Neural Network |
| **ResNet** | Residual Neural Network |
| **DenseNet** | Densely Connected Convolutional Network |
| **LSTM** | Long Short-Term Memory |
| **VGG** | Visual Geometry Group |
| **Cubic SVM** | Cubic Support Vector Machine |

**List of figures**

|  |  |  |
| --- | --- | --- |
| **Figure no.** | **Name** | **Page no.** |
| 1 | Architecture of Resnet152 | 34 |
| 2 | Architecture of Resnet101 | 36 |
| 3 | Architecture of Inceptionv3 | 38 |
| 4 | Architecture of DenseNet121 | 40 |
| 5 | Architecture of DenseNet201 | 42 |
| 6 | Architecture of MobilNetV1 | 43 |
| 7 | Component of LSTM | 46 |
| 8 | Architecture of MobilNetV2 | 46 |
| 9 | Architecture of VGG19 | 48 |
| 10 | Architecture of VGG16 | 51 |
| 11 | DenseNet-121 architecture | 59 |
| 12 | Custom layers architecture | 60 |
| 13 | Confusion matrix of DenseNet121 | 65 |

**List of tables**

|  |  |  |
| --- | --- | --- |
| **Table** | **Name** | **Page no.** |
| 1 | Datasets Comparison | 28 |
| 2 | Algorithms Comparison | 53 |
| 3 | Skin Diseases That Have Been Worked On | 58 |
| 4 | Data Augmentations | 58 |
| 5 | Model Evaluation Metrics of DenseNet121 | 64 |
| 6 | Experiments Comparisons | 66 |

**Introduction**

* 1. **Problem overview**

Skin diseases, prevalent across age groups, profoundly impact well-being globally, posing multifaceted challenges. Beyond physical discomfort, individuals navigating dermatological issues face mental health struggles, societal stigmatization, and economic burdens. Accessibility to timely and accurate medical assistance compounds the problem, with barriers like geographical limitations, lack of local expertise, and personal concerns hindering prompt examination. This delay not only extends suffering but also amplifies psychological distress as individuals seek reassurance about their health. Addressing these accessibility challenges is crucial for developing innovative solutions in skin disease identification. Additionally, skin conditions are estimated to affect 1.8 billion people worldwide, with skin infections being the commonest cause of disease in tropical and resource-poor settings [[1](#ref_intro)].

* 1. **Aim and goal**

The objective is tackling the existing challenges in the detection of skin diseases by Leveraging the capabilities of deep learning, the project aims to provide patients with a user-friendly platform that enables them to independently initiate the identification process for various skin conditions. By integrating sophisticated algorithms and intuitive interfaces, our goal is to empower individuals to take a proactive approach towards their dermatological health. This solution seeks to bridge the gap in accessibility to timely medical assistance, particularly in areas with geographical limitations or a lack of local expertise. Through this project, we aspire to ease the way individuals engage with their skin health, fostering a sense of autonomy and timely intervention.

* 1. **Documentation overview**

In our pursuit to contribute to the nexus of dermatology and deep learning, this documentation embarks on a focused exploration, undertaking a comparative analysis of key skin disease datasets and deep learning algorithms. By delving into the intricacies of these datasets, we aim to uncover not only their structural nuances but also the unique contributions they make to advancing our understanding of dermatological conditions.

The datasets under scrutiny HAM10000, ISIC 2019, PAD-UFES-20, BCN20000, Derm7pt, ISIC 2017, DermNet, PH2, and Dermofit - each offer a distinct lens through which we can view skin diseases. We traverse through the year of publication and authors, acknowledging the diverse origins of these datasets. Their structures, from the organization of images to metadata management, are meticulously outlined, providing a foundation for understanding their utility.

Within this visual tapestry, we explore the characteristics of the images, ranging from resolution to capturing techniques. The datasets predominantly feature colorful JPEG images captured through various methods, adding a layer of complexity to the classification task. Encompassing a spectrum of skin diseases, these datasets contribute to a simple understanding of dermatological conditions, embracing both non-cancerous which is our goal and cancerous afflictions.

Yet, our exploration doesn't shy away from challenges. We critically examine issues within each dataset, ensuring transparency about their limitations. Solutions proposed aim to enhance the quality and reliability of the datasets, providing a roadmap for optimizing their utility in decision-making processes and applications reliant on accurate data.

At the end of the section on datasets, we put together a table that contains a comparison between all the types of data that we discussed in this section, in addition to the best types of datasets that are likely to be used in our project.

Transitioning to the realm of deep learning models, our focus extends to ResNet-152, ResNet101, Inception-v3, DenseNet-121, DenseNet-201, MobileNet v1, MobileNet v2 & LSTM, VGGNet, and VGG16 - algorithms that have demonstrated efficacy in skin disease classification. Through a structured lens, we explore their origins, dataset usage, preprocessing techniques, structures, layers, model-building processes, and result metrics, ensuring a comprehensive understanding, and we offer observations about the pros and cons of employing these skin disease classification algorithms.

At the end of the section on algorithms, we put together a table that contains a comparison between all the research papers that we discussed in this section, in addition to the best types of algorithms that are likely to be used in our project.

**Datasets**

Skin disease recognition and classification using deep learning techniques rely on the availability and quality of skin disease datasets, which are collections of skin images with labels or annotations. However, the existing datasets for skin disease research are often incomplete, inconsistent, or imbalanced. In this documentation, we provide a comprehensive overview of the current state-of-the-art skin disease datasets and at the end of the datasets chapter, depended on our comparison we will determine what is best collection of datasets to be used in our project.

* 1. **Introduction**

we rely on nine points to detail each dataset:

Overview: Offering a comprehensive look into skin disease datasets. Each dataset’s purpose, structure, and characteristics are succinctly outlined.

Goal: The primary objective behind assembling these datasets

Year of publication/Author: The datasets discussed here span a range of publication years and authors, showcasing contributions from multiple researchers and institutions.

Dataset structure: Each dataset exhibits its unique structure, ranging from the arrangement of images into distinct train, test, and validation sets to the organization of metadata to track the images.

Image characteristics: Across these datasets, the images vary in terms of resolution, format, and capturing techniques. They predominantly comprise colorful JPEG images captured through dermoscopic examinations, skin surface microscopy, and smartphone devices.

Diseases characteristics: The datasets cover a spectrum of skin diseases, including both cancerous and non-cancerous conditions.

Problems: we aim to elucidate and articulate the challenges and issues inherent in each dataset and shortcomings, anomalies, or discrepancies within the data.

Consequences: offers insights into how each issue may impact the quality, reliability, interpretability, decision-making processes, or any applications reliant on accurate and consistent data.

Solutions: proposing strategies and practical and actionable steps that can be taken to enhance the quality and reliability of the datasets to make informed decisions.

* 1. **HAM 10000**

The HAM10000 dataset [[2](#ref_ham1000_paper)] is a comprehensive collection of dermatoscopic images aimed at facilitating research in automated diagnosis of pigmented skin lesions. It contains 10,015 dermatoscopic images representing various diagnostic categories related to pigmented lesions.

This dataset aims to facilitate the training of neural networks for automated diagnosis by employing cleaning methods, ensuring comprehensive representation of diagnostic categories, and validating cases through pathology, expert consensus, or other confirmatory techniques.

The dataset was published on 2018-06-04 by Tschandl, Philipp (Medical University of Vienna).

Dataset does not specify divisions into distinct train, test, and valid sets. Instead, it comprises a collection of 10,015 dermatoscopic images.

The images are stored in two separate folders:

HAM10000\_images\_part1.zip with 5000 images and HAM10000\_images\_part2.zip with 5015 Images.

HAM10000\_metadata is a csv file which contains lesion\_id column to track lesions images and its disease type [[3](#ref_ham1000_dataset)].

The images are colorful, resolution is 600\*450, and format is JPEG. collected using technique involving skin surface microscopy from Medical University of Vienna.

It includes seven different skin diseases, including five non-cancerous diseases and two cancerous diseases:

Non-Cancerous:

Actinic Keratoses - 327 images,

Melanocytic Nevi - 6705 images,

Vascular Lesions - 142 images.

Benign Keratosis - 1099 images,

Dermatofibroma - 115 images,

Cancerous:

Melanoma - 1113 images,

Basal Cell Carcinoma - 514 images

There are two problems in HAM1000:

1. Dataset does not specify divisions into distinct train, test, and valid sets.
2. The number of samples in each class varies widely.

The consequences of the two problems:

1. Overfitting which leads to poor generalization.
2. biased predictions where the model tends to predict the majority class, ignoring the minority classes.

So, we can solve these problems using some strategies and techniques like

1. Split data into 80% train / 10% test / 10% validate.
2. Data Augmentation to Enhance dataset diversity.
   1. **ISIC 2019**

The ISIC 2019 dataset [[4](#ref_isic2019_paper)] comprises 25,331 dermatoscopic images spanning eight skin lesion categories. It includes varied sizes of 600 × 450 and 1024 × 1024 but exhibits an unbalanced distribution across labels. Evaluated via balanced multi-class accuracy (recall), it aids in skin cancer detection algorithm development and dermatological research.

The International Skin Imaging Collaboration developed the ISIC archive for engaging clinicians and computer vision researchers. The ISIC 2019 dataset includes dermoscopic image databases for common skin lesions.

The dataset was published on December 17, 2018, by (Tschandl, Codella, Gutman, Celebi, Helba, Marchetti, Dusza, Kalloo, Liopyris, Mishra, Kittler, Halpern, Combalia, Codella, Rotemberg, Helba, Vilaplana, Reiter, Carrera, Barreiro, Halpern, Puig, Malvehy).

The dataset comprises 25,352 images divided into Test, Train, and Validate subsets. The Test subset includes 1,930 images (7.6%), Train consists of 21,512 images (84.8%), and validate contains 1,910 images (7.5%).

The images are colorful, contain images of size 600 × 450 and 1024 × 1024, and format is JPG. Gathered from multiple global sources, such as healthcare providers, clinics, and research centers specializing in dermatology.

It includes eight different skin diseases, including five non-cancerous diseases and three cancerous diseases:

Non-Cancerous:

Melanocytic nevus: Test - 965, Train - 11000, Valid - 931

Actinic keratosis: Test - 75, Train - 716, Valid - 76

Benign keratosis: Test - 203, Train - 2215, Valid - 206

Dermatofibroma: Test - 11, Train - 206, Valid - 22

Vascular lesion: Test - 24, Train - 202, Valid – 27

Cancerous:

Melanoma: Test - 360, Train - 3812, Valid - 350

Basal cell carcinoma: Test - 250, Train - 2820, Valid - 253

Squamous cell carcinoma: Test - 42, Train - 542, Valid – 45

There are two problems in ISIC 2019:

1. The number of samples in each class varies widely, especially in the training and validation folder.
2. There is hair on the skin that covers parts of the disease.

The consequences of the problems:

1. bias during the training process towards classes with a greater number of samples and biased predictions, where the model tends to predict the majority class, ignoring the minority classes.
2. Overfitting or Poor Generalization where the model may be overfit on the classes with more samples and fail to generalize and learn distinguishing features samples.
3. Hair can introduce noise and interference in the images and cover key features of skin diseases and making it harder for the model to distinguish patterns or textures for classification.

So, we can solve these problems using some strategies and techniques like

1. Apply data augmentation techniques to artificially increase the number of samples in the classes with fewer images. Techniques like rotation, flipping, scaling, and adding noise can help diversify the dataset.
2. Marge with other dataset to increase samples of classes with fewer samples and balance the number of images for all disease.
3. Reducing the number of images for disease with many images to balance them with diseases with a small number of images.
4. Apply various denoising techniques like median filtering, non-local means denoising, wavelet transforms, thresholding to reduce the impact of noise caused by hair.
   1. **PAD-UFES-20**

The PAD-UFES-20 dataset [[5](#ref_padufes20_paper)] is a skin lesion benchmark composed of clinical images collected from smartphone devices and a set of patient clinical data containing up to 21 features. It was developed to tackle skin lesion analysis.

