### **CHAPTER 1**

### **1. Introduction**

#### 1.1 Introduction of Project

The Skin Disease Classification project aims to build an automated diagnostic system using Deep Learning to identify various skin conditions from dermatoscopic images. Skin diseases like melanoma, basal cell carcinoma, and benign lesions are common worldwide, and their early detection is critical to effective treatment. Traditional diagnostic methods rely heavily on dermatologists' expertise, which can be inconsistent and time-consuming.

This project leverages Convolutional Neural Networks (CNNs) for feature extraction and classification, using the HAM10000 dataset, a comprehensive collection of dermatoscopic images. Additionally, Explainable AI (XAI) techniques are integrated to provide interpretability in the system’s decisions, making it a valuable tool in the clinical domain.

#### 1.2 Motivation of Project

The motivation for this project arises from the increasing incidence of skin cancers globally and the need for faster, more accurate diagnostic tools. Given the complexity and similarity of many skin diseases, it is often difficult for general practitioners or non-specialized clinics to correctly diagnose conditions. AI-powered systems can aid in early detection and reduce misdiagnosis.

The use of deep learning not only automates the feature extraction process but also adapts better to visual patterns compared to traditional machine learning. By incorporating Explainable AI, this system can assist dermatologists by showing why a certain classification was made, thereby increasing trust and aiding clinical decision-making.

#### 1.3 Problem Statement and Objective

**Problem Statement**

Traditional skin disease diagnosis heavily relies on the subjective interpretation of dermatology experts, which introduces variability and inconsistency in diagnostic outcomes. This dependence on expert knowledge poses significant challenges, especially in regions with limited access to specialized healthcare professionals. The manual analysis process is often time-consuming and susceptible to human error, leading to delayed or inaccurate diagnosis. Furthermore, the conventional computer vision approaches for skin disease classification demand extensive preprocessing steps and handcrafted feature extraction, which are not only labor-intensive but also inefficient for handling large-scale datasets. These limitations hinder the scalability and robustness of automated diagnostic tools in clinical settings. As skin diseases vary widely in appearance and complexity, creating an automated system that can generalize well across diverse cases remains a critical challenge.

**Objective:**

1. To design and develop an accurate, automated skin disease classification system leveraging Convolutional Neural Networks (CNNs) for efficient and reliable diagnosis.
2. To utilize the high-resolution dermatoscopic images provided by the HAM10000 dataset, which contains a diverse range of skin lesion types, thereby ensuring comprehensive model training and validation.
3. To implement advanced data augmentation and preprocessing techniques to enhance model generalization and robustness against image variability and noise.
4. To integrate Explainable AI (XAI) methods such as Grad-CAM and SHAP to provide transparent and interpretable visualizations of the model’s decision-making process, aiding clinicians in understanding and trusting the predictions.

### **CHAPTER 2**

#### 2.1 Literature Survey

Recent advancements in medical image analysis have enabled the development of intelligent systems for automated skin disease detection. The Skin Disease Classification System (SDCS) integrates Convolutional Neural Networks (CNN), Explainable AI (XAI) techniques, and Retrieval-Augmented Generation (RAG)-based chatbot frameworks to improve diagnostic accuracy and user interpretability.

CNNs have become the foundation of skin disease classification due to their powerful feature extraction capabilities. When trained on large-scale image datasets like ImageNet and HAM10000 [1]. The HAM10000 dataset, consisting of over 10,000 dermatoscopic images across seven categories, serves as a standard benchmark in dermatological image classification research [2].

To address the black-box nature of deep learning models, Explainable AI techniques such as Grad-CAM and SHAP are integrated into SDCS. Grad-CAM, proposed by Selvaraju et al., generates visual heatmaps that highlight the most influential regions of an input image used by the CNN for classification, enhancing clinical transparency [3]. SHAP (SHapley Additive exPlanations), on the other hand, offers feature-based interpretability by quantifying each input’s contribution to the model’s output [4].

Further enhancing user interaction and system intelligence, RAG-based chatbot architectures are employed to provide personalized feedback. LangChain’s RAG framework enables the chatbot to retrieve relevant contextual information and generate informed responses, even with limited training data, supporting few-shot learning approaches [5]. This hybrid retrieval-generation mechanism ensures accurate query handling and improves user experience, particularly in a healthcare setting.

