# DermaAI: Revolutionizing Skin Health with Automated Skin Disease Diagnosis and Care

*Project Report submitted by*


**ADITYA RAJ**
(4NM20CS016)

**RESHAL ANNE MENDONCA**
(4NM20CS145)

**MOHD AZHAR USMANI**
(4NM20CS108)

**SHAINA JYOTHICA CRASTA**
(4NM20CS160)


*Under the Guidance of*


**Dr. JYOTHI SHETTY**
**Professor & Head**
**Department of CSE, NMAMIT**


*In partial fulfillment of the requirements for the award of*

*the Degree of*

**Bachelor of Engineering in Computer Science and Engineering**

*from*

***Visvesvaraya Technological University, Belagavi***


Department of Computer Science and Engineering

NMAM Institute of Technology, Nitte - 574110

(An Autonomous Institution affiliated to VTU, Belagavi)


**APRIL 2024**

N.M.A.M. INSTITUTE OF TECHNOLOGY
(An Autonomous Institution affiliated to Visvesvaraya Technological University, Belagavi)
Nitte – 574 110, Karnataka, India

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

# CERTIFICATE

*Certified that the project work entitled*

***" DermaAI: Revolutionizing Skin Health with Automated Skin Disease Diagnosis and Care"***

*is a bonafide work carried out by*

***Aditya Raj (4NM20CS016)*** ***Reshal Anne Mendonca (4NM20CS145)***

***Mohd Azhar Usmani (4NM20CS108)*** ***Shaina Jyothica Crasta (4NM20CS160)***

*in partial fulfilment of the requirements for the award of*

***Bachelor of Engineering Degree in Computer Science and Engineering***

*prescribed by* ***Visvesvaraya Technological University, Belagavi***

*during the year 2023-2024.*

*It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library.*

*The project report has been approved as it satisfies the academic requirements in respect of the project work prescribed for the Bachelor of Engineering Degree.*

| Jyothi Shetty | Jyothi Shetty | Niranjan N.C. |
|---|---|---|
| **Signature of the Guide** | **Signature of the HOD** | **Signature of the Principal** |

**Semester End Viva Voce Examination**

| **Name of the Examiners** | **Signature with Date** |
|---|---|
| 1. Dr. Jyothi Shetty | Jyothi Shetty 18/4/2024 |
| 2. RAJASHREE | Rshmy 18/4/24. |

# ACKNOWLEDGEMENT

The satisfaction that accompanies the completion of any task would be incomplete without the mention of all the people, without whom this endeavour would have been a difficult one to achieve. Their constant blessings, encouragement, guidance, and suggestions have been a constant source of inspiration.

First and foremost, my gratitude to my project guide **Dr. Jyothi Shetty** for the constant guidance throughout the course of this Project and for the valuable suggestions.

I also take this opportunity to express a deep sense of gratitude to the project coordinators for their valuable guidance and support.

My thanks to our beloved HOD, **Dr. Jyothi Shetty,** for extending support in carrying out this project in the department and providing us with all the necessary facilities.

My sincere thanks to our beloved principal, **Dr. Niranjan N Chiplunkar** for permitting us to carry out this project at our college and providing us with all the necessary facilities.

Finally, thanks to the staff members of the Department of Computer Science and Engineering and our friends for their honest opinions and suggestions throughout the course of our project.

Aditya Raj (4NM20CS016)
Mohd Azhar Usmani (4NM20CS108)
Reshal Anne Mendonca (4NM20CS145)
Shaina Jyothica Crasta (4NM20CS160)

# ABSTRACT

DermaAI is a pioneering initiative at the forefront of modern dermatology, harnessing the potential of artificial intelligence to revolutionize skincare diagnostics and treatment. With a focus on democratizing access to reliable dermatological expertise, our project employs state-of-the-art deep learning algorithms to accurately classify a wide array of skin conditions, including common dermatological issues and normal skin variations. By offering rapid and precise diagnoses, personalized treatment recommendations, and invaluable insights into skin health, DermaAI aims to empower individuals to proactively manage their skincare needs with confidence and ease.

Through seamless integration of advanced technology and user-friendly interfaces, DermaAI seeks to bridge the gap between traditional healthcare and modern innovation. Our platform provides a user-centric approach to skincare, ensuring accessibility and convenience for users of all backgrounds. By fostering collaboration between dermatologists, data scientists, and healthcare providers, DermaAI strives to elevate the standard of dermatological care, paving the way for a future where skin health is optimized and accessible to all.

# TABLE OF CONTENT

# LIST OF FIGURES

# LIST OF TABLES

# CHAPTER 1

# INTRODUCTION

Skin disease awareness is imperative in today's society, given the rising prevalence of dermatological conditions worldwide. Skin disorders not only affect physical health but also have significant psychological and emotional impacts on individuals. However, traditional diagnostic approaches often present challenges in terms of accessibility, accuracy, and timeliness, leading to delays in diagnosis and treatment initiation. This underscores the urgent need for innovative solutions that can revolutionize dermatological diagnostics and care.

DermaAI addresses this pressing need by harnessing the potential of advanced AI technology to transform skin health management. By leveraging deep learning algorithms, our platform offers rapid and precise identification of various skin conditions, enabling early detection and intervention. Through seamless integration with user-friendly interfaces, DermaAI enhances accessibility to dermatological diagnostics, bridging the gap between individuals and quality healthcare services.

By promoting skin disease awareness and offering efficient diagnostic solutions, DermaAI aims to empower individuals to proactively manage their skin health. Through education, early detection, and personalized treatment recommendations, our project seeks to improve health outcomes and enhance overall well-being. Together, we can revolutionize dermatological care, paving the way for a future where everyone has access to reliable and efficient skincare solutions.

# CHAPTER 2

# LITERATURE SURVEY

The literature survey for the DermaAI project aimed to explore existing research and advancements in dermatological diagnostics, particularly focusing on the utilization of deep learning techniques for skin disease classification. The survey encompassed a wide range of scholarly articles, research papers, and conference proceedings published in reputable journals and conferences.