The goal of the PAD-UFES-20 dataset is to support future research and the development of new tools to assist clinicians in detecting skin cancer.

Dataset was authored by a team of researchers including Andre G. C. Pacheco, Gustavo R. Lima, Amanda S. Salomão, Breno A. Krohling, Igor P, and many more. The dataset was collected from 2018 to 2019 and was released in October 2020

The dataset consists of 1373 patients, 1641 skin lesions, and 2298 images for six different diagnostics: three skin diseases and three skin cancers. In addition, each image has up to 21 patient clinical features including the patient’s age, skin lesion location, Fitzpatrick skin type, and skin lesion diameter also four identifying features (patient ID, lesion ID, Image ID, and if the sample is biopsy-proven) in CSV file.

The images are stored in two separate folders:

imgs\_part\_1.zip 1.16 GB

imgs\_part\_2.zip 1.05 GB

imgs\_part\_3.zip 1.14 GB

All images are raw, no image processing to enhance visualization, and format is PNG. Since the images in this dataset are collected using different smartphone devices, they present different resolutions, sizes, and lighting conditions. Thus, it aims to simulate the real world.

Among six skin diseases in the dataset, there are three cancerous conditions and three non-cancerous conditions represented.

Non-Cancerous:

* Actinic Keratosis - 730 images.
* Melanocytic Nevus of Skin - 244 images.
* Seborrheic Keratosis - 235 images.

Cancerous:

* Squamous Cell Carcinoma - 192 images.
* Basal Cell Carcinoma of skin - 845 images.
* Malignant Melanoma - 52 images.

There are problems in PAD-UFES-20 dataset:

1. Variability in image Properties because images are collected from different smartphone devices may vary in resolution, size, and lighting conditions.
2. Images in folders are not previously categorized but it’s needed to be put in its categories using meta data which require additional code for this process.

The consequence of the problems is that model might struggle to generalize to unseen or differently captured data, affecting their real-world applicability.

These problems can be resolved using some strategies like using standardization techniques, image preprocessing (resizing, normalization) can help mitigate the variability across images and improve model robustness.

* 1. **BCN 20000**

The BCN 20000 dataset [[6](#ref_bcn20000_paper)] is a collection of dermoscopic images of skin lesions that can be used for skin cancer diagnosis using artificial intelligence. It was created by researchers from the Hospital Clinic in Barcelona and contains images of lesions from different body locations, sizes, and colors. The dataset was released in 2019 and is one of the datasets used in the ISIC Challenge.

It aims to study the problem of unconstrained classification of dermoscopic images of skin cancer, including lesions found in hard-to-diagnose locations (nails and mucosa), large lesions which do not fit in the aperture of the dermoscopy device, and hypo-pigmented lesions.

The BCN 20000 dataset, published in 2019 by Combalia et al., was curated by a multidisciplinary team including researchers from Hospital Clínic Barcelona, IBM Research AI, Memorial Kettering Cancer Center, Kitware, and Universitat.

the BCN20000 dataset, composed of 19424 dermoscopic images of skin lesions captured from 2010 to 2016.

The BCN 20000 dataset comprises JPEG images with resolutions spanning from 600x450 to 4288x2848 pixels, offering a diverse set of colorful images. Dermoscopic images were captured using dermoscopic attachments on three high-resolution cameras and stored on a hospital server.

The BCN20000 dataset eight different skin diseases, including five non-cancerous diseases and three cancerous diseases:

Non-Cancerous:

Nevus - 9,467

Seborrheic Keratosis - 1,099 images

Actinic Keratosis - 1,082 images

Dermatofibroma - 1,030 images

Vascular Lesions – 587 images

Cancerous:

Melanoma - 1,113 images

Basal Cell Carcinoma - 5,016 images

Squamous Cell Carcinoma - 1,030 images

There are two problems in BCN20000:

1. The number of samples in each class is imbalanced.
2. There is hair on the skin that covers parts of the disease.

The consequences of the problems:

1. bias during the training process towards classes with a greater number of samples and biased predictions, where the model tends to predict the majority class, ignoring the minority classes.
2. Overfitting or Poor Generalization where the model may be overfit on the classes with more samples and fail to generalize and learn distinguishing features samples.
3. Hair can introduce noise and interference in the images and cover key features of skin diseases and making it harder for the model to distinguish patterns or textures for classification.

So, we can solve these problems using some strategies and techniques like

1. Apply data augmentation techniques to artificially increase the number of samples in the classes with fewer images. Techniques like rotation, flipping, scaling, and adding noise can help diversify the dataset.
2. Marge with other dataset to increase samples of classes with fewer samples and balance the number of images for all disease.
3. Reducing the number of images for disease with many images to balance them with diseases with a small number of images.
4. Apply various denoising techniques like median filtering, non-local means denoising, wavelet transforms, thresholding to reduce the impact of noise caused by hair.
   1. **Derm7pt**

The Derm7pt dataset [[7](#ref_derm7pt_paper)] by Kawahara includes 1011 resized colorful skin lesion images. It assesses seven-point checklist-based prediction, offering 15 disease categories with varying image counts. Train-validation-test splits allocate 413, 203, and 395 cases respectively, enabling comparative analysis of computerized and expert diagnoses in dermatology.

Created to evaluate computerized image-based prediction of the seven-point checklist and compare it with human experts. The authors also proposed a multitask multimodal neural network that can simultaneously predict the diagnosis and the seven criteria from clinical and dermoscopy images.

The dataset was published in March 2019 by Jeremy Kawahara, Sara Daneshvar, Giuseppe Argenziano, and Ghassan Hamarneh.

The dataset consists of 1011 clinical and dermoscopic color images of skin lesions. use 413 cases to train the model, 203 cases to validate design decisions and 395 cases to test and report results.

Images are colorful, Images are in JPEG format. All images were resized to 512×512×3.

The Derm7pt dataset has 15 different skin diseases, including 13 non-cancerous diseases and 2 cancerous diseases:

Non-Cancerous:

Blue nevus - 28 images

Clark nevus - 399 images

Combined nevus - 13 images

Congenital nevus - 17 images

Dermal nevus - 33 images

Recurrent nevus - 6 images

Reed or spitz nevus - 79 images

Dermatofibroma - 20 images

Lentigo - 24 images

Melanosis - 16 images

Miscellaneous - 8 images

Vascular lesion -29 images

Seborrheic keratosis - 45 images

Cancerous:

Basal cell carcinoma - 42 images

Melanoma- 252 images

There are two problems in derm7pt:

1. Imbalanced Dataset for example there are 399 images for Clark nevus but only 6 for Recurrent nevus.
2. Limited Data With only 1011 images, the dataset might be too small for training.

The consequences of the problems:

1. Imbalances in the dataset can lead to biased predictions, model perform well on the majority class.
2. Poor performance of a weak model.

We can solve these problems with:

1. Apply data augmentation techniques to artificially increase the number of samples in the classes with fewer images. Techniques like rotation, flipping, scaling, and adding noise can help diversify the dataset.
2. Marge with other dataset to increase samples of classes with fewer samples and balance the number of images for all disease.
   1. **ISIC 2017**

The ISIC 2017 dataset [[8](#ref_isic2017_paper)] is used in challenge consisted of 3 tasks: lesion segmentation, dermoscopic feature detection, and disease classification. For each, data consisted of images and corresponding ground truth annotations, split into training 2000 image, validation 150image, and holdout test 600 image datasets. Predictions could be submitted on validation and test datasets. The data was collected from different sources.

The goal of the challenge is to help participants develop image analysis tools to enable the automated diagnosis of melanoma from dermoscopic images. Image analysis of skin lesions is composed of 3 parts, Part 1: Lesion segmentation Part 2: Detection and localization of Visual dermoscopic Features, Part 3: Disease classification.

The dataset was published in 8 Jan 2018 by (Noel C. F. Codella, David Gutman, M. Emre Celebi, Brian Helba, Michael A. Marchetti, Stephen W. Dusza, Aadi Kalloo, Konstantinos Liopyris, Nabin Mishra, Harald Kittler, Allan Halpern)

Use training data 2000 images for participants to engage in all 3 components of lesion image analysis. validation dataset 150 images and blind held-out test dataset 600 images.

Data was collected from various sources and different imaging devices. Images are colorful, images in JPEG, PNG and Json format with different resolutions.

The ISCI 2017 dataset has three skin diseases, including two non-cancerous diseases and one cancerous disease:

Non-Cancerous

seborrheic keratosis :254 train, 42 valid, 90 test.

benign nevi:1372 train, 78 valid, 393 test.

Cancerous:

melanoma: 374 train, 30 valid, 117 test.

The Problem of ISIC 2017 is the dataset is imbalanced, as there are more benign nevi than melanoma or seborrheic keratosis images.

Consequence of problem is Poor performance of the model (poor generalization)

Solution of problems is that Employ data augmentation techniques to artificially augment the volume of samples within classes featuring fewer images. Utilizing methods such as rotation, flipping, scaling, and introducing noise can effectively enhance the diversity of the dataset.

* 1. **DermNet**

The DermNet dataset [[9](#ref_dermnet_paper)] is the world's leading free dermatology resource, dedicated to providing comprehensive information on skin conditions. Designed for healthcare professionals, it offers evidence-based content, including a library of over 20,000 dermatology images.

DermNet provides authoritative skin information to healthcare professionals and patients, promotes awareness, supports dermatology research and education, and collaborates with organizations for similar objectives.

DermNet was founded in 1996 by three dermatologists: Dr Amanda Oakley, Dr Mark Duffill, and Dr Marius Rademaker. The current Editor-in-Chief is Dr Ian Coulson, who took over from Dr Delwyn Dyall-Smith in 2021.

The dataset contains 19,500 images in 23 folders but have approximately 40 skin disease categories, organized into train 15,500 images, and test 4,000 images.

The images are stored in JPEG format and have a resolution of 600 x 450 pixels and colored.

There are approximately 40 skin diseases with 2 cancerous conditions and 38 non-cancerous and we will make lines under cancer conditions.

Acne and Rosacea: 1152,

Actinic Keratosis and Basal Cell: 1437

Atopic Dermatitis: 612,

Bullous Disease: 561,

Cellulitis Impetigo and bact: 361

Eczema: 2544,

Exanthems and Drug Eruptions: 505,

Herpes: 507

Hair Loss Photos Alopecia: 299,

Light and Disorders of Pigmentation: 711

Lupus and Connective Tissue: 525,

Melanoma and Nevi and Mole: 579

Nail Fungus: 1301,

Poison Ivy/Contact Dermatitis: 325

Psoriasis Lichen Planus: 1757,

Scabies Lyme: 539,

Systemic: 758

Seborrheic Keratoses: 1714,

Tinea Ringworm: 1625

Urticaria Hives: 265,

Vascular Tumors: 603,

Vasculitis: 521

Warts Molluscum: 2358

The problems of dermnet dataset are:

1. the dataset is that Within each of the 23 folders, various types and forms of skin diseases may coexist, and there is randomness in the combination of images and diseases in all folders.
2. images for each disease does not exceed 200 images.