To support non-specialist users, the chatbot component also facilitates communication of diagnosis results and explanations. The integration of CNN-based diagnostics with natural language interaction bridges the gap between complex model outputs and end-user comprehension [6].

Lastly, challenges such as data imbalance and domain generalization have been addressed using transfer learning techniques. Fine-tuned on dermatological datasets to enhance model performance with limited labeled data [7]. Ensemble learning strategies and hybrid models combining CNN with traditional classifiers like SVM have also been explored to improve classification robustness [8].

#### 2.2 Limitations of Existing Systems / Research Gaps

The primary limitations of existing systems include:

* **High Data Requirements:** Most CNN-based diagnostic systems depend on large, labeled datasets (e.g., HAM10000) for training. However, acquiring high-quality, annotated medical images across diverse skin types and diseases is challenging, especially in underrepresented regions or for rare conditions.
* **Lack of Scalability:** Existing models are typically trained for specific diseases or image types and cannot easily be extended to cover additional skin conditions or demographic groups without complete retraining.
* **Latency and Performance Issues:** Deep learning models, especially those without optimization, often require substantial computational power. This results in slow inference times, making real-time diagnosis on edge devices (e.g., smartphones or rural clinics) impractical without specialized hardware.
* **Limited Explainability:** Most systems operate as black boxes, offering limited insights into how predictions are made. This lack of transparency reduces trust among clinicians and patients and hinders adoption in clinical workflows.

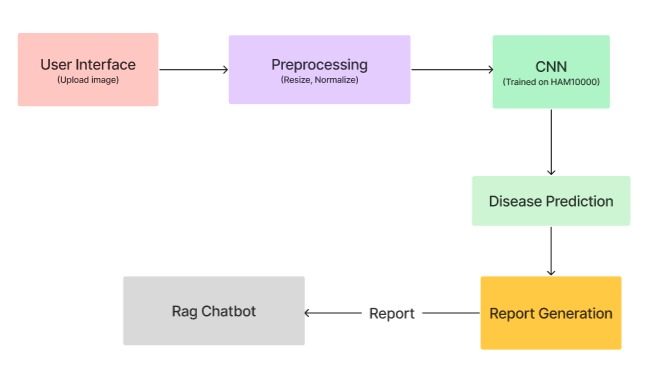
### 

### **CHAPTER 3**

#### 3.1 Proposed System

#### 1 Analysis/Framework/Algorithm/UML Diagrams:

#### 



### **3.2 System Architecture Overview**

The proposed system consists of the following key components:

1. **User Interface**

* Provides an intuitive platform for users to upload skin images, view disease predictions, and interact with the chatbot.

1. **Tech Used**:

* Resizes, normalizes, and optionally augments the image to prepare it for model input.
* **Key Tools**: OpenCV, NumPy, TensorFlow/Keras image preprocessing utilities.

1. **CNN-Based Disease Classifier**

* Classifies the skin lesion using a CNN model trained on the HAM10000 dataset.
* **Model Info**: TensorFlow/Keras CNN, Trained on HAM10000, 7-class classification.

1. **Explainable AI (XAI) Module**

* Generates heatmaps to highlight image regions that influenced the model’s decision.
* **Techniques Used**: Grad-CAM (primary), LIME/SHAP (optional for advanced interpretability)

1. **Report Generation Module**

* Compiles the prediction and XAI output into a structured JSON or report.
* **Formats**: JSON structure, Optional PDF/HTML report, Includes disease, confidence, and heatmap.

1. **RAG Chatbot (Retrieval-Augmented Generation)**

* Provides personalized treatment guidance by combining medical document retrieval and AI generation.
* **Tech Used**: LLM (e.g., GPT), Medical knowledge base, Retrieval system for disease-specific information.

#### 3.3 Component Description

#### 1. Image and Metadata Input

* **Input Type**: RGB images and CSV metadata
* **Source**: HAM10000 dataset
* **Preprocessing**: Each image is mapped using its unique ID from the metadata.

#### 2. Preprocessing Module

* **Resize**: All images are resized to **32x32 pixels**.
* **Normalization**: Pixel values are scaled to [0, 1].
* **Balancing**: Each class is sampled to have **500 images**, ensuring equal representation.

#### 3. Train-Test Split

* Data is split into **75% training** and **25% testing** using stratification to maintain class balance.