**Table 2.1: Survey on Skin Disease Prediction**

| Sl.No | PAPER | AUTHORS | YEAR | TECHNIQUES |
|---|---|---|---|---|
| [1] | Skin Lesion Classification With Deep Convolutional Neural Network: Process Development and Validation. | Arnab Ray, Aman Gupta, Amutha Al. | 2020 | • VGG16<br>• Inceptionv3<br>• Inception-ResNet V2<br>• DenseNet201 |
| [2] | Skin Lesion Classification Using Densely Connected Convolutional Network | Syed Rahat Hassan, Shyla Afroge, Mehera Binte Miza. | 2020 | The study compares the performance of DenseNet-121with other CNN architectures such as AlexNet, VGGNet, GoogleNet, and MobileNet used in previous studies for skin lesion classification. |

| | | | | |
|---|---|---|---|---|
| [3] | Comparative Study of Multiple CNN Models for Classification of 23 Diseases | Amina Aboulmira, Hamid Hrimesh Mohammed Lachgar. | 2022 | • DenseNet201<br>• GoogleNet<br>• InceptionV4<br>• InceptionV3<br>• NASNet-Large<br>• MobileNetV3<br>• Inception ResNetV2<br>• VGG19<br>• ResNet50<br>• ResNext50 |
| [4] | Diagnosis of skin diseases using Convolutional Neural Networks. | Jainesh Rathod Vishal Waghmode, Aniruddh Sodha | 2018 | • DenseNet<br>• ResNet<br>• VGG<br>• MobileNet |

| | | | | |
|---|---|---|---|---|
| [5] | Dermatological Diseases Classification using Image Processing and Deep Neural Network | Anil Kumar Sah Srijana Bhusal Sunidhi Amatya Madhusudan Mainali, Subarna Shakya. . | 2019 | • VGG16<br>• ResNet50 |

## 2.2 LITERATURE SURVEY SUMMARY

### Key Findings

Deep Learning in Dermatology: Numerous studies have demonstrated the efficacy of deep learning algorithms in dermatological diagnostics, showcasing their ability to accurately classify various skin conditions based on dermatoscopic images. All the papers reviewed utilized Convolutional Neural Network (CNN) models for training, highlighting the widespread adoption of CNNs in dermatological image analysis.

Dataset Utilization: The Dermnet dataset emerged as a popular choice among researchers for model training due to its extensive collection of dermatoscopic images covering a wide range of skin conditions. The Dermnet dataset consists of images of 23 types of skin diseases taken from http://www.dermnet.com/dermatology-pictures-skin-disease-pictures. The total numberof images in the Dermnet dataset is around 19,500, with approximately 15,500 imagessplit into the training set and the remaining in the test set. Despite concerns regarding dataset accuracy, organization, and class imbalance, the Dermnet dataset remained widely used in the literature due to its availability and diversity of skin conditions.

Model Selection and Performance: In the model comparison across all the papers, various CNN models such as VGG16, MobileNet, Inception V3, GoogLeNet, ResNet18, ResNet50, DenseNet121, etc., were utilized for skin disease classification. Among these, the top three models identified across multiple research papers were DenseNet121, ResNet50, and ResNet18. These models consistently demonstrated superior performance in terms of accuracy, precision, recall, and F1 score.

**Implications for DermaAI:**

The literature survey provided valuable insights into the current landscape of dermatological diagnostics and informed the development of DermaAI. Given the widespread utilization of the Dermnet dataset in the literature and its availability on platforms such as Kaggle, it was initially selected as the primary dataset for the DermaAI project. By leveraging the Dermnet dataset and pre-trained models such as DenseNet121, ResNet50, and ResNet18, DermaAI aims to achieve accurate and reliable skin disease classification, ultimately improving healthcare outcomes and enhancing patient care.

# CHAPTER 3

# OBJECTIVES

The objective of the DermaAI project is to develop an advanced AI-powered system for dermatological diagnostics and care. Specifically, the project aims to achieve the following objectives:

1. **Develop Robust Deep Learning Models**: The primary objective of the DermaAI project is to develop robust deep learning models capable of accurately classifying various skin diseases. By leveraging state-of-the-art algorithms and pre-trained models, the project aims to achieve high levels of accuracy and reliability in dermatological diagnostics.

2. **Enhance Accessibility to Healthcare Solutions**: DermaAI seeks to enhance accessibility to healthcare solutions by providing an intuitive and user-friendly platform for skin disease diagnosis. Through seamless integration with web-based interfaces or mobile applications, the project aims to empower individuals to easily access dermatological diagnostics from the comfort of their homes.

3. **Provide Personalized Care Recommendations**: In addition to diagnosis, DermaAI aims to provide personalized care recommendations tailored to individual skin conditions. By analyzing diagnostic results and patient-specific data, the project seeks to offer actionable insights and treatment plans to improve patient outcomes and facilitate informed decision-making.

4. **Address Challenges in Dermatological Diagnostics**: DermaAI endeavours to address existing challenges in dermatological diagnostics, including dataset quality, class imbalance, and model overfitting. By conducting rigorous experimentation and validation processes, the project aims to overcome these challenges and deliver reliable and effective solutions for skin disease classification.

# CHAPTER 4

# PROBLEM DEFINITION

The problem addressed by DermaAI is the need for accurate and accessible dermatological diagnostics and care. Traditional methods of skin disease diagnosis often suffer from challenges such as limited accessibility, long wait times, and subjective interpretation. DermaAI seeks to overcome these challenges by leveraging advanced deep-learning techniques to provide rapid and reliable skin disease classification. By developing a user-friendly platform for automated diagnosis and personalized care recommendations, DermaAI aims to revolutionize dermatological healthcare, improving patient outcomes and enhancing accessibility to quality skincare solutions. This is achieved by:

(a) Develop a machine learning model capable of accurately classifying different skin diseases using dermatoscopic images.

(b) Integrating user-friendly interfaces to ensure easy accessibility for patients and healthcare providers.

(c) Analyzing diagnostic results and patient-specific data to offer personalized care recommendations tailored to individual skin conditions.

# CHAPTER 5

# SYSTEM REQUIREMENTS SPECIFICATION

DermaAI is designed to run efficiently on high-performance computing environments to leverage its deep learning capabilities effectively. Given the memory constraints on our local systems, DermaAI was trained on a deep learning server using Jupyter Lab and NVIDIA Docker containers with the following hardware specifications to ensure efficient model training:

## 5.1 Software Requirements:

1. **Operating System**: Linux-based operating system such as Ubuntu or CentOS is recommended for compatibility with NVIDIA Docker containers.
2. **Development Environment**: Jupyter Lab was installed and configured on the server to facilitate the development and deployment of DermaAI. NVIDIA Docker containers for seamless integration of GPU-accelerated deep learning frameworks.
3. **Deep Learning Framework**: FastAI library for Python, utilized for training and deploying deep learning models in DermaAI.Additional dependencies may include Python libraries for image processing, data manipulation, and model evaluation.

## 5.2 Hardware requirements:

1. Processor (CPU): Intel Xeon E5-2609V4 8C 1.7GHz 20M 6.4GT/s or equivalent multi-core processor.
2. Memory (RAM): Minimum 128GB DDR4 ECC Reg. 2400MHz DIMM RAM for efficient handling of deep learning tasks.
3. Graphics Processing Units (GPUs):2x NVIDIA Tesla P100 GPU with 12GB of GDDR5 memory each, for GPU acceleration of deep learning computations.