The consequences of this problem are:

1. The diverse range of skin diseases in a single folder increases complexity of training deep learning models. Manual handling of dataset for labeling and annotation becomes more challenging due to multiple diseases in each folder and time-consuming and error-prone to assign accurate labels to each image.
2. Poor and weak the model predictions

Solutions of these problems are:

1. Clearly organize the data within each folder to separate different types of skin diseases. This could involve creating subfolders for each disease or adopting a structured naming convention.
2. This dataset can be used as a secondary source to increase the main dataset.
   1. **PH2**

The PH2 dataset [[10](#ref_ph2_paper)] acquired at Hospital Pedro Hispano in Portugal, contains 200 melanocytic lesions, categorized into 80 common nevi, 80 atypical nevi, and 40 melanomas. This valuable dataset aids in dermatology research and diagnosis using dermoscopic images.

The PH2 database is intended to be freely accessible for research and benchmarking, addressing the current challenge of inconsistent evaluation sets in the field of dermoscopic image analysis.

The PH2 dermoscopic image database was published on July 8, 2013, by Dra. Joana Rocha and Dra. Marta Pereira.

PH Dataset images folder: Inside this folder there is a dedicated folder for every image of the database, which contains the original dermoscopic image, the binary mask of the segmented lesion as well as the binary masks of the color classes presented in the skin lesion [[11](#ref_ph2_dataset)].

The PH2 dataset features 768 x 560 pixels images of skin lesions with various characteristics and bmp format, including colors, pigment network, dots/globules, streaks, regression areas, and blue-whitish veil.

It includes three different skin diseases, including two non-cancerous diseases and one cancerous disease:

Non-Cancerous:

80 images of common nevi.

80 images of atypical nevi.

Cancerous:

40 images of melanomas.

The problem with this dataset is that small datasets not large enough for deep learning.

Consequence of this problem is that Deep learning require large amounts of data to learn effectively.

we can solve these problems using some strategies and techniques like

1. Split data into 80% train / 10% test / 10% validate.
2. Data Augmentation to Enhance dataset diversity.
   1. **Dermofit**

The Dermofit dataset [[12](#ref_dermofit_paper)] develops an interactive skin cancer image database indexing tool, where users compare 'live' skin lesions/spots to images selected from a database. This could potentially improve the diagnostic accuracy of dermatologists and save lives.

The project goal is to develop an interactive skin cancer image database indexing tool, where users compare live skin lesions/spots to images selected from a database. Based on user feedback on the goodness of the selections, the database will be searched for better matches.

The project starts on August 1, 2008, and continues until July 31, 2011 by Rees, Aldridge, Fisher, and Ballerini.

This data will require purchase. It is not divided into train, test, and valid folders.

There are 1300 images are normal RGB captured with a quality SLR camera under controlled (ring flash) indoor lighting.

It includes ten different skin diseases, including seven non-cancerous diseases and three cancerous diseases:

Non-Cancerous:

Actinic Keratosis – 45 images

Melanocytic Nevus (Mole) - 331 images

Seborrhoeic Keratosis: 257 images

Intraepithelial carcinoma - 78 images

Pyogenic Granuloma - 24 images

Haemangioma - 96 images

Dermatofibroma - 65 images

Cancerous:

Melanoma - 76 images

Basal Cell Carcinoma - 239 images

Squamous Cell Carcinoma - 88 images

The problems of dermofit are:

1. The Dermofit Image Library comprises only 1,300 images across ten different lesion types. Some classes, such as Pyogenic Granuloma, have a small number of images (24)
2. Access to the Dermofit Image Library involves a one-off £75 licence fee. Submission must occur via an institutional email address, and approval by an authorized representative of the institution is required.

consequences of the Dermofit problems are:

1. overfitting and reduced generalization.
2. creating potential access barriers for individual researchers and limiting dataset accessibility.

Solutions of problems are:

1. Increase the effective size of the dataset through augmentation techniques, such as rotation, flipping, and scaling, to provide additional variations for model training.
2. Institutions could consider waiving fees for researchers or providing grants to cover the costs, ensuring broader access to the Dermofit Image Library.
   1. **Comparison table**

**Table 2.1** Datasets Comparison



**Table 2.2** Datasets Comparison (continued)



**Table 2.3** Datasets Comparison (continued)

* 1. **Conclusion**

In conclusion, the meticulous comparison of skin disease datasets has led to the selection of ISIC 2019 dataset for our project. Recognizing their status as the most challenged source of dermatological data.

This selection of ISIC2019 is based on four important criteria. Firstly, the availability of the dataset where it can be accessed. Secondly, the primary objective of the dataset focuses on training neurons for artificial intelligence, rather than being exclusively tailored to the medical domain. Thirdly, the dataset contains an appropriate number of skin diseases. Fourth, each type of skin disease contains a relatively large number of images. which lays a solid foundation for the subsequent phases of our project.

**Related work and algorithms**

Skin disease recognition and classification using deep learning techniques rely on the effectiveness of algorithms employed for image analysis and pattern recognition. However, the choice of algorithms is crucial as they directly impact the accuracy and robustness of the classification system. In this documentation, we provide a comprehensive overview of the current state-of-the-art algorithms utilized in skin disease recognition and classification tasks. At the end of the algorithms chapter, based on our comparison, we will determine the most suitable set of algorithms to be used in our project.

* 1. **Introduction**

Research papers were discussed in some detail in terms of the models and algorithms used in them, in addition to explaining the structure of each algorithm in terms of layers, hyperparameters, and the build of the model, in addition to the data set that was used, preprocessing techniques performed on the data set, and the final accuracy of the model.

We rely on ten points to detail each algorithm:

Paper Name: Name of the paper used for skin disease classification.

Publication/Author: The publication date and authors of the research paper or study where the algorithm was introduced or applied.

Dataset used: The dataset used for training, testing, and validating the algorithm, including its source, size, composition, attributes like image resolution, format, and diversity, and whether it includes dermoscopic examinations or skin surface microscopy.

Preprocessing Techniques: Description of any preprocessing methods or techniques applied to the input data before feeding it into the algorithm.

Structure: This part provides an overview of an algorithm used for skin disease classification, including its architecture, functions, and hyperparameters. It also details the key functions performed by the algorithm's components, such as activation, loss, and optimization algorithms.

Layers: It covers the number of layers it uses and their components.

Model building: When describing the process of model building in the context of skin disease classification, it involves the steps taken to construct, train, and optimize the machine learning or deep learning model used for classification tasks.

Result: This section, specifically focusing on evaluation metrics such as Accuracy, Recall, and Precision, serves to quantitatively assess the performance of the skin disease classification model.

Limitation: It's included factors that may impact the performance, reliability, and generalizability of the model.

Observation about pons and pros: regarding the use of skin disease classification algorithms can help provide a balanced view of their potential benefits and limitations from our point of view.

* 1. **ResNet-152**

Skin Lesions Classification Using Convolutional Neural Networks in Clinical Images [[13](#ref_resnet152_paper)].

This paper was published on 6 Dec 2018 by Danilo Barros Mendes, Nilton Correia da Silva.

This paper merge three datasets:

* The MED-NODE dataset has 170 images divided between 70 melanoma and 100 nevus cases.
* The Edinburgh Dermofit Image Library dataset has 1,300 images divided into 10 lesions.
* Atlas dataset has 3816, images Basal Cell Carcinoma 1,561, Lentigo 69, Malignant Melanoma 228, Melanocytic nevus (mole) 626, Seborrheic keratosis 897, Wart 435.

After merging the datasets, 10% to 20% of them were taken as a test, and the remaining part was increased using data augmentation. After that, it was divided into 80% for training and 20% for validation.

Increase images count by Rotation 0.5, Random zoom 0.4, Flip horizontally 0.7, Flip vertically 0.5, Random distortion 0.8, Lightning variance 0.5. The images of diseases in the data were increased by 20 to 30 times.

Structure: ResNet-152 model is discussed in this paper.

ReLU activation function

Neither optimizer nor loss function is mentioned in this paper.

Hyperparameter:

Base in 0.01, weight decay 0.00001, momentum 0.9, gamma 0.1, batch size 5, max tier 176180, test tier 22023, test interval 2000, step size 17618, tier size 12, epochs to 10.

ResNet-152 consists of 152 layers with 1 MaxPool layer, 151 conv layer, and 1 avgPool layer.

Transfer learning applied: ResNet-152 pre-trained for the ImageNet dataset first.

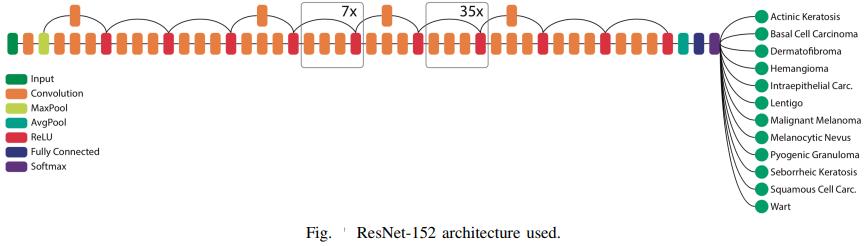


Fig.1 Architecture of Resnet152

Input Layer: Initially receives input data, often images, formatted to the appropriate dimensions for processing.

Convolutional Layers: multiple residual blocks facilitate learning deeper, each containing several convolutional layers responsible for feature extraction.

Max Pooling: reducing spatial dimensions and enhancing computational efficiency.

Rectified Linear Unit (ReLU): An activation function applied after convolutional layers.

Average Pooling: Towards the end of the network, average pooling is applied, aiding in further abstraction and feature summarization.

Fully Connected Layer: Positioned at the end of the network, these dense layers, enabling the network to output predictions.

SoftMax Activation: The final layer utilizes the SoftMax function, converting the network's raw outputs into probabilities for different classes in classification tasks.

Finally, the model used in the testing phase was the product of the iteration number 38,000. This training phase took an uninterrupted total time of 35 hours (approximately 167 seconds for every 50 iterations).

All 11 lesions except for Actinic Keratosis had accuracy over 80% in the confusion matrix, accounting for 78% total accuracy. However, this metric has a bias due to uneven class distribution.

Predicted as Hemangioma, Basal Cell Carcinoma with 100% confidence,

Predicted as Melanocytic Nevus with 97% confidence.

Metric has a bias attached to it, since the distribution of the classes is not even.

It is necessary to further test the model with more data, with more diversity (different ethnicities and ages), and then investigate the results for improvements.

Pros: All 11 lesions apart from Actinic Keratosis had accuracy over 80%.

Cons: Actinic Keratosis accuracy is less than other diseases because its number of images is very small compared to other diseases which lead to bias, A very small number of images were determined for the testing process.

* 1. **ResNet-101**

Early Detection of Skin Cancer Using Deep Learning Architectures: Resnet-101 and Inception-v3 [[14](#ref_resnet101_paper)].

published in October 2019 by Ahmet Demir, Feyza Yilmaz, Onur Kose

Dataset is taken from ISIC-Archive, which is balanced dataset of images for benign skin moles and malignant skin moles.

The training data set contains 2437 images in total, 1330 for benign class and 1107 for malignant class.

660 images are used in the test process, 360 benign and 300 malignant.

200 which are taken from the training part of original dataset randomly are used in validation process, 100 in benign and 90 in malignant.

ResNet-101 Uses residual connections to address the vanishing gradient problem and allows for the training of very deep networks.

ResNet-101 model contains 104 convolutional layers in total, which are used to extract features from the input images. Which are divided into 33 blocks of layers, each consisting of one or more convolutional layers, followed by a summation operator that combines the output of the block with the input of the block or a previous block possible by using residual connections which can be written as y = F(x) + x.