#### 4. Data Augmentation

* Introduces variability in training data using:
  + Horizontal and vertical flips
  + Rotation (up to 20 degrees)
  + Zoom, shear, and shift transformations

#### 5. CNN Architecture

* **Input**: (32x32x3)
* **Convolutional Layers**:
  + 256 → 128 → 64 filters
  + ReLU activations
  + Batch Normalization and MaxPooling
  + Dropout layers to reduce overfitting
* **Flatten Layer**: Converts 3D output to 1D vector
* **Dense Layer**: 32 neurons
* **Output Layer**: Softmax with 7 neurons for 7 classes

#### 6. Training Configuration

* **Loss Function**: Categorical Crossentropy
* **Optimizer**: Adam
* **Epochs**: 50
* **Batch Size**: 32
* **Framework**: TensorFlow/Keras

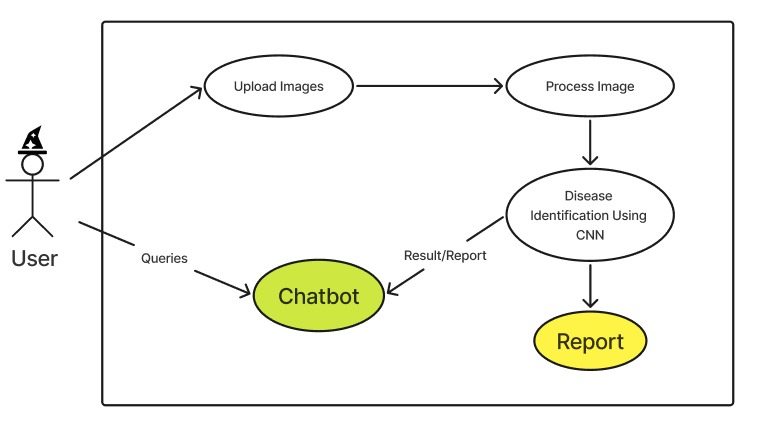
#### 7. Evaluation Module

* Tracks:
  + Training and validation accuracy/loss
  + Confusion matrix for per-class performance
  + Final test accuracy

#### 8. Explainability Layer (Optional)

* Uses Grad-CAM or similar XAI tools to highlight image areas most influential in the model’s decision.

**3.4 Use Case Diagram**



### **Use Cases:**

1. **Upload Images**
   * **Description**: The user uploads a dermoscopic image of the skin lesion to the system.
   * **Actor**: User
   * **Goal**: To initiate disease identification.
2. **Process Image**
   * **Description**: The system preprocesses the uploaded image (e.g., resizing, normalization) to make it suitable for analysis by the CNN.
   * **Actor**: System (automated)
   * **Goal**: Prepare the image for classification.
3. **Disease Identification Using CNN**
   * **Description**: A Convolutional Neural Network (CNN) analyzes the image and classifies it into one of the skin disease categories.
   * **Actor**: System
   * **Goal**: Accurately identify the skin disease.
4. **Report Generation (Report)**
   * **Description**: After disease identification, a result or medical report is generated, possibly containing the predicted class, confidence score, and explanation (e.g., Grad-CAM).
   * **Actor**: System
   * **Goal**: Provide interpretable results to the user.
5. **Chatbot Interaction (Queries)**
   * **Description**: The user can interact with a chatbot to ask queries related to the result, skin disease, or next steps.
   * **Actor**: User
   * **Goal**: Clarify doubts or gain more information via natural language interaction.
6. **Chatbot Response (Result/Report)**
   * **Description**: The chatbot responds with relevant explanations, links, or interpretations retrieved via RAG (Retrieval-Augmented Generation).
   * **Actor**: System (Chatbot)
   * **Goal**: Support the user in understanding the report or disease better.

### **CHAPTER 4**

### **4. Methodology for Skin Disease Classification System**

The methodology for the Skin Disease Classification project involves several systematic stages, from data preprocessing to model evaluation. The primary goal is to build a Convolutional Neural Network (CNN)-based classifier capable of distinguishing between seven types of skin diseases using dermatoscopic images from the HAM10000 dataset.