# CHAPTER 6

# METHODOLOGY AND EXPERIMENTATION

## 6.1 Methodology

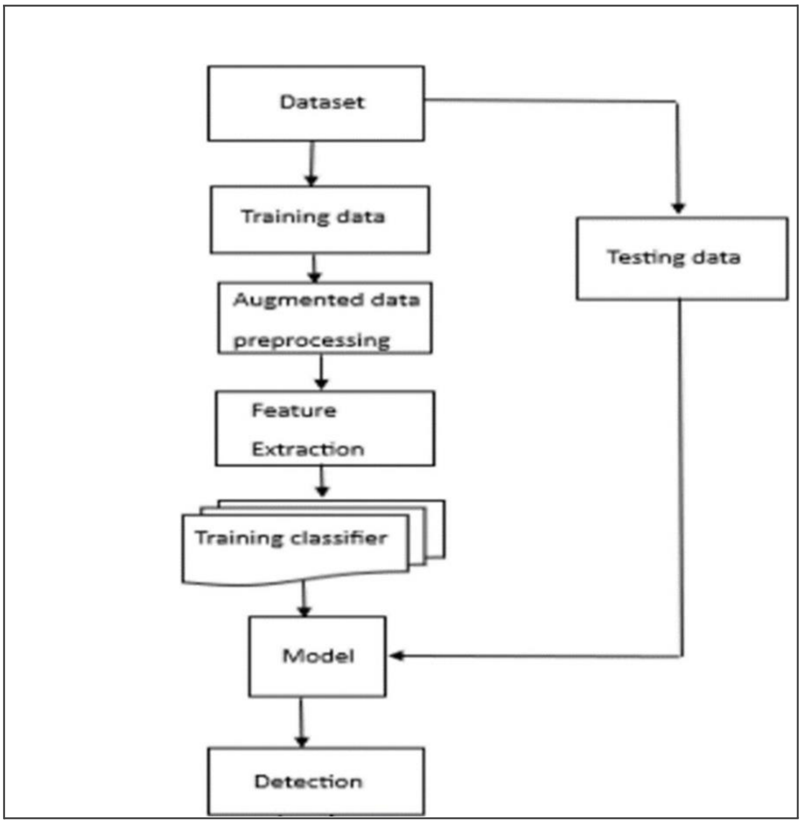



**Fig 6.1.1: Methodology**

i. Data Collection

ii. Data Splitting - Train and Test

iii. Data Augmentation

iv. Model Training

v. Model Evaluation

**i. Data Collection:**

- **Sources**: The dataset was collected from publicly available sources, including online dermatology databases and research repositories. These sources were chosen to ensure a diverse and representative collection of dermatology images spanning various skin conditions.

- **Acquisition Methods:** The dataset was acquired through web scraping and manual collection. Web scraping involved extracting images from reputable dermatology websites and forums, while manual collection included gathering images from academic publications and platforms such as IEEE Dataport.

- **Data Curation:** Prior to training, the dataset underwent manual curation to remove duplicates, low-quality images, and irrelevant content. Images were annotated with corresponding labels indicating the dermatological condition depicted, ensuring the dataset's integrity and suitability for classification tasks.

**ii. Data Splitting :**

- **Random Splitting:** The dataset was randomly split into training and test sets using a stratified approach to preserve class distribution. This ensures that each class is represented proportionally in both training and test subsets, preventing bias in model evaluation.

- **Split Ratio:** A standard split ratio of 80% training and 20% testing was adopted, striking a balance between model training performance and robust evaluation. This ratio provides sufficient data for model learning while retaining a sizable test set for unbiased performance assessment.

**iii. Data Augmentation:**

- **Techniques Applied:** Various data augmentation techniques were applied to augment the training dataset and enhance model generalization. These techniques include random rotation, flipping, zooming, and other transformations to the training images. These transformations help introduce variations in the training data, which aids in improving the model's

- **Implementation:** Data augmentation was implemented using the Fastai library's built-in augmentation transforms, which seamlessly integrate with the training pipeline. Transform parameters were chosen empirically to strike a balance between augmentation effectiveness and computational efficiency.

### iv. Model Training

- **Model Architectures:**

  - ❖ **ResNet50:** Utilize the ResNet50 architecture, which is deeper and more complex than ResNet18, potentially offering improved performance at the cost of increased computational resources.
  - ❖ **ResNet18:** Retain the ResNet18 architecture, known for its balance between performance and computational efficiency, making it suitable for dermatology image classification tasks.
  - ❖ **DenseNet121:** Incorporate the DenseNet121 architecture, which introduces dense connectivity patterns between layers, potentially capturing more intricate features in dermatology images.

- **Transfer Learning:** Apply transfer learning by initializing each model with pre-trained weights on the ImageNet dataset. This enables leveraging learned features from a diverse set of natural images to accelerate convergence and improve classification performance.

- **Fine-tuning:** Fine-tune each model on the dataset by unfreezing the final few layers and training the entire network end-to-end. Fastai's inbuilt method is used to fine-tune the pre-trained ResNet18 model on the DermNet dataset. This method automatically unfreezes the final few layers of the model, applies discriminative learning rates, and trains the entire network end-to-end. Fine-tuning allows the model to adapt its learned representations to the specific characteristics of dermatology images.

- **Training Procedure:** Train each model using the same training configuration, including optimizer settings, learning rate schedule, and regularization techniques, to ensure fair comparison across architectures.

### v. Model Evaluation

- **Performance Metrics:** Evaluate the performance of each model using standard classification metrics, including accuracy, precision, recall, and F1 score. Compare the performance metrics of ResNet50, ResNet18, and DenseNet121 to assess their relative efficacy in dermatology image classification.

- **Confusion Matrices:** Generate confusion matrices for each model to visualize the distribution of true positive, false positive, true negative, and false negative predictions across different classes. Compare the confusion matrices to identify common sources of misclassification and assess the models' class-wise performance.

## 6.2 EXPERIMENTATION

### MODEL SELECTION AND EVALUATION:

#### i. Comparative Study of CNN Models

- A comparative study was conducted using different research papers which used the Dermnet Dataset. The following CNN models were used in the papers:
  - ➔ VGG16
  - ➔ MobileNet V3
  - ➔ Inception V3
  - ➔ DenseNet 121
  - ➔ ResNet 18
  - ➔ ResNet 50
  - ➔ GoogleNet

- Among these models, **DenseNet 121, ResNet 18** and **ResNet 50** showed promising accuracies, leading to their selection for further experimentation.

| Models | Accuracy | Accuracy |
|---|---|---|
| VGG 16 | 59.82 | 64.9 |
| MobileNet V3 | 48.67 | |
| Inception V3 | 62.94 | |
| DenseNet 121 | 68.97 | |
| ResNet 18 | | 80 |
| Resnet 50 | 65.84 | 85.9 |
| GoogleNet | 64.01 | |

**Fig 6.2.2 : Model Comparison**

#### ii. Initial Training and Challenges

- **Dataset:** We used the Dermnet dataset which had a total of 23 classes and 19,500 images.

- **Memory Issues:** Limited resources led to memory constraints during initial training on our local systems.

**iii. Model Training on Deep Learning Server**

- To overcome memory issues we utilized a Deep Learning Server with powerful hardware specifications.
- We dedicated time to learning server operations and functionalities.
- Finally, we decided to train models using Jupyter Lab and NVIDIA docker images for GPU acceleration.