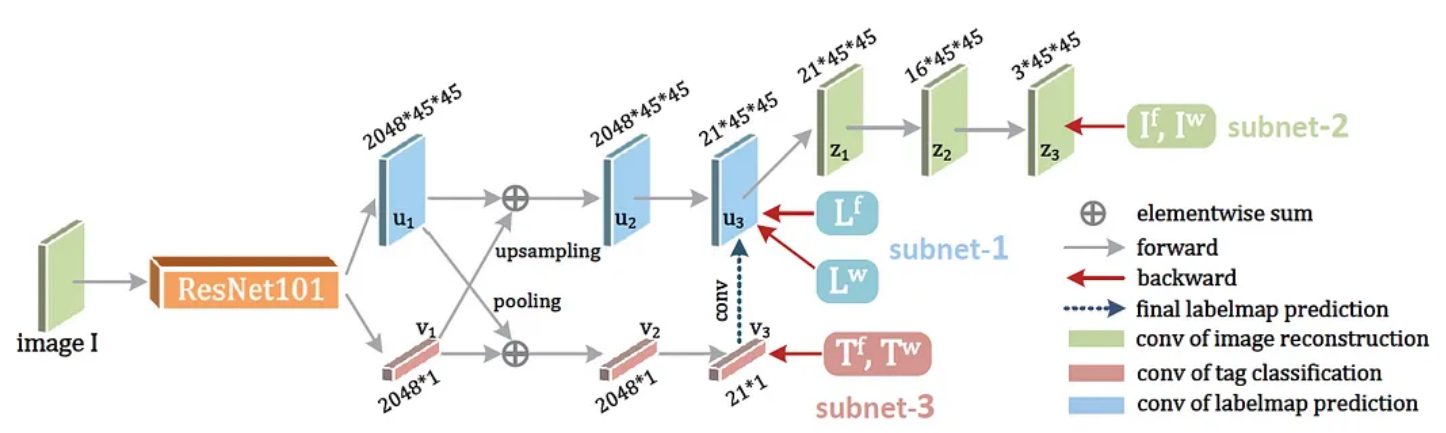


Fig 2. Architecture of Resnet101

The convolutional layers use different filter sizes, strides, and padding to control the size and shape of the output feature maps.  
ResNet-101 model used residual connections to enable the flow of information and gradients across the blocks, and to avoid the vanishing gradient problem that affects deep neural networks.

Out of the 33 blocks, 29 blocks use the previous block’s output directly by using a residual connection and use it as the first operand of the summation operator at the end of the block. The second operand is the output of the convolutional layers within the block.

The remaining 4 blocks use the previous block’s output indirectly by applying a convolution layer with a filter size of 1x1 and a stride of 1, followed by a batch normalization layer, which performs normalization and scaling operations. The output of this layer is then used as the first operand of the summation operator at the end of the block. The second operand is the same as before.

The use of 1x1 convolution and batch normalization layers helps to reduce the number of parameters and to increase the efficiency and stability of the model.

The training process of these two architectures is completed in 60 epochs. The learning rate is determined as 0.001 at the beginning of the training process and this ratio diminishes gradually at every stage of training.

ResNet-101 train and loss lines are almost constant after 50 epochs. No farther increase of accuracy was reached after this point. Validation loss is nearly constant after 30 epochs.

ResNet-101 has a validation accuracy of 0.8900, a benign accuracy of 0.8277, a malign accuracy of 0.8566, and an F-1 score of 0.8409.

Pros of ResNet-101 that it has many layers, allowing it to capture complex features and patterns in images also It utilizes residual connections, which help in addressing the vanishing gradient problem.

And ResNet-101 may also suffer from overfitting if the data is not sufficient or noisy.

* 1. **Inception-v3**

Early Detection of Skin Cancer Using Deep Learning Architectures: Resnet-101 and Inception-v3

October 2019 and published by Ahmet Demir, Feyza Yilmaz, Onur Kose

Dataset is taken from ISIC-Archive, which is balanced dataset of images for benign skin moles and malignant skin moles.

The training data set contains 2437 images in total, 1330 for benign class and 1107 for malignant class.

660 images are used in the test process, 360 benign and 300 malignant.

200 which are taken from the training part of original dataset randomly are used in validation process, 100 in benign and 90 in malignant.

Inception-v3 is a commonly used image recognition model that has been shown to attain an accuracy rate of greater than 78.1% on the ImageNet dataset. Composed of a 42-layer deep neural network

Inception-v3 model contains 42-layer deep neural network and also consists of symmetric and asymmetric building blocks, including convolutions, max pooling layers, average pooling, dropouts, and fully connected layers.

Inception-v3 Uses a combination of convolutional layers with different filter sizes and pooling operations to capture features at different scales.

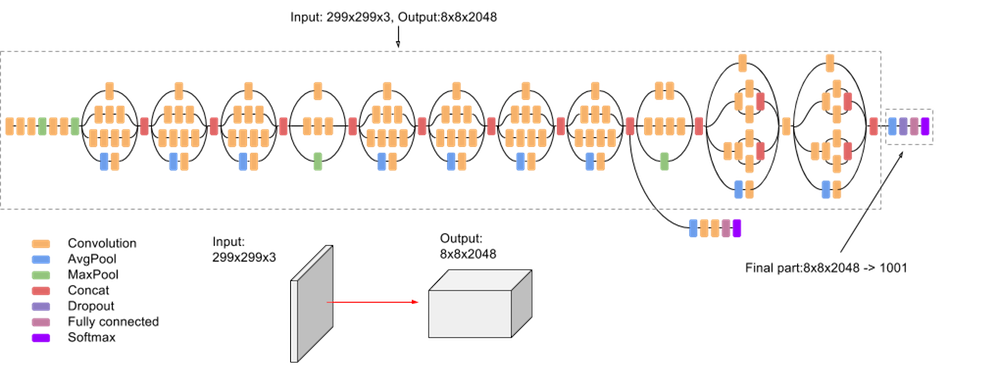


Fig3. Architecture of Inceptionv3

Inception-v3 train, loss and validation loss is scarcely constant after 40 epochs. The train and validation accuracy saturation points are reached after 40 epochs.

Inception-V3 has slightly higher values than ResNet-101 in all four metrics Validation Accuracy, Benign Accuracy, Malign Accuracy and F-1 Score. Where Inception-V3 has a validation accuracy of 0.9000, a benign accuracy of 0.8861, a malign accuracy of 0.8600, and an F-1 score of 0.8742. ResNet-101 has a validation accuracy of 0.8900, a benign accuracy of 0.8277, a malign accuracy of 0.8566, and an F-1 score of 0.8409.

Cons that both can be computationally expensive due to its large number of layers and complex architecture.

Inception-V3 may also lose some information or details by using multiple filter sizes and concatenation.

* 1. **DenseNet-121**

Classification of Skin Cancer with Deep Transfer Learning Method [[15](#ref_densnet121_paper)].

It published on Oct.10,2022 and published by Doaa Khalid Abdulridha Al-Saedi, Serkan Savaş

The dataset used is an open-access dataset called the International Skin Imaging Collaboration (ISIC) skin cancer which included 1800 benign and 1497 malignant tumor images.

In preprocessing dataset were converted to 224 x 224 x 3 dimensions to be used in the pre-trained model also implementing the preprocess input function for every transfer learning algorithm and This function applies image scaling.

In the next step in this phase contains feature extraction pre-trained models were used to extract features through transfer learning.

Use every kind of pre-trained model was implemented separately without the output layer from the original model and added three layers in the end of each model, the first and the second layers are same in their Components.

Every layer of these consists of 128 neurons with ReLU activation function the feature selection and classification are optimized using a newly created Red Fox Optimization (DRFO) algorithm,

The training Loss value decreased below a very small value of 0.025, especially after the 10th epoch.

DenseNet including those in the same dense block and transition layers, spread their weights over several inputs, deeper layers can utilize characteristics that were collected earlier. 120 Convolutions and 4 Avg Pool make up DenseNet-121

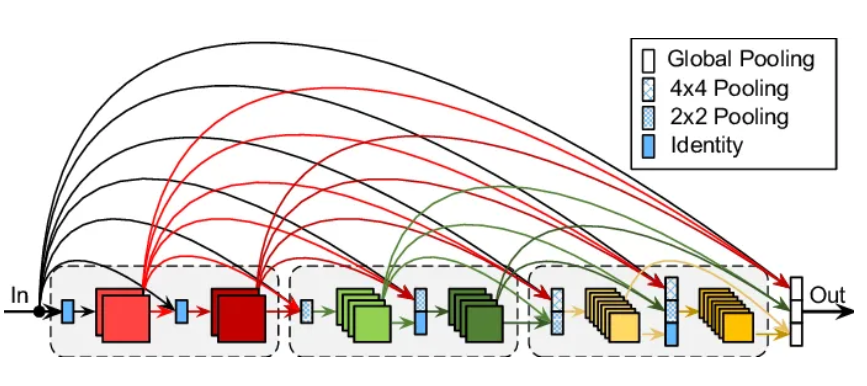


Fig.4 Architecture of DenseNet121

They use data set (ISIC) which included 1800 benign and 1497 malignant tumor images, use preprocessing dataset were converted to 224 x 224 x 3 dimensions to be used in the pre-trained model also implementing the preprocess input function for every transfer learning algorithms and This function apply image scaling and use algorithm (densent121) which including the same dense block and transition layers, spread their weights over several inputs, deeper layers can utilize characteristics that were collected earlier. 120 Convolutions and 4 Avg Pool make up DenseNet-121.They use the feature selection and classification are optimized using a newly created Red Fox Optimization (DRFO) algorithm.in the end they got best accuracy rate that can be achieved with the densent121 is 99.6% in the study. In addition, the validation accuracy rate of this model has reached a high rate of 98.35%

The study uses various models, including DenseNet, Xception, InceptionResNetV2, ResNet50, and MobileNet, to achieve high accuracy rates. The model is implemented separately without the output layer and adds three hidden layers for feature extraction. The DenseNet121 model achieved a 99.6% accuracy rate. However, it faces challenges in dealing with similar images and the similarity of colors in melanoma lesions. To address these issues, data augmentation techniques are employed.

* 1. **DenseNet-201**

Classification of Skin Cancer Lesions Using Explainable Deep Learning [[16](#ref_densnet201_paper)].

It published in 10 May 2023and published by Muhammad Zia Ur Rehman, Fawad Ahmed, Suliman A. Alsuhibany, Sajjad Shaukat Jamal, Muhammad Zulfiqar Ali and Jawad Ahmad.

The dataset available with name (Skin Cancer: Malignant vs. Benign),

is part of ISIC archive it consists of two categories: malignant and benign. A total of 3297 images are present in this dataset.

The category (benign: contains 1800 images), (malignant:contains 1497 images)

They used) contrast expansion technology (which is used in image processing to improve the clarity of features in the image. Which helps in seeing details better, especially in cases where some points aren’t defined or faint.

In this model 201 layers of the original model, DenseNet201 model consists of 4 dense blocks and 3 transition layers that act as connections between the two dense blocks, Inside the dense block, each convolution layer is connected to other convolution layers. After every dense block, the size of the feature map is increased. The transition layers act as the down sampling layers. In DenseNet201, down sampling is performed using average pooling.

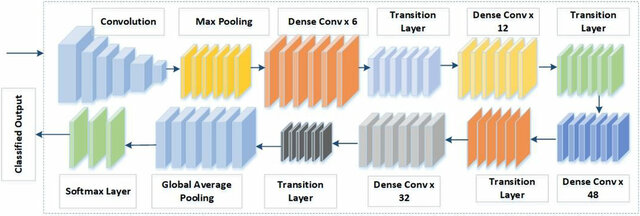


Fig.5 Architecture of DenseNet201

They use part of ISIC archive it consists of two categories: malignant and benign. A total of 3297 images are present in this dataset. The category benign: contains 1800 images, malignant: contains 1497images.

Used contrast expansion technology, which is used in image processing to improve the clarity of features in the image. model consists of 4 dense blocks and 3 transition layers that act as connections between the two dense blocks, Inside the dense block, each convolution layer is connected to other convolution layers. After every dense block, the size of the feature map is increased. The transition layers act as the down sampling layers. In DenseNet201, down sampling is performed using average pooling.