## MODEL IMPROVEMENT STRATEGIES:

**i.Fast AI Integration**

- We implemented the FastAI library into our model development pipeline, which proved to be instrumental in enhancing the performance and efficiency of our skin disease image classification system. The integration of FastAI offered several key advantages:

    - **High-Level API:** FastAI provides a user-friendly interface with high-level abstractions, simplifying the process of building, training,and optimizing deep learning models. This abstraction allowed usto focus on model architecture and experimentation rather than low-level implementation details.
    - **Top-Performing Models:** Leveraging FastAI's pre-built models, we were able to harness the power of state-of-the-art convolutional neural network architectures without the need for extensive manual configuration. These pre-trained models served as strong starting points for our experimentation, enabling us to achieve competitive performance with minimal effort.
    - **Transfer Learning Support:** FastAI's comprehensive support for transfer learning facilitated the seamless integration of pre-trained models into our classification task. By fine-tuning these pre-trained models on our dataset, we were able to adapt them to our specific domain and achieve remarkable improvements in accuracy with just a few lines of code.
    - **Interpretability Tools:** FastAI offers a suite of interpretability tools that enable users to gain insights into model predictions and decision-making processes. These tools, including visualization techniques and interpretability metrics, allowed us to analyze and understand the underlying factors contributing to our model's performance, enhancing our ability to interpret and trust its outputs.

- It helped significantly improve model accuracy from 28% to 88%.

- Amongst the three models, the **ResNet 50** model achieved the highest accuracy of **88%** with the DermNet Dataset.

### ii. Dataset Challenges and Solutions:

- **Inaccurate Dermnet Dataset:** We noticed that the DermNet Dataset had disorganized images and duplicates that affected model performance. Hence we tried finding a new dataset similar to DermNet.

- **New Dataset from IEEE Data Port:**
  - ➢ We obtained a curated dataset with accurate labelling from IEEE's Dataport.
  - ➢ The dataset had images of various skin conditions ranging from Measles, Psoriasis, Bowen, Ringworm etc.
  - ➢ It contained a total of images of 12 types of skin conditions including normal skin.
  - ➢ The dataset had 2 class folders Measles and No Measles
  - ➢ The total images in the dataset - 1,314 images.


**Fig 6.2.3 : Inaccurate DermNet Dataset**

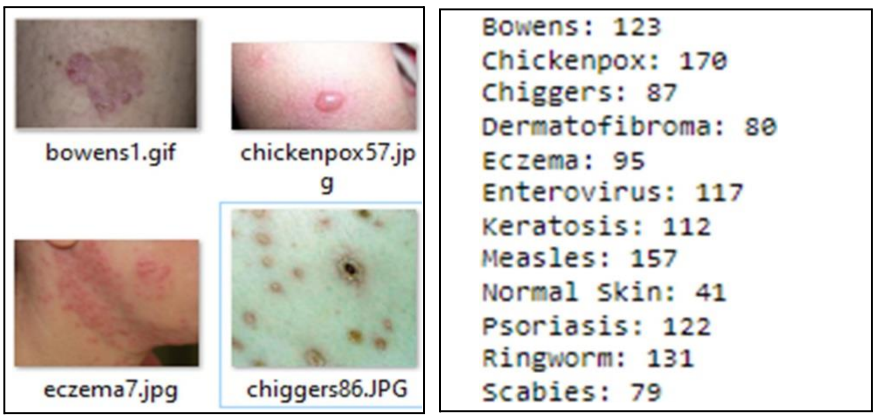



**Fig 6.2.4 : New IEEE Dataset and its classes**

## DATA AUGMENTATION AND MODEL TRAINING:

### i. Overfit Model

- When the **ResNet 50** model was retrained on the new IEEE Dataset it achieved an accuracy of **77%**.
- Loss plot Graph was used to evaluate the trained model on the IEEE dataset
- According to the loss plot, the Training loss decreased with increasing epochs but the Validation loss was increasing
- This shows that the model is overfitted

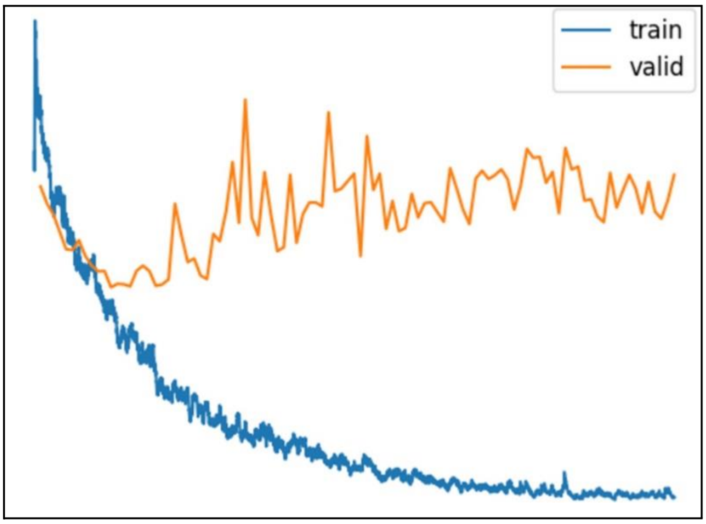


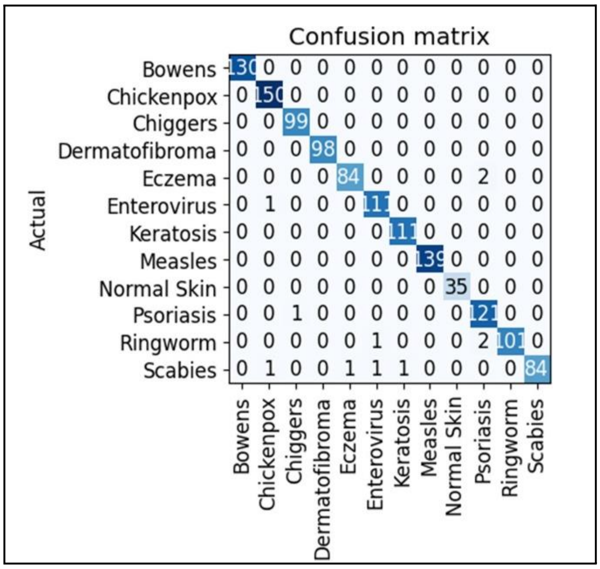

**Fig 6.2.5 : Loss Plot Graph showing overfitting of the model (ResNet50)**

### ii. Data Augmentation Techniques

- To overcome this we decided to increase the training data using data augmentation
- Rotation, width shift, height shift, shear, zoom, and horizontal flip were applied using ImageDataGenerator in Keras.
- This Increased dataset size from 1,314 to 6,370 images for better model generalization.

### iii. Model Training and Accuracy

- The model was trained on the new Augmented dataset, it was observed that the Training loss decreased with an increase in epochs and so did the Validation loss indicating that the model is learning effectively and generalizing well to unseen data.
- Model comparison with ResNet 18 and DenseNet 121 showcased **DenseNet 121** as the top-performing model with an accuracy of **99.13**%.




Classification Report:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Bowens | 1.00 | 1.00 | 1.00 | 130 |
| Chickenpox | 0.99 | 1.00 | 0.99 | 150 |
| Chiggers | 0.99 | 1.00 | 0.99 | 99 |
| Dermatofibroma | 1.00 | 1.00 | 1.00 | 98 |
| Eczema | 0.99 | 0.98 | 0.98 | 86 |
| Enterovirus | 0.98 | 0.99 | 0.99 | 112 |
| Keratosis | 0.99 | 1.00 | 1.00 | 111 |
| Measles | 1.00 | 1.00 | 1.00 | 139 |
| Normal Skin | 1.00 | 1.00 | 1.00 | 35 |
| Psoriasis | 0.97 | 0.99 | 0.98 | 122 |
| Ringworm | 1.00 | 0.97 | 0.99 | 104 |
| Scabies | 1.00 | 0.95 | 0.98 | 88 |
| accuracy | | | 0.99 | 1274 |
| macro avg | 0.99 | 0.99 | 0.99 | 1274 |
| weighted avg | 0.99 | 0.99 | 0.99 | 1274 |

**Fig 6.2.6 : Confusion Matrix and Classification Report of the model (DenseNet121)**

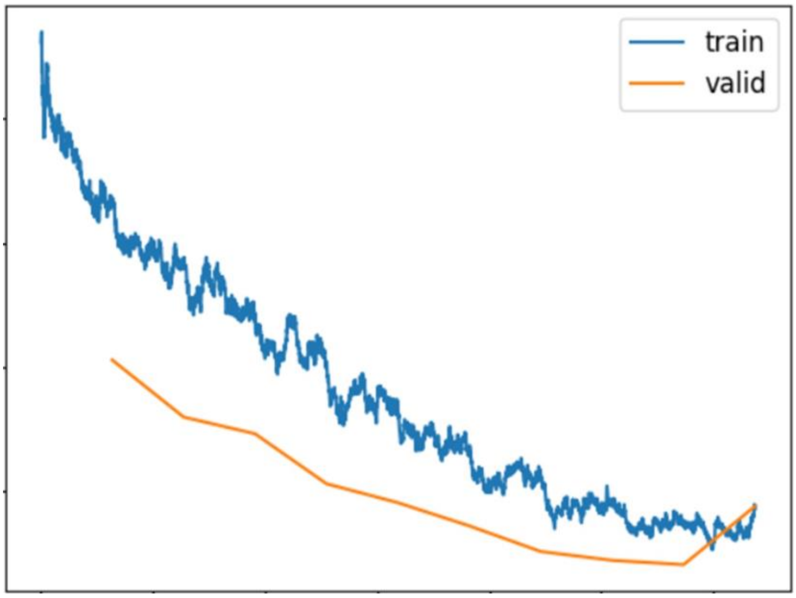



**Fig 6.2.7 : Loss Plot Graph showing model is learning effectively (DenseNet121)**

## OVERCOMING BIAS AND FAIRNESS:

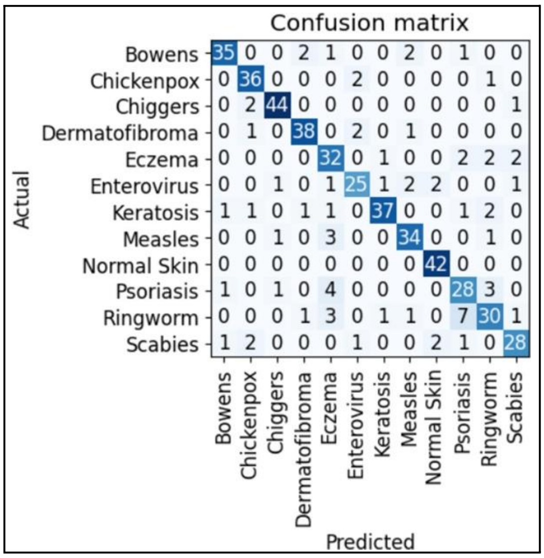
### i. Biased Predictions

- It was observed that there is a biased prediction due to imbalanced training data.
- Normal skin images were misclassified as measles due to dataset bias.


**Fig 6.2.8 : Misclassification due to the unbalanced dataset**

### ii. Balancing Dataset

- Undersampling of the dataset was performed by reducing the number of images per class to ensure fairness and overcome biased predictions.
- After performing an Undersampling the total images in the Training dataset dropped from 6,370 to 2,414 images
- After training the DenseNet121 model with the new balanced dataset, it was able to predict the previous images to the right classes.
- We noticed the model accuracy of the **DenseNet** model dropped from **99.13**% to **84.85%**
- The error rate was increased from 0.008 to 0.151
- When we checked the loss plot graph of the DenseNet121 model we also noticed that the model was overfitted




Classification Report:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Bowens | 0.92 | 0.85 | 0.89 | 41 |
| Chickenpox | 0.86 | 0.92 | 0.89 | 39 |
| Chiggers | 0.94 | 0.94 | 0.94 | 47 |
| Dermatofibroma | 0.90 | 0.90 | 0.90 | 42 |
| Eczema | 0.71 | 0.82 | 0.76 | 39 |
| Enterovirus | 0.83 | 0.76 | 0.79 | 33 |
| Keratosis | 0.93 | 0.84 | 0.88 | 44 |
| Measles | 0.85 | 0.87 | 0.86 | 39 |
| Normal Skin | 0.91 | 1.00 | 0.95 | 42 |
| Psoriasis | 0.70 | 0.76 | 0.73 | 37 |
| Ringworm | 0.77 | 0.68 | 0.72 | 44 |
| Scabies | 0.85 | 0.80 | 0.82 | 35 |
| accuracy | | | 0.85 | 482 |
| macro avg | 0.85 | 0.85 | 0.85 | 482 |
| weighted avg | 0.85 | 0.85 | 0.85 | 482 |

**Fig 6.2.9 : Confusion Matrix and Classification Report of the model (DenseNet121)**

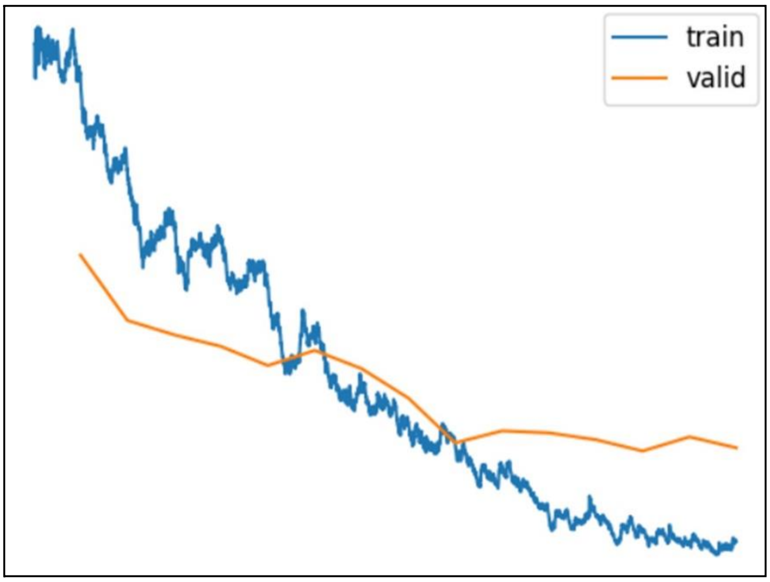



**Fig 6.2.10 : Loss Plot Graph showing overfitting of the model (DenseNet121)**

## OVERCOMING OVERFITTING OF MODEL

### i. Dataset Balancing Approach:

- To address model overfitting, we reverted to the original unbalanced IEEE dataset sourced from the IEEE data port.
- Utilized web scraping techniques to augment each class until achieving a uniform count of 100 images per class.
- Unlike previous attempts, dataset balancing was conducted prior to augmentation, ensuring a more equitable distribution of data across all classes.