It Reach to Accuracy: 95.50%, Sensitivity: 93.96%, Specificity: 97.06%, Precision: 97.02%, F1 Score: 95.46%.

Gradient-weighted Class Activation Mapping (Grad-CAM) offers feature learning insights and Kaggle data availability. However, it lacks recall and provides limited information on training parameters like epochs, learning rate, and optimization strategy.

* 1. **MobileNet v1**

Skin Lesion Analyzer: An Efficient Seven-Way Multi-class Skin Cancer Classification Using MobileNet [[17](#ref_mobilenetv1_paper)].

It will be published on 26 May 2020 and published by Saket S. Chaturvedi, Kajol Gupta & Prakash S. Prasad.

Dataset used is an open-access dataset called HAM10000, it is a benchmark dataset with over 50% of lesions confirmed by pathology. The dataset consists of a total of 10.015 dermoscopy images.

The dermoscopy images in the dataset were downscaled to 224 \* 224-pixel resolution from 600 \* 450-pixel resolution to make images compatible with MobileNet model.

There are 10.015 images in the dataset that were split into the training ser 9077 images & validation 938 images.

The MobileNet convolutional neural network was utilized. It was pretrained on a dataset of 1,280,000 images, which included 1,000 object classes from the 2014 ImageNet Challenge.

The model was trained on a training set containing 38,569 images.

The architecture used in the study consists of 25 layers, including four Conv2D layers, seven Batch Normalization layers, seven ReLU layers, three ZeroPadding2D layers, and single DepthwiseConv2D, Global Average Pooling, Dropout, and Dense layers (refer to Fig. 2 for a visual representation).

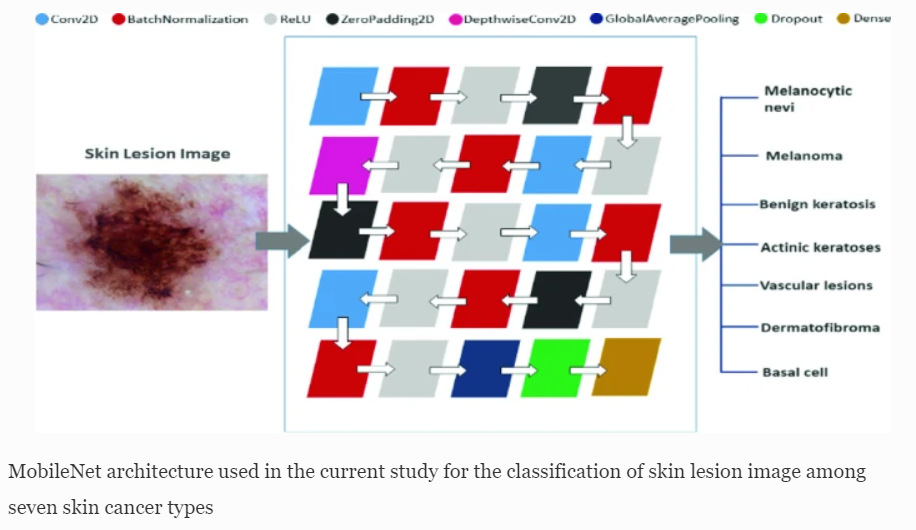


Fig.6 Architecture of MobileNetV1

The weighted average of precision, recall, and f1-score were found to be 89%, 83%, and 83%, respectively.

The model's performance was evaluated using various metrics, including accuracy, precision, recall, and F1-score.

The model achieved high precision and recall values for all seven skin lesion classes, indicating its ability to correctly classify different types of lesions.

Limited Dataset Size for Some Classes:

The ISIC 2019 dataset used for training and testing the model contains a limited number of images for some skin lesion classes, such as Dermatofibroma and Vascular Lesion. This imbalance in the dataset could potentially affect the model's performance on these classes.

Limited Clinical Expertise: The paper does not involve dermatologists or other medical experts in the evaluation of the model's performance. Clinical expertise is important to assess the model's ability to correctly classify skin lesions in a way that is meaningful for medical diagnosis.

Pros about its High accuracy: The proposed deep learning model achieved an overall accuracy of 91.3% on the ISIC 2019 test set, which is comparable to the performance of dermatologists.

Multi-class classification: The model can classify seven different types of skin lesions.

Cons about it No balance between images of diseases. This imbalance in the dataset could potentially affect the model's performance on these classes.

The model was trained on a dataset that may contain biases, such as overrepresentation of certain skin types or lesion characteristics. This could potentially lead to biased predictions, particularly for underrepresented groups.

* 1. **MobileNet v2 & LSTM**

Classification of Skin Disease Using Deep Learning Neural Networks with MobileNetV2 and LSTM [[18](#ref_mobilenetv2_lstm_paper)].

It will be published on 18 April 2021 by Parvathaneni NagaSrinivasu, Jalluri Gnana SivaSai, Mohammed Fazal Ijaz, Akash Kumar, Wonjoon Kim &James Jin Kang.

The HAM10000 dataset is used in this paper. This dataset contains more than 10,000 dermoscopy images.

The input size of the image is 224 × 224 × 3. The first two values (224× 224) indicate the height and width of the image. These values should always be greater than 32. The third value suggests that it has 3 input channels. The proposed architecture has 32 filters, and the filter size is 3 × 3 × 3 × 32.

A random (rand) function is applied to split the data into the training data (7224) and validation data (1255).

The main function of the paper is to bring in the state of art technique, namely the MobileNet V2 with LSTM component for the purpose of the precise classification of skin disease from the image that is captured from the mobile device.

MobileNet V2 is used in classifying the type of skin disease, and LSTM is used to enhance the performance of the model by maintaining the state information of the features that it comes across in the previous generation of the image classification.

The proposed MobileNet V2 with LSTM performance is evaluated through the hyperparameters like training and validation loss measures that determine the proposed model’s capabilities.

The MobileNet V2 architecture consists of a residual layer and a downsizing layer with a ReLu component. It includes three sub- layers: 1 × 1 convolution with ReLu6, Depth-Wise Convolution, and a 1 × 1 convolution layer without non-linearity. ReLu6 is used in the output domain for robustness and randomness. All layers have the same output channels, and a 3 × 3 filter is used for training.

A diagram of a complex structure

Description automatically generated

Fig.7 component of LSTM

A diagram of a flowchart

Description automatically generated

Fig.8 Architecture of MobileNetV2

Here is a general overview of the process:

Data Collection: Gather a dataset of skin images, the paper used the specific dataset used (HAM10000).

Preprocessing: preprocess the collected images to ensure they are standardized & suitable for training the model. This may involve resizing, cropping, normalizing pixel values & applying data augmentation techniques.

Model architecture: the paper mentions the use of a deep learning neural network based on MobileNet & LSTM. MobileNet V2 -> is a convolutional neural network architecture optimized for mobile and embedded devices. LSTM (Long Short-Term Memory) is a type of recurrent neural network that can capture temporal dependencies. including the number of layers, the configuration of the LSTM, and any additional components.

Training: Split the preprocessed dataset into training and validation sets.

Evaluation: Evaluate the trained model on the validation set to assess its performance.

Testing: Once the model is trained and evaluated, you can use it to classify new, unseen skin disease images.

The Fine-Tuned Neural Network-based skin disease classification model has achieved a reasonable accuracy of 89.90% for the validation set.

The model achieved high precision and recall values for all seven skin lesion classes, indicating its ability to correctly classify different types of lesions.

From limitations A random (rand) function is applied to split the data into the training data (7224) and validation data (1255).

The considered dataset is slightly imbalanced because some skin diseases are more, and some are less in number.

Pros that the main advantage of using the MobileNet architecture is that the model needs comparatively less computational effort than the conventional CNN model.

Cons that No balance between images of diseases. This imbalance in the dataset could potentially affect the model's performance on these classes.

* 1. **VGGNet**

Classification of skin cancer using VGGNet model structures [[19](#ref_vggnet_paper)].

It Published in 8/12/2022 by Volkan KAYA, İsmail AKGÜL

In this study, a dataset containing images of benign skin cancers and malignant skin cancers was use (Kaggle,2021).

In this dataset, 1800 benign skin cancer images with 224x224 pixel size and 1497 malignant skin cancer images are in RGB color format. A total of 3297 skin images are divided into training (80%), and test (20%).

There was no mention of image processing explicitly, but they said the VGGNet architecture takes an input image in RGB color format with a size of 224×224 pixels during the training and a 3×3 filter is used in each convolution layer.

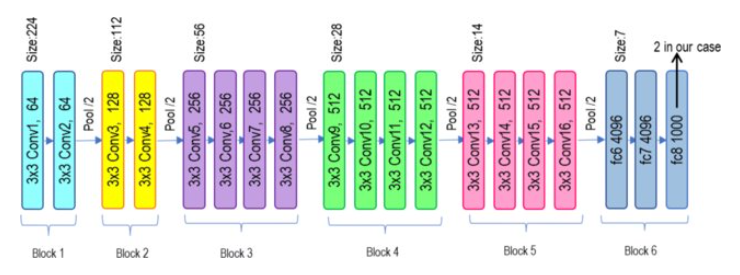


Fig.9 Architecture of VGG19

VGGNet is a convolutional neural network architecture with multi-layer GPU support, used for object recognition models. It uses RGB input images, 3x3 filters, and a 2x2 pooling layer to reduce data size.

In this study, VGG-11, VGG-13, VGG-16, and VGG-19 model structures for the training of these models, a dataset containing 1000 benign and 1000 malignant skin cancer types was used, and it was seen that the VGG-11 model had superior performance compared to other model structure.

VGGNet consists of several layers, including input, convolutional, pooling, fully connected, and output layers.

The specific number of layers in VGGNet depends on the model structure being used.

VGG-11: 18 layers

VGG-13: 20 layers

VGG-16: 23 layers

VGG-19: 26 layers

The models were implemented using the Python programming language, these model constructs were trained and tested using a dataset of 3297 skin cancer images.

The best classification success accuracy was obtained with the VGG-11 model structure, with 83 and 0.45 loss rates.

The model structure with the lowest classification success accuracy was 81 with VGG-19 and the loss rate was 0.37 As a result.

These model constructs were trained and tested using a dataset of 3297 skin cancer images. According to the training and test results, the best classification success accuracy was obtained with the VGG-11 model structure, with 83% and 0.45 loss rates. The model structure with the lowest classification success accuracy was 81% with VGG-19 and the loss rate was 0.37.

From Pros The deep learning methods used in VGGNet models, which analyze raw data and classify complex structures, have shown successful results in disease recognition and detection, including skin cancer.

-VGGNet model structures, such as VGG-11, VGG-13, VGG-16, and VGG-19, have shown good success in classifying skin cancer types with high accuracy.

From Cons The study did not provide detailed information about the specific architecture and training process of the VGGNet models, such as the number of layers and optimization techniques used.

-The dataset used in the study consisted of two classes of skin cancer, which may limit the generalizability of the models to datasets with multiple classifications.

* 1. **VGG16**

Identification and Classification of Skin Diseases using Deep Learning Techniques [[20](#ref_vgg16_paper)].

Posted by Venkatesh R Pai1, Soujanya G Pai, P M Suhasi and P M Rekha1 Date: March 1st, 2023.

The collected dataset consists of 5 classes namely, Impetigo (Bacterial), Cellulitis (Bacterial), Sporotrichosis (Fungal), Ringworm (Fungal), and Healthy Skin. This contains nearly 500 images where the frequency of each class is unequal.

Label Encoding and One-hot Coding The 5 class labels of the dataset are Cellulitis Bacteria, Healthy Skin, impetigo Bacteria, Ringworm Fungus, and Sporotrichosis Fungus. These class labels are converted to the integers 0,1,2,3, and 4 respectively.