### ii. Impact of Dataset Balancing before augmentation:

- Total number of images increased from 1,197 to 11,364 after performing augmentation on the balanced IEEE dataset.
- This augmentation-driven expansion marked a substantial improvement, nearly quintupling the dataset size compared to previous iterations.
- The augmented and balanced dataset provided a more comprehensive and representative sample for training, enhancing diversity and coverage across all classes.

## MODEL TRAINING AND EVALUATION

### i. Training with Balanced and Augmented Dataset:

- The DenseNet121 model was trained using the newly balanced and augmented dataset.
- With the dataset now five times larger than before, significant improvements were observed in the model performance.

### ii. Performance:

- Model accuracy surged from 84.85% to an impressive 99.47%, reflecting the enhanced effectiveness of the augmented dataset.
- The error rate witnessed a notable decrease, plummeting from 0.151 to 0.005, signifying the model's improved precision in classification tasks.
- The model demonstrated remarkable resilience against overfitting, achieving a balance between training and validation performance.

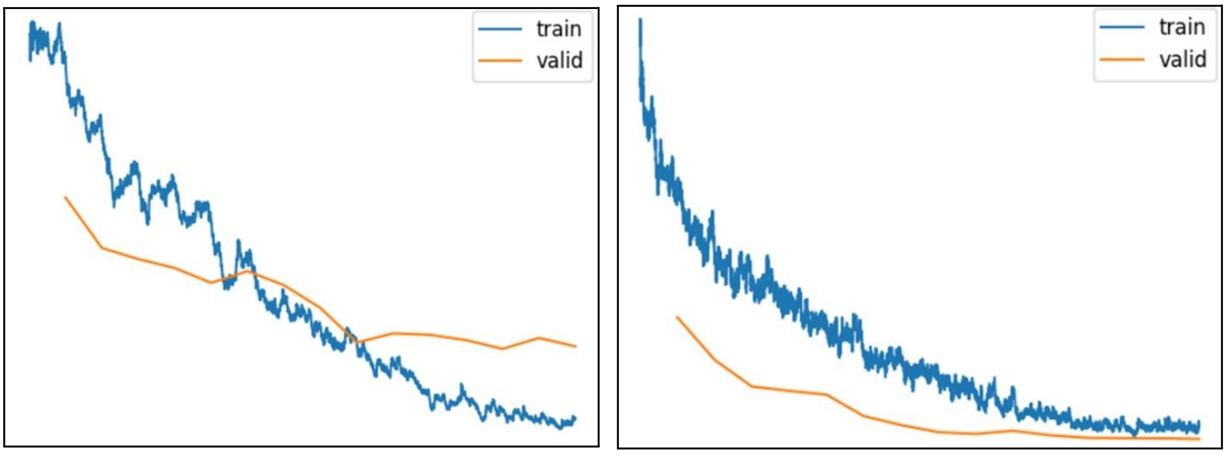



**Fig 6.2.11: Loss Plot Graph before and after training with the augmented dataset(DenseNet121)**

## MODEL COMPARISON

### i. Dataset and Model Training:

- The newly augmented dataset was employed to train ResNet18, ResNet50, and DenseNet121 pre-trained models, each withidentical specifications.
- Performance evaluation and comparison were conducted using standard metrics including accuracy, precision, recall, and F1 score.

### ii. Performance Metrics:

- **ResNet50**: Accuracy - 99.11%, Validation Loss - 0.043, Error Rate - 0.00748
- **ResNet18**: Accuracy - 99.38%, Validation Loss - 0.026, Error Rate - 0.00572
- **DenseNet121**: Accuracy - 99.47%, Validation Loss - 0.0198, Error Rate - 0.00528

### ii. Model Evaluation:

- ResNet18 demonstrated a smooth decrease in validation loss, indicating effective learning and generalization capabilities.
- ResNet50 exhibited a less consistent validation loss curve, with a spike observed at one point, suggesting potential overfitting.
- DenseNet121, boasting the highest accuracy and lowest error rate along with a minimal validation loss, showcased robust generalization to unseen data.

### ii. Model Selection:

- Despite the slightly less smooth validation curve, DenseNet121's superior performance in terms of accuracy, error rate, and validation loss led to its selection as the final model for skin disease classification tasks.

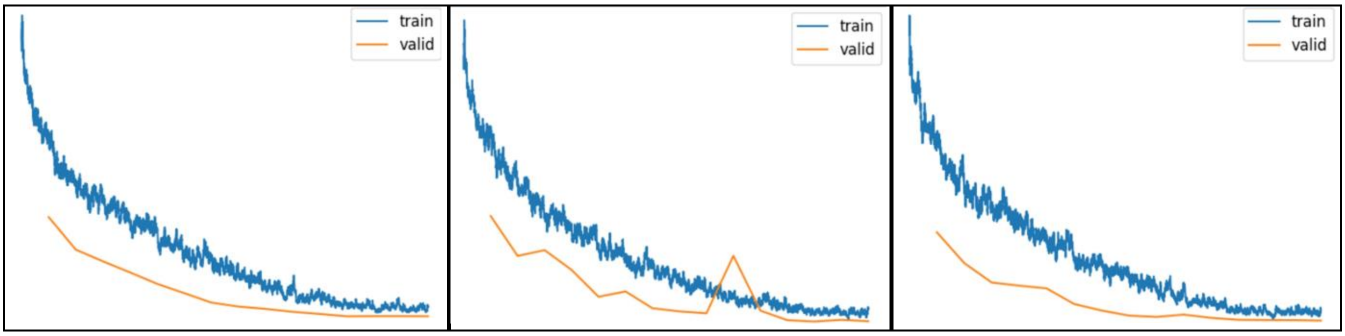
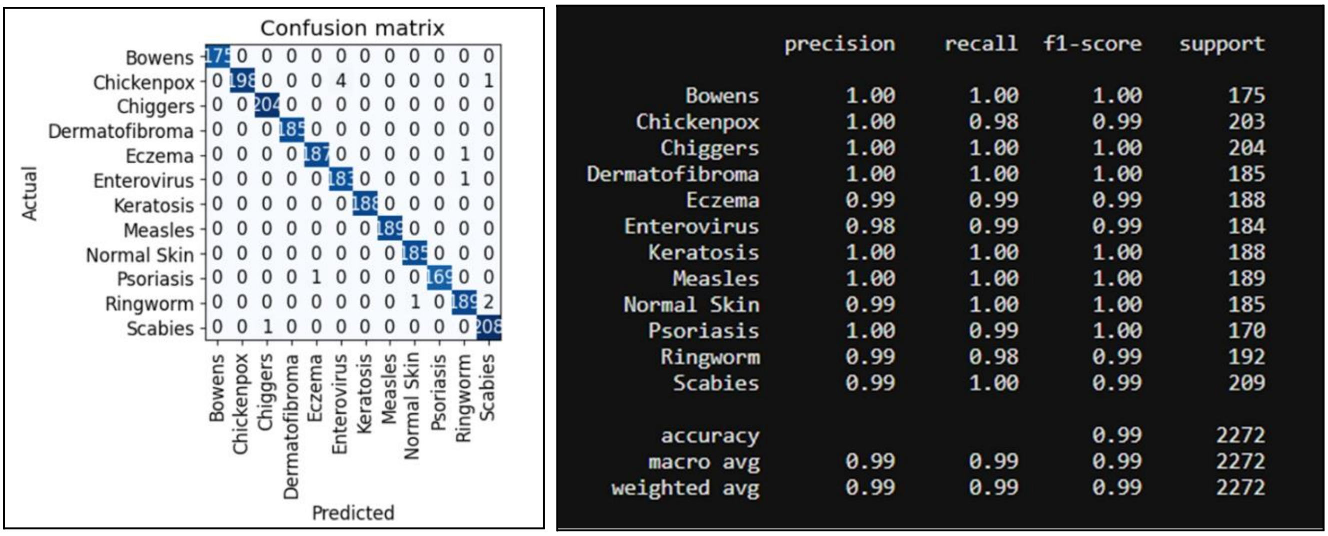