Normalization Each pixel of an image is divided by 255 which places all the pixel values in the common range of [0,1]

Balancing Dataset For the image dataset in consideration, the Class Weight method is performed. If class weights are set to 'balanced', the model automatically adjusts the class weights inversely proportional to the frequency of each class.

Image Augmentation It reduces overfitting and increases the ability of the CNN model to generalize while extracting the features from the images.

Augmentation techniques such as width shift, height shift, zooming, and vertical, and horizontal flips are performed.

The paper proposes a methodology that utilizes deep learning techniques, specifically a Convolutional Neural Network (CNN) with the fine-tuned version of the VGG16 architecture, for identifying and classifying skin diseases caused by bacteria and fungi using non-dermoscopic images.

-The dataset used in the study consists of five classes: Cellulitis, Impetigo (bacterial skin diseases), Ringworm, Sporotrichosis (fungal skin diseases), and Healthy Skin.

VGG16 consists of 21 layers but only 16 layers of those are trainable. The architecture includes 13 convolutional layers, 5 max-pooling layers, and 3 fully connected layers. Several small filters are used throughout the network with max-pooling operations being performed periodically. Instead of using large filters, VGG16 architecture considers smaller filters with more depth in the network.

The system adopts a Convolutional Neural Network (CNN) as the deep learning algorithm for image processing applications, The system makes use of the fine-tuned version of the VGG16 architecture, which is a type of CNN, to leverage the power of transfer learning.

The system trains the CNN model on the preprocessed images and evaluates its performance.

A diagram of a computer program

Description automatically generated

Fig.10 Architecture of VGG16

The developed model is tested with the testing set images that consisted of 97 images. The model achieved a test accuracy of 86% and an F1-score of 85%.

From Pros Deep learning techniques, specifically Convolutional Neural Networks (CNN), are known to be highly effective for image processing tasks, making them suitable for identifying and classifying skin diseases.

The use of transfer learning, specifically the fine-tuned version of the VGG16 architecture, leverages pre-trained models and can improve the accuracy and performance of the system.

From Cons The system relies on non-dermoscopic images, which may limit its effectiveness In accurately diagnosing certain skin diseases that require dermoscopic images for accurate analysis.

The dataset used for training the model consists of only five classes of skin diseases, which may not cover the full spectrum of skin conditions, potentially limiting the model's ability to accurately classify less common or rare skin diseases.

* 1. **Comparison table**

**Table 3.1** Algorithms Comparison



**Table 3.4** Algorithms Comparison (continued)



**Table 3.2** Algorithms Comparison (continued)

* 1. **Conclusion**

After extensive research, we based our algorithm selection for the project on several key factors outlined in previous papers. These factors included the attributes of the dataset and the performance metrics of various models.

Our dataset selection process prioritized experiments conducted with DenseNet-121. We chose this model based on its demonstrated capacity to capture intricate features and effectively manage complex image data. Notably, they consistently achieved high levels of accuracy, which was deemed satisfactory for our project's objectives.

We chose DenseNet-121 because it showed high efficiency with an accuracy of up to 99% in the process of classifying between malignant and benign skin diseases. Therefore, we decided to use it to detect more types of skin diseases and not just classify between two types (malignant and benign).

**Proposed Model**

* 1. **Introduction**

The proposed model for skin disease detection is based on a CNN architecture designed to classify various types of skin diseases from dermatoscopic images. The motivation behind choosing CNN is its proven effectiveness in image recognition tasks. We try to achieve more inspiring results and contribute to the development of skin disease classification applications using deep learning.

* 1. **Dataset**

Description of the Dataset:

The dataset used in this project is the ISIC 2019 dataset, a comprehensive collection of 25,352 dermatoscopic images categorized into eight different skin lesion classes. The dataset is a resource for developing and evaluating skin diseases detection algorithms and conducting dermatological research. It was published on December 17, 2018, by a group of researchers including Tschandl, Codella and others.

Dataset Composition:

The dataset is divided into three subsets:

Training Set: 21,512 images (84.8%)

Validation Set: 1,910 images (7.5%)

Test Set: 1,930 images (7.6%)

The images in the dataset vary in size, predominantly 600 × 450 pixels and 1024 × 1024 pixels and are stored in JPG format. These images were sourced from a variety of global healthcare providers, clinics, and research centers specializing in dermatology.

Focused Skin Diseases:

For the scope of this project, we focus on four specific skin diseases:

|  |  |  |  |
| --- | --- | --- | --- |
| **Skin Disease** | **Training Images** | **Validation Images** | **Test Images** |
| **Actinic Keratosis** | 944 | 76 | 75 |
| **Dermatofibroma** | 420 | 22 | 43 |
| **Melanocytic Nevus** | 4428 | 931 | 423 |
| **Vascular Lesion** | 347 | 27 | 24 |

Table 3 Skin Diseases That Have Been Worked On

Data Preprocessing and Augmentation:

Given the unbalanced distribution across labels, data augmentation techniques are crucial to enhance the variability and robustness of the model.

The following augmentation techniques were applied to the training dataset and validation dataset

|  |  |
| --- | --- |
| **Augmentation Technique** | **Parameter** |
| **Rescaling** | 1.0/255 |
| **Rotation** | rotation\_range=40 |
| **Width Shift** | width\_shift\_range=0.2 |
| **Height Shift** | height\_shift\_range=0.2 |
| **Shear** | shear\_range=0.2 |
| **Zoom** | zoom\_range=0.2 |
| **Horizontal Flip** | horizontal\_flip=True |
| **Fill Mode** | fill\_mode=nearest |
| **Feature-wise Standard Normalization** | featurewise\_std\_normalization=True |

Table 4 Data Augmentations

Additionally, the dull razor technique was employed for hair removal to enhance image clarity and focus on the lesion area. All images were resized to 224 × 224 pixels to maintain uniformity and compatibility with the model input requirements.

* 1. **Model Architecture**

The model architecture consists of a base DenseNet121 model with custom convolutional, pooling, dropout, and fully connected layers tailored for skin disease classification. Batch Normalization and dropout layers were used for regularization and improved training performance.

The base model is DenseNet121, loaded with pre-trained ImageNet weights. It includes convolutional blocks with skip connections, facilitating feature reuse and gradient flow.

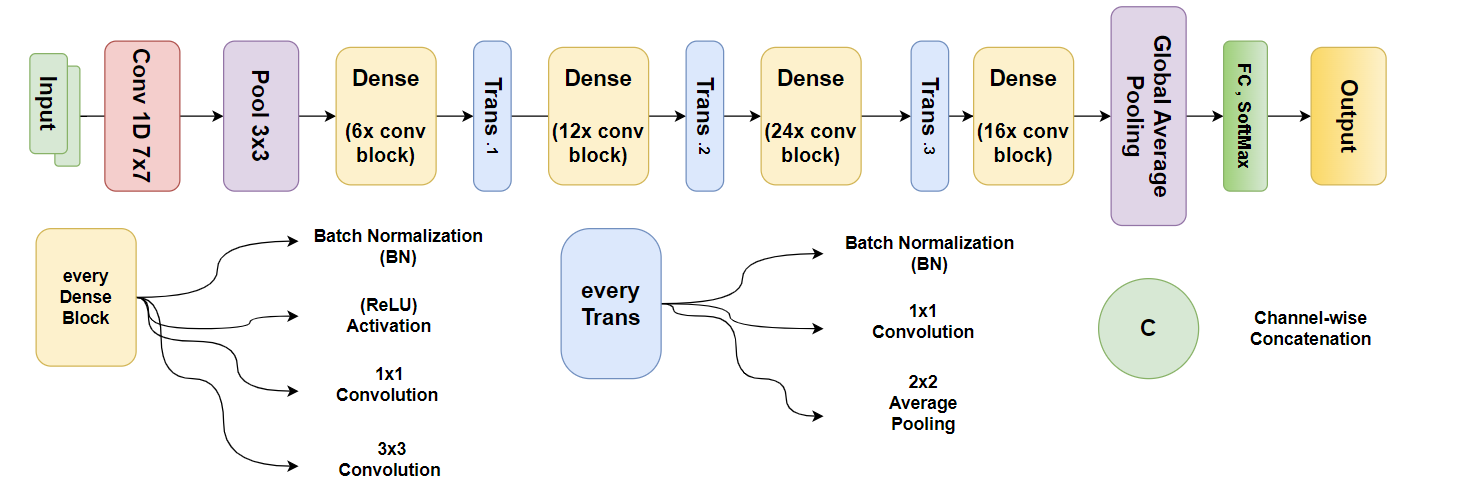


Fig11 DenseNet-121 architecture

Custom layers were added on top of the base model to adapt it to the specific task of skin disease classification.

A diagram of a program

Description automatically generated

Fig12 Custom layers architecture

Convolutional Layers:

Layer 1: 256 filters, (3, 3) kernel size, ReLU activation, padding='same'

Layer 2: 128 filters, (3, 3) kernel size, ReLU activation, padding='same'

Layer 3: 64 filters, (3, 3) kernel size, ReLU activation, padding='same'

Layer 4: 32 filters, (3, 3) kernel size, ReLU activation, padding='same'

Pooling Layers:

Layer 1: Max pooling with (2, 2) pool size, padding='same'

Layer 2: Max pooling with (2, 2) pool size, padding='same'

Layer 3: Max pooling with (2, 2) pool size, padding='same'

Dropout Layers:

Dropout rate of 0.3 after Layer 1

Dropout rate of 0.2 after Layer 2 and Layer 3 and Layer 4

Flatten Layer:

Flatten the output of the convolutional layers to prepare for fully connected layers.

Fully Connected Layers:

Layer 1: 512 units, ReLU activation, with BatchNormalization and dropout rate of 0.5

Layer 2: 256 units, ReLU activation, with BatchNormalization and dropout rate of 0.3

Output Layer:

Dense layer with SoftMax activation for multi-class classification.

* 1. **Training Procedure**

The model was trained on a system with the following specifications:

RAM: 16 GB, GPU: NVIDIA GTX 1650 Ti, CPU: Intel Core i7-10750H, Framework: TensorFlow 2.10, Python Version: 3.10, CUDA Toolkit: 11.2, cudnn: 8.1.0, Package Manager: Anaconda 2.5.2

The hyperparameters used for training the model are as follows:

Epochs: 100, Batch Size: 100, Learning Rate: 0.001, Optimizer: Adam, Loss Function: Categorical Cross-Entropy

To evaluate the performance of the model, the following metrics were used:

Accuracy: Measures the overall correctness of the model's predictions.

Precision: Indicates the proportion of true positive results among all positive predictions.

Recall: Represents the proportion of true positive results among all actual positive cases.

F1-Score: The harmonic means of precision and recall, providing a balance between the two.

These metrics were chosen to provide a comprehensive evaluation of the model’s performance, ensuring that both the accuracy and the quality of the predictions are considered.

**Experimental Results**

* 1. **Main experiment**

This experiment was carried out using densenet121, and Cubic SVM was combined with it, as mentioned in the proposed model chapter, and the evaluation matrix contains accuracy, precision, Recall, F1-Score

Accuracy measures the overall correctness of the model by comparing the number of correct predictions to the total number of predictions.

Precision measures the proportion of positive identifications that were actually correct.

Recall measures the proportion of actual positives that were correctly identified.