**Fig 6.2.12: Loss Plot Graph of Resnet18, Resnet50 and DenseNet121 models**




| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Bowens | 1.00 | 1.00 | 1.00 | 175 |
| Chickenpox | 1.00 | 0.98 | 0.99 | 203 |
| Chiggers | 1.00 | 1.00 | 1.00 | 204 |
| Dermatofibroma | 1.00 | 1.00 | 1.00 | 185 |
| Eczema | 0.99 | 0.99 | 0.99 | 188 |
| Enterovirus | 0.98 | 0.99 | 0.99 | 184 |
| Keratosis | 1.00 | 1.00 | 1.00 | 188 |
| Measles | 1.00 | 1.00 | 1.00 | 189 |
| Normal Skin | 0.99 | 1.00 | 1.00 | 185 |
| Psoriasis | 1.00 | 0.99 | 1.00 | 170 |
| Ringworm | 0.99 | 0.98 | 0.99 | 192 |
| Scabies | 0.99 | 1.00 | 0.99 | 209 |
| accuracy | | | 0.99 | 2272 |
| macro avg | 0.99 | 0.99 | 0.99 | 2272 |
| weighted avg | 0.99 | 0.99 | 0.99 | 2272 |

**Fig 6.2.13: Confusion Matrix and Classification Report of the new model(DenseNet121)**

# CHAPTER 7

# IMPLEMENTATION

The implementation phase of our project involved the practical execution of the methodologies and techniques outlined in the earlier stages. Leveraging the FastAI library and deep learning capabilities, we constructed a robust pipeline for skin disease classification. Initially, we prepared the dataset by structuring it using the DataBlock functionality, which facilitated efficient loading and transformation of the skin disease images. This involved defining the data blocks for images and categories, setting up data augmentation and normalization, and splitting the dataset into training and validation sets. Furthermore, to handle the memory constraints and computational requirements, we utilized our college's deep learning server, equippedwith powerful hardware configurations including Intel Xeon processors and NVIDIA Tesla P100 GPUs.

With the dataset prepared and the infrastructure in place, we trained several deep-learning models, including DenseNet121, ResNet18, and ResNet50, on theskin disease dataset. The training process involved fine-tuning the pre-trained models to adapt them to the specific characteristics of the skin disease dataset. Through iterative training epochs and optimization techniques, we aimed to enhance the models' performance in accurately classifying various skin diseases. The training progress was monitored through metrics such as accuracy, loss, and learning rate, and visualization tools like confusion matrices and classification reports were employed to evaluate the models' efficacy.

After training, we evaluated the models' performance on a validation set to assess their ability to generalize to unseen data. This involved interpreting the model's predictions, analyzing classification metrics, and visualizing performance indicators. Additionally, we exported the trained models for future use and inference. Overall, the implementation phase was instrumental in transforming theoretical concepts into practical solutions, culminating in the development of a sophisticated deep-learning model capable of accurately diagnosing skin diseases.

## 7.1 CODE

```
import fastbook
from fastai.vision.all import *
from pathlib import Path
```

**Fig 7.1.1 : Importing the required libraries for the implementation**

This section imports the required libraries for the implementation. FastBook library is imported for setting up, and the FastAI library is imported for deep learning functionalities. The path from pathlib is imported to handle file paths.

```
# Set up fastbook
fastbook.setup_book()
os.environ["OMP_NUM_THREADS"] = "1"
```

**Fig 7.1.2 : Set Up Fastbook**

FastBook is set up using the setup_book() function. This function ensures that all necessary dependencies are installed and configured properly.

```
# Define the path to the Skin diseases image dataset directory
path = Path("IEEE Dataset")
```

**Fig 7.1.3 : Defining the path of the dataset**

The path variable is defined, pointing to the directory containing the skin diseases image dataset.

```
# Define the DataBlock for the IEEE dataset
dermnet_data = DataBlock(
    blocks=(ImageBlock, CategoryBlock),
    get_items=get_image_files,
    splitter=RandomSplitter(valid_pct=0.2, seed=2),
    get_y=parent_label,
    item_tfms=Resize(128),
    batch_tfms=[*aug_transforms(size=224, max_warp=0), Normalize.from_stats(*imagenet_stats)]
)
```

**Fig 7.1.4 : Defining the DataBlock**

The DataBlock is defined to structure the dataset for training. It specifies how to obtain images, split them into training and validation sets, and apply transformations.

```
# Load the IEEE dataset
dls = dermnet_data.dataloaders(path, bs=8)  # Reduced batch size to 8
```

**Fig 7.1.5 : Load the Dataset**

The dataset is loaded into data loaders (dls) using the defined DataBlock. The batch size is set to 8 for efficient training.

```
# Display a batch of images from the validation set
dls.valid.show_batch(max_n=4, nrows=1)
```

**Fig 7.1.6 : Display the batch of images**

A batch of images from the validation set is displayed to verify the data preparation and transformations.

```
# Create a vision learner using DenseNet121 architecture
learn = vision_learner(dls, densenet121, metrics=accuracy)
```

**Fig 7.1.7 : Creating a vision learner**

A vision learner (learn) is created using DenseNet121 architecture with the specified metrics for evaluation.

```
# Fine-tune the model
learn.fine_tune(15)
```