Since Precision and Recall can be defined as above, substituting them into the F1-Score equation gives:

The table below presents a comprehensive evaluation of our classification model's performance across four types of skin lesions: Actinic Keratosis, Dermatofibroma, Melanocytic Nevus, and Vascular Lesion. This table includes key metrics such as Precision, Recall, F1-Score, and Support, providing a detailed assessment of how well the model predicts each class.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Precision** | **Recall** | **F1-Score** | **Support** |
| Actinic keratosis | 0.97 | 0.89 | 0.93 | 75 |
| Dermatofibroma | 0.85 | 0.95 | 0.9 | 43 |
| Melanocytic nevus | 0.99 | 0.98 | 0.98 | 423 |
| Vascular lesion | 0.8 | 1 | 0.89 | 24 |
| **Accuracy** |  |  | **0.96** | 565 |
| **Macro avg** | 0.9 | 0.96 | 0.93 | 565 |
| **Weighted avg** | 0.97 | 0.96 | 0.97 | 565 |

Table 5 Model Evaluation Metrics of densenet121

Overall, the model achieves an impressive accuracy of 0.96. The macro average and weighted average metrics also indicate strong and consistent performance across all classes, validating the robustness of our classification model. These metrics are crucial for understanding the strengths and potential areas for improvement in our model's predictions.

The integration of a **Cubic SVM** classifier with DenseNet121 further enhanced the accuracy, achieving a test accuracy of **0.99** in the best experimental setup.

The below image represents the confusion matrix for our classification model, which was trained to identify four different types of skin lesions: Actinic Keratosis, Dermatofibroma, Melanocytic Nevus, and Vascular Lesion. A confusion matrix is a useful tool for evaluating the performance of a classification algorithm. It provides a detailed breakdown of correct and incorrect predictions, allowing us to see how well the model distinguishes between different classes.

A graph of confusion matrix

Description automatically generated

Fig 13 Confusion matrix of densenet121

* 1. **Experiments comparisons**

We have done many experiments where we adjust each hyperparameter independently just until we get the best result for the model containing that hyperparameter, and then we fixed the best result for that hyperparameter and change another hyperparameter until we get the best result and so on with all hyperparameters such as target size, unfreeze layers ,epochs, batch size, learning rate, dropout, early stopping, reduced learning rate, loss function, and optimizer. Therefore, we obtained very many results and experiments, so we presented a comparison between the best of these results in a way that shows the sequence in development.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date** | **2-Apr** | **3-May** | **15-Apr using Cubic SVM method** | **30-Apr Cubic SVM method**  **Under sampling** |
| **Epochs** | 10 | 15 | 100 | 100 |
| **Batch Size** | 32 | 64 | 100 | 100 |
| **Target Size** | 224,224 | 250,250 | 224,224 | 224,224 |
| **Reduce Learning Rate** | No | Yes (factor=0.1, patience=15) | No | No |
| **Unfreeze Layers** | 30 top layers | 30 top layers | 30 top layers | 30 top layers |
| **Early Stop** | No | No | (monitor='loss', mode='min', patience=10) | (monitor='loss', mode='min', patience=10) |
| **Dropout** | 0.3 | 0.5 | 0.5 | 0.5 | 0.5 | 0.3 | 0.5 | 0.3 |
| **Test Accuracy** | 0.86 | 0.914 | 0.959 | **0.964** |
| **Test Loss** | 0.717 | 0.422 | 0.137 | 0.127 |
| **Cubic SVM Test Accuracy** | No | No | 0.9 | 0.99 |
| **Cubic SVM Test Loss** | No | No | 0.09999 | 0.01 |

Table 6 Experiments Comparisons

**Web development**

* 1. **Front-End Development**

Our skin disease classification web application features a thoughtfully designed front-end that prioritizes user experience, information delivery, and responsiveness. This section provides an overview of the structure, technologies, libraries, and methodologies used in developing our front-end.

The website opens with a Header section that includes the website logo and a navigation bar with links to Home, Discover, App, and About Us sections. This layout ensures easy navigation and a cohesive user experience right from the start.

Following the header, the Intro Section engages users by introducing the application's purpose and value. This section aims to immediately capture the user's interest and provide a clear understanding of what the app offers and how it can benefit them.

Next, the Discover Dermatology section educates users about the various skin diseases our app can classify. It provides brief and informative descriptions of each condition, helping users gain a better understanding of their potential diagnoses and the importance of dermatological health.

The Consultation Section is designed to facilitate user interaction with dermatological experts. It provides a platform for users to seek more detailed information and professional advice about their skin conditions, enhancing the overall value of the application.

In the App Section, users can upload images of their skin conditions via an upload button or drag-and-drop functionality. Once an image is uploaded, the app processes it to classify the disease, offering a diagnosis along with treatment advice. This section is the core functionality of our application, providing users with immediate and actionable insights into their skin health.

The About Us section provides information about the development team, highlighting the expertise and efforts behind the project. It gives users a glimpse into the people responsible for the app, fostering trust and transparency.

Following this is the Acknowledgment Section, where we express our gratitude to the engineering and medical supervisors who contributed to the project. This section is a testament to the collaborative efforts that made the application possible.

Finally, the Footer mirrors the header with the navigation bar and logo, ensuring consistency and easy access to other sections of the website. It helps users navigate the site seamlessly and ensures that important links are always within reach.

The website is built using React v18, a powerful framework that allows us to create a dynamic and responsive user interface. This ensures that the website performs optimally across various devices, from desktops to mobile phones.

To enhance functionality, we employed several libraries. Axios is used for making API requests to the backend, efficiently handling responses and errors. React-Dropzone provides a seamless drag-and-drop interface for image uploads, enhancing the user experience in the App section. React-Slideshow is utilized to create engaging and interactive slideshows, adding a visually appealing element to the site.

In terms of tools, we used ESLint for code linting, which helps maintain high code quality and consistency by identifying and correcting issues in JavaScript code. Vite was employed for module management, significantly improving the development experience by providing a fast and efficient build tool.

Our data flow is designed to ensure smooth interaction between the front-end and backend. Using Axios, the front-end fetches data from the backend API, handling responses and errors to provide a seamless user experience. State management is another critical aspect, where the application state is efficiently updated to reflect changes, ensuring that the user interface remains dynamic and responsive.

Overall, our front-end development strategy focuses on creating a robust, user-friendly, and responsive web application. By leveraging modern technologies, libraries, and tools, we have developed an interface that not only meets the functional requirements but also provides an engaging and informative user experience.

* 1. **Back-End Development**

Our back-end development is focused on providing a robust and efficient server that handles image classification requests for skin diseases. The back-end is developed using Flask, a lightweight web framework for Python, specifically version 3.0.3. This framework allows us to create web servers and APIs with minimal setup and code, making it an ideal choice for our application.

We start by importing the necessary libraries. Flask is used for creating the web server, flask\_cors for handling Cross-Origin Resource Sharing (CORS) to allow our front-end to communicate with the back-end, and TensorFlow for loading and using our pre-trained machine learning model. Additionally, we use the Keras API, which is part of TensorFlow, for image preprocessing and model predictions.

The Flask application is initialized, and CORS is enabled to allow requests from different origins. This setup is crucial for enabling seamless interaction between our front-end (which might be hosted on a different domain) and the back-end.

We then load our pre-trained DenseNet model, which is stored in the 'densenet\_svc\_model.h5' file. This model is designed to classify skin diseases based on the images provided by the users. The model was trained on a dataset of skin disease images and can predict one of several classes: Actinic keratosis, Dermatofibroma, Melanocytic nevus, and Vascular lesion.

To provide a simple interface for users and ensure that our server is working, we define a home route (/) that returns a greeting message when accessed. This route serves as a basic check to confirm that the server is running correctly.

The core functionality of our back-end lies in the (/upload) route, which is set up to handle POST requests. When a user uploads an image through this endpoint, the server first checks if the file is present in the request. If not, it returns an error message. Once a file is detected, the image data is read and processed. The image is loaded and resized to 224x224 pixels to match the input size expected by our model. It is then converted to an array, expanded in dimensions, and normalized by dividing pixel values by 255.

The processed image is fed into the DenseNet121 model to make a prediction. The model outputs a set of probabilities corresponding to each class, and the class with the highest probability is selected as the predicted class. This class is then mapped to its corresponding label from the class\_labels list.

Finally, the server responds with a JSON object containing the predicted label. This response is sent back to the front-end, which can then display the result to the user.

Overall, our back-end leverages Flask to create a responsive and efficient server that integrates a machine learning model for classifying skin diseases, handling user uploads, and returning predictions in real-time.

* 1. **Deployment**

In this section, we outline the deployment process for our skin disease classification web application. The deployment strategy ensures that our application is accessible, scalable, and performs efficiently.

For deploying our application, we utilized DigitalOcean, a cloud infrastructure provider renowned for its simplicity and scalability. DigitalOcean offers virtual servers known as Droplets, which are highly configurable and can be tailored to meet specific needs. For our project, we chose DigitalOcean’s Basic Droplets, which provided us with 1 vCPU, 1GB RAM, and 35GB of storage. These resources are sufficient for our initial deployment and allow for easy scaling as the application's user base grows. The Droplets provided a stable and robust environment for running our application, ensuring that it performs efficiently and remains responsive under load.

To host our React application and manage HTTP requests efficiently, we employed Nginx as our web server. Nginx is known for its high performance, stability, and low resource consumption, making it an ideal choice for serving our application. It handles incoming traffic, serves static files, and proxies API requests to the Flask back-end seamlessly.

The deployment process began with setting up a new Droplet on DigitalOcean, selecting the Basic Droplet configuration to meet our requirements. After accessing the Droplet via SSH, we installed the necessary software packages, including Nginx, Python, and required Python libraries. This initial setup provided the foundational environment required to run both our front-end and back-end applications.

The React application was built using the build process, generating static files that were then moved to the appropriate directory for Nginx to serve. Concurrently, the Flask back-end application, along with the machine learning model, was deployed and configured to run as a service. This ensures that the back-end remains active and responsive to incoming API requests, providing the necessary functionality for our application.

Configuring Nginx to serve the React application involved setting it up to handle static files and proxy API requests to the Flask back-end.

By leveraging DigitalOcean’s scalable infrastructure and Nginx’s efficient web serving capabilities, we have deployed a robust and scalable web application. This setup not only ensures high availability and performance but also provides a foundation for future growth and enhancements.

**Conclusion and Future work**

* 1. **Conclusion**

In conclusion, we explored key skin disease datasets, including HAM10000, ISIC 2019, PAD-UFES-20, BCN20000, Derm7pt, ISIC 2017, DermNet, PH2, and Dermofit, and provided valuable insights into their structures and characteristics, and examine their limitations and proposed solutions aim to enhance the quality and reliability of these datasets, laying a foundation for their optimal utilization in decision-making processes and applications dependent on accurate data. Building on our exploration, our project will rely on the ISIC 2019 dataset.

In our analysis of deep learning models such as ResNet-152, ResNet101, Inception-v3, DenseNet-121, DenseNet-201, MobileNet v1, MobileNet v2 & LSTM, VGGNet, and VGG16, we have gained a comprehensive understanding of their effectiveness in skin disease classification. By delving into their origins, dataset usage, preprocessing techniques, structures, layers, model-building processes, and result metrics, we are equipped with valuable knowledge for implementing robust deep learning model. Building on our exploration we chose DenseNet-121 algorithm based on their demonstrated capacity to capture intricate features and effectively manage complex image data.

The proposed CNN-based model for skin disease detection has demonstrated significant effectiveness in classifying various skin lesion types from dermatoscopic images. Leveraging the ISIC 2019 dataset, the model was trained and evaluated using a robust set of metrics, including accuracy, precision, recall, and F1-score, across four specific skin diseases: Actinic Keratosis, Dermatofibroma, Melanocytic Nevus, and Vascular Lesion.