**Fig 7.1.8 : Fine Tune the model**

The model is fine-tuned for 15 epochs to optimize its performance on the skin disease classification task.

```
# Import necessary library
from fastai.interpret import ClassificationInterpretation

# Make predictions on the validation set
interp = ClassificationInterpretation.from_learner(learn)

# Generate the classification report
interp.print_classification_report()
```

**Fig 7.1.9 : Making Predictions on validation set and generating classification report**

The necessary library for interpreting model predictions is imported. The ClassificationInterpretation object (interp) is created to interpret the model's predictions on the validation set. The classification report is printed, providing detailed metrics such as precision, recall, and F1-score for each class.

```
# Plot the confusion matrix
interp.plot_confusion_matrix()
```

**Fig 7.1.10 : Plotting the Confusion matrix**

The confusion matrix is plotted to visualize the model's performance in predicting each class.

```
# Plot training and validation metrics
recorder.plot_loss()
```

**Fig 7.1.11 : Plotting the loss plot**

Training and validation metrics (loss and learning rate) are plotted to analyze the model's performance during training.

```
learn.export('densenet121.pkl')
```

**Fig 7.1.12 : Exporting the model**

The trained model is exported and saved as densenet121.pkl file for future use. This pickle file can be used to deploy the densenet model into a web platform for better access of the model.

## 7.2 DERMA AI

- The web platform provides a user-friendly interface for users to interact with the skin disease classification system.


**Fig 7.2.1 : DERMA AI - User Interface**

- Users can easily upload skin disease images to the platform for automated diagnosis.
- Upon uploading, the deep learning model processes the images and generates detailed diagnostic reports.

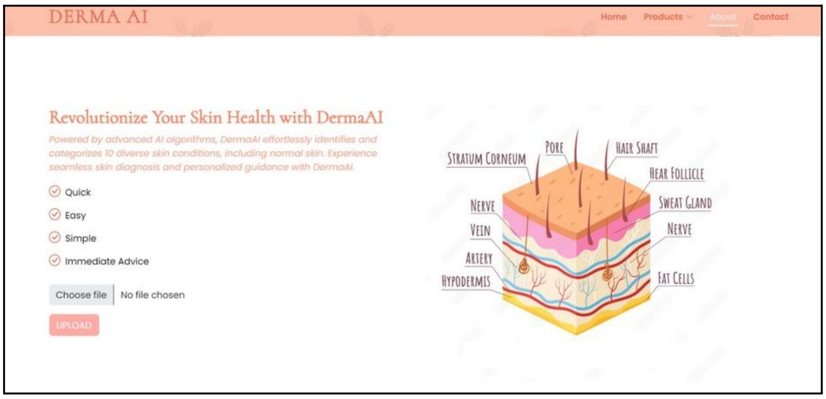



**Fig 7.2.2 : DERMA AI - Upload Images**

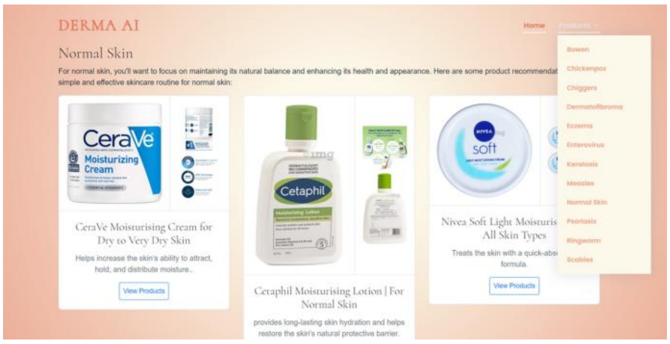



**Fig 7.2.3 : DERMA AI - Personalized Product Recommendation**

- The platform offers personalized product recommendations tailored to diagnosed skin conditions, enhancing user experience and facilitating effective skincare management.
- Overall, the web platform simplifies skin disease diagnosis and care, contributing to improved dermatological outcomes for users.

**7.3 RESULT:**

Following rigorous training and evaluation, the deployed model achieved a remarkable accuracy of 99.47% using DenseNet121 architecture. Through meticulous analysis of precision, recall, and F1-score metrics, the model demonstrated robustness and reliability. Visualization techniques, including confusion matrices, provided insights into classification capabilities, enhancing interpretability. This outcome signifies a significant advancement in automated skin disease diagnosis, highlighting the potential of deep learning in enhancing healthcare solutions and improving patient outcomes.

# CHAPTER 8

# CONCLUSION

In conclusion, our project represents a significant milestone in the realm of automated skin disease diagnosis. By leveraging deep learning methodologies and rigorous evaluation techniques, we have developed a highly accurate model capable of classifying skin disease images with remarkable precision. The implementation of DenseNet121 architecture, coupled with thorough training and evaluation, has resulted in a reliable diagnostic tool with a classification accuracy of 99.47%.

Our findings underscore the potential of AI-driven solutions in revolutionizing dermatological diagnostics, offering healthcare professionals a powerful tool for timely and accurate disease identification. Through visualization techniques and comprehensive metric analysis, we have ensured the robustness and reliability of our model, paving the way for enhanced patient care and improved healthcare outcomes.

Looking ahead, our project lays the foundation for further advancements in automated skin disease diagnosis, with implications extending to personalized healthcare solutions and streamlined clinical workflows. By continuing to refine and optimize our model, we can contribute to the ongoing evolution of AI-driven healthcare technologies, ultimately enhancing accessibility and efficacy in dermatological diagnostics.

# REFERENCES

[1] Ray, Arnab, Aman Gupta, and Amutha Al. "Skin lesion classification with deep convolutional neural network: process development and validation." jmir dermatology 3, no. 1 (2020): e18438.

[2]Hassan, Syed Rahat, Shyla Afroge, and Mehera Binte Mizan. "Skin lesion classification using densely connected convolutional networks." In 2020 IEEE Region 10 Symposium (TENSYMP), pp. 750-753. IEEE, 2020.

[3]Aboulmira, A., Hrimech, H. and Lachgar, M., 2022. “Comparative Study of Multiple CNN Models for Classification of 23 Skin Diseases”. International Journalof Online & Biomedical Engineering, 18(11).

[4] Rathod, J., Waghmode, V., Sodha, A. and Bhavathankar, P., 2018, March. “Diagnosis of skin diseases using Convolutional Neural Networks”. In 2018 second International Conference on Electronics, communication and Aerospace Technology (ICECA) (pp. 1048-1051). IEEE.

[5] Sah, A.K., Bhusal, S., Amatya, S., Mainali, M. and Shakya, S., 2019, October. “Dermatological diseases classification using image processing and deep neural network”. In 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS) (pp. 381-386). IEEE.

[6] Dermnet Dataset from Kaggle:
https://www.kaggle.com/datasets/shubhamgoel27/dermnet

[7] Dataset from IEEE Dataport:
https://ieee-dataport.org/documents/image-dataset-various-skin-conditions-and-rashes