Key highlights from our experimental results include:

* The model achieved a high overall accuracy of 0.96, indicating its strong capability in correctly classifying the skin lesion images.
* High precision and recall metrics across all classes, with particularly notable performance for Melanocytic Nevus (Precision: 0.99, Recall: 0.98, F1-Score: 0.98), underscoring the model's reliability in identifying the most common lesion in the dataset.
* Consistent macro and weighted average metrics validate the model's robustness and balanced performance across different classes, highlighting its potential for real-world dermatological applications.

To enhance the model's performance, extensive data preprocessing and augmentation techniques were employed to address the imbalance in the dataset and improve the model's generalization capabilities. The application of techniques such as rescaling, rotation, width and height shifts, shear, zoom, horizontal flip, and feature-wise standard normalization contributed to the model's high accuracy and robustness.

Furthermore, the model's architecture, based on DenseNet121 with custom convolutional, pooling, dropout, and fully connected layers, was carefully designed to optimize feature extraction and classification performance. The inclusion of pre-trained ImageNet weights and the strategic addition of batch normalization and dropout layers facilitated efficient training and regularization.

Experimental comparisons revealed that fine-tuning hyperparameters such as epochs, target size, learning rate, dropout rates, and early stopping criteria significantly impacted the model's performance. The integration of a Cubic SVM classifier further enhanced the accuracy, achieving a test accuracy of 0.99 in the best experimental setup.

In summary, the proposed model exhibits excellent potential for accurate skin disease classification, paving the way for advanced dermatological diagnostic tools. The thorough evaluation and iterative optimization underscore the model's capability and readiness for practical deployment in clinical settings. Future work will focus on expanding the dataset to include more diverse skin lesions and exploring additional augmentation and optimization techniques to further improve model performance.

* 1. **Future work**

In the next phases of development for our skin disease classification web application, we aim to enhance the platform's capabilities and extend its reach to a broader audience. Here are some key areas we plan to focus on:

Expanding Service Reach: We aim to increase the number of dermatologists available on our platform and expand the service to new regions. By doing so, we can provide more comprehensive coverage and ensure that patients from diverse locations have access to expert dermatological care. This expansion will involve targeted outreach and partnerships with medical institutions to onboard qualified dermatologists.

User Profile Management: To improve user experience and ensure continuity of care, we plan to introduce user profiles. This feature will allow users to keep track of their personal data, medical history, and previous diagnoses. By maintaining detailed medical records, users and doctors can access a patient's history seamlessly, facilitating more accurate diagnoses and personalized treatment plans.

Categorize Users Groups: We will enhance our platform by creating distinct categories for users, specifically dividing membership into patients and doctors. This differentiation will enable us to tailor the user interface and functionalities to meet the specific needs of each group. For patients, this includes simplified appointment scheduling and access to their medical history, while for doctors, it includes tools for patient management and collaboration with peers.

Research and Publication: Given our achievement of an accuracy rate of up to 0.99 in skin disease classification, we will prepare a comprehensive research paper detailing our methodologies and findings. This paper will contribute to the academic and medical communities by presenting new results in the field of dermatological diagnostics. Publishing our research will not only validate our work but also provide a foundation for future advancements and collaborations in medical AI.

By focusing on these areas, we aim to not only improve our current service but also contribute meaningfully to the field of dermatology through innovation and research. Our commitment to enhancing user experience, expanding access to care, and advancing medical knowledge underpins our future development goals.

**References**

1. Asiedu, K. B. Retrieved, WHO's inaugural global meeting on Neglected Tropical Diseases affecting the skin urges intensified efforts to address their impact, World Health Organization. (2023, March 31), <https://www.who.int/news/item/31-03-2023-who-first-global-meeting-on-skin-ntds-calls-for-greater-efforts-to-address-their-burden>
2. Tschandl, P., Rosendahl, C., & Kittler, H. (2018). The HAM10000 dataset, a large collection of multi-sources dermatoscopic images of common pigmented skin lesions, Scientific Data, (2018) <https://doi.org/10.1038/sdata.2018.161>
3. Skin Cancer MNIST: HAM10000, Kaggle, (2018, September 20). <https://www.kaggle.com/datasets/kmader/skin-cancer-mnist-ham10000>
4. International Skin Imaging Collaboration. (n.d.), ISIC, <https://www.isic-archive.com/>
5. Pacheco, A. G. C., Lima, G. R., Salomão, A. S., Krohling, B. A., Biral, I. P., De Angelo, G. G., Alves, F. C. R., Esgario, J. G. M., Simora, A. C., Castro, P. B. C., Rodrigues, F. B., Frasson, P., Krohling, R. A., Knidel, H., Santos, M. C. S., Santo, R. B. D. E., Macedo, T. L. S. G., Canuto, T. R. P., & De Barros, L. F. S. PAD-UFES-20: A skin lesion dataset composed of patient data and clinical images collected from smartphones. Data in Brief, 32, 106221,(2020), <https://doi.org/10.1016/j.dib.2020.106221>
6. Combalia, M. BCN20000: Dermoscopic lesions in the wild, arXiv.org, (2019, August 6), <https://arxiv.org/abs/1908.02288>
7. J. Kawahara, S. Daneshvar, G. Argenziano and G. Hamarneh, "Seven-Point Checklist and Skin Lesion Classification Using Multitask Multimodal Neural Nets," in *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 2, pp. 538-546, March 2019, <https://doi.org/10.1109/JBHI.2018.2824327>.
8. Codella, N. C. F., Gutman, D., Celebi, M. E., Helba, B., Marchetti, M. A., Dusza, S. W., Kalloo, A., Liopyris, K., Mishra, N., Kittler, H., & Halpern, A. Skin Lesion Analysis Toward Melanoma Detection: A Challenge at the 2017 International Symposium on Biomedical Imaging (ISBI), Hosted by the International Skin Imaging Collaboration (ISIC), ArXiv:1710.05006 [Cs], (2018), <https://arxiv.org/abs/1710.05006>
9. Dr Ian Coulson. DermNet. (2021). <https://dermnetnz.org/>
10. Mendonca, T., Ferreira, P. M., Marques, J. S., Marcal, A. R. S., & Rozeira, J. PH2 - A dermoscopic image database for research and benchmarking. 2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), (2013), <https://doi.org/10.1109/embc.2013.6610779>
11. ADDI - Automatic computer-based Diagnosis system for Dermoscopy Images, (2013), <https://www.fc.up.pt/addi/ph2%20database.html>
12. Rees, Aldridge, Fisher, Ballerini, A Color and Texture Based Hierarchical K-NN Approach to the Classification of Non-melanoma Skin Lesions, Color Medical Image Analysis, Lecture Notes in Computational Vision and Biomechanics 6 (M. E. Celebi, G. Schaefer (eds.)), (2013) <https://licensing.edinburgh-innovations.ed.ac.uk/product/dermofitimage-library>
13. Mendes, D. B. Skin lesions classification using convolutional neural networks in clinical images. arXiv.org, (2018, December 6), <https://arxiv.org/abs/1812.02316>
14. Demir, A., Yilmaz, F., & Kose, O. Early detection of skin cancer using deep learning architectures: resnet-101 and inception-v3, TIPTEKNO 2019, <https://doi.org/10.1109/tiptekno47231.2019.8972045>
15. Al-Saedi̇, D. K. A., & Savaş, S. Classification of Skin Cancer with Deep Transfer Learning Method, Computer Science, (2022), <https://doi.org/10.53070/bbd.1172782>
16. Rehman, M. Z. U., Ahmed, F., Alsuhibany, S. A., Jamal, S. S., Ali, M. Z., & Ahmad, J. Classification of skin cancer lesions using explainable deep learning, Sensors, 22(18), 6915, (2022), <https://doi.org/10.3390/s22186915>
17. Chaturvedi, S. S., Gupta, K., & Prasad, P. S. Skin lesion analyzer: An efficient Seven-Way multi-class skin cancer classification using MobileNet, In Advances in intelligent systems and computing (pp. 165–176),(2020), <https://doi.org/10.1007/978-981-15-3383-9_15>
18. Srinivasu, P. N., SivaSai, J. G., Ijaz, M. F., Bhoi, A. K., Kim, W., & Kang, J. J. Classification of Skin Disease Using Deep Learning Neural Networks with MobileNet V2 and LSTM, Sensors, 21(8), 2852, (2021), <https://doi.org/10.3390/s21082852>.
19. Kaya, V., & Akgül, İ. Classification of skin cancer using VGGNet model structures. Gümüşhane Üniversitesi Fen Bilimleri Enstitüsü Dergisi, (2022), <https://doi.org/10.17714/gumusfenbil.1069894>
20. Pai, V. R., Pai, S. G., Suhasi, P. M., & Rekha, P. M. Identification and Classification of Skin Diseases using Deep Learning Techniques, Research Square (Research Square), (2023), <https://doi.org/10.21203/rs.3.rs-2628782/v1>.

**Appendices**

* 1. **Code**
     1. **Front-End code**

Index.htmlA computer screen with text

Description automatically generated

A screen shot of a computer code

Description automatically generated

Main.jsx

A screen shot of a computer code

Description automatically generated

App.jsx

A screen shot of a computer

Description automatically generated

Header.jsx

A computer screen shot of a program code

Description automatically generated

A screen shot of a computer program

Description automatically generated

Logo.jsx

A screen shot of a computer code

Description automatically generated

PageNav.jsx

A screen shot of a computer program

Description automatically generated

BurgerBar.jsx

A screen shot of a computer program

Description automatically generated

HomeSection.jsx

A screen shot of a computer program

Description automatically generated

DiseasesSection.jsx

A screenshot of a computer program

Description automatically generated

A screen shot of a computer screen

Description automatically generated

ConsultSection.jsx

A screen shot of a computer program

Description automatically generated

AppSection.jsx

A screen shot of a computer program

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screen shot of a computer program

Description automatically generated



A screen shot of a computer program

Description automatically generated

ResultSection.jsx

A screenshot of a computer program

Description automatically generated

A screen shot of a computer program

Description automatically generated

A screen shot of a computer program

Description automatically generated

AboutUsSection.jsx

A screen shot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screen shot of a computer program

Description automatically generated

A screen shot of a computer program

Description automatically generated

Acknowledge.jsx

A screen shot of a computer screen

Description automatically generated

Footer.jsx

A computer screen shot of a program code

Description automatically generated

* + 1. **Back-End code**

**A screen shot of a computer

Description automatically generated**

**A computer screen shot of a program code

Description automatically generated**

* + 1. **AI code**

**A screen shot of a computer program

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A black rectangular object with a white border

Description automatically generated**

**A screenshot of a black screen

Description automatically generated**

**A screen shot of a computer

Description automatically generated**

**A screen shot of a computer code

Description automatically generated**

**A screen shot of a computer program

Description automatically generated**

**A screen shot of a computer program

Description automatically generated**

**A screen shot of a computer program

Description automatically generated**

**A screen shot of a computer program

Description automatically generated**

**A black and white rectangular object with a white line

Description automatically generated**

**A screenshot of a computer program

Description automatically generated**

**A screen shot of a computer program

Description automatically generated**

**A screen shot of a computer program

Description automatically generated**

**A screenshot of a computer program

Description automatically generated**

**A screen shot of a computer program

Description automatically generated**

**A computer screen shot of a program code

Description automatically generated**

**A screenshot of a computer program

Description automatically generated**

**A black background with yellow and green text

Description automatically generated**

**A screenshot of a computer program

Description automatically generated**

**A screenshot of a computer program

Description automatically generated**

**A screen shot of a computer program

Description automatically generated**

* 1. **Images**

**A person's face with a picture of her

Description automatically generated**

**A close-up of a skin with a mole

Description automatically generated**

**A blue screen with white text

Description automatically generated**

**A screenshot of a computer screen

Description automatically generated**

**A close-up of a person's skin

Description automatically generated**

**A person in a suit

Description automatically generated**

**A screenshot of a computer

Description automatically generated**