#### 4.1 Model Performance Visualization

1. **Sample Dermoscopic Images from HAM10000 Dataset**

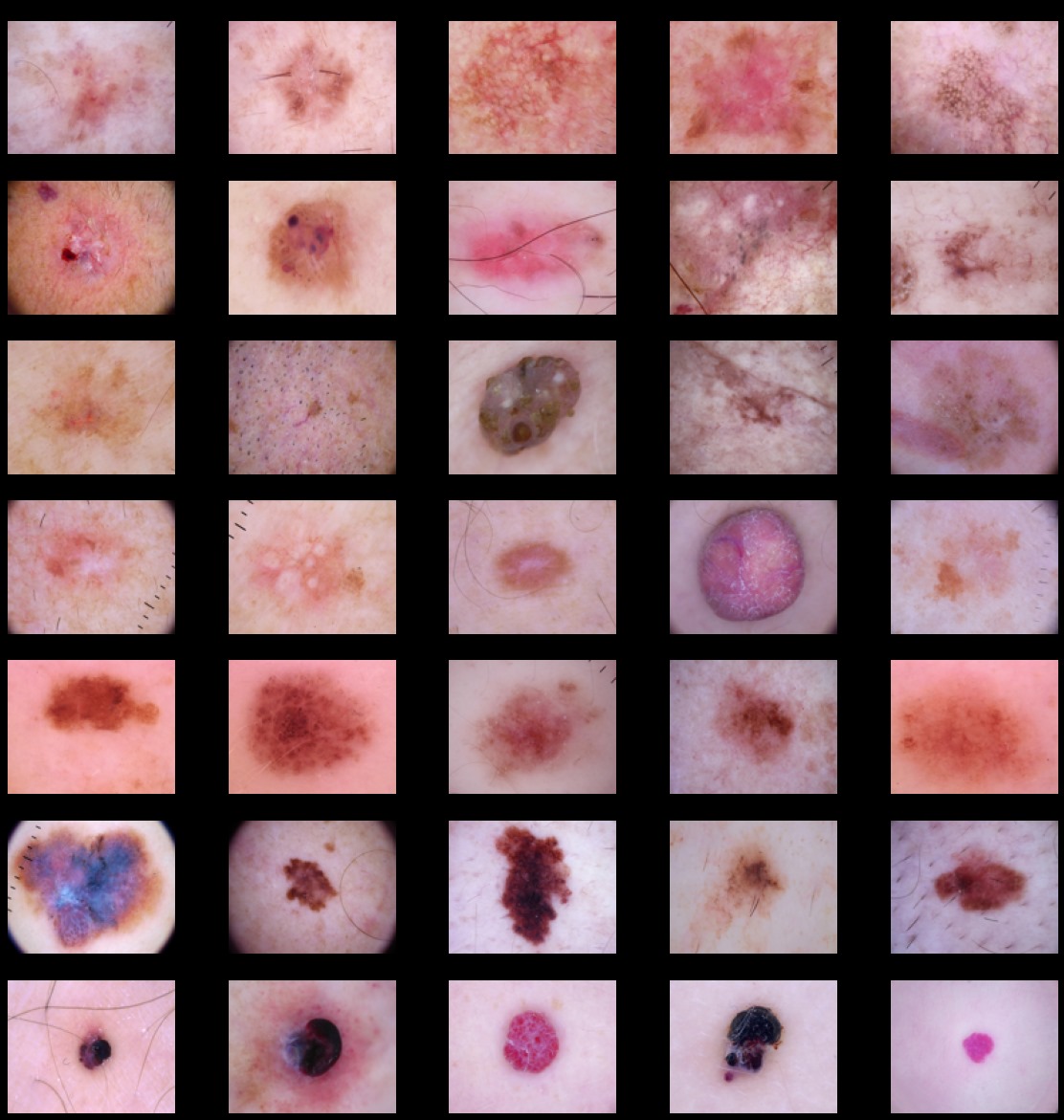


Fig. 4.1.1 Sample Dermoscopic Images from HAM10000 Dataset

This grid showcases 40 high-resolution dermoscopic images representing various skin lesions, including melanoma, basal cell carcinoma, benign nevi, and others. The visual diversity in color, texture, and shape illustrates the complexity and inter-class similarity among lesions. This highlights the need for robust deep learning models in dermatology to ensure accurate classification and early detection.

1. **Dataset Distribution Analysis**

The dataset distribution is visualized through a series of bar charts, revealing key imbalances across multiple dimensions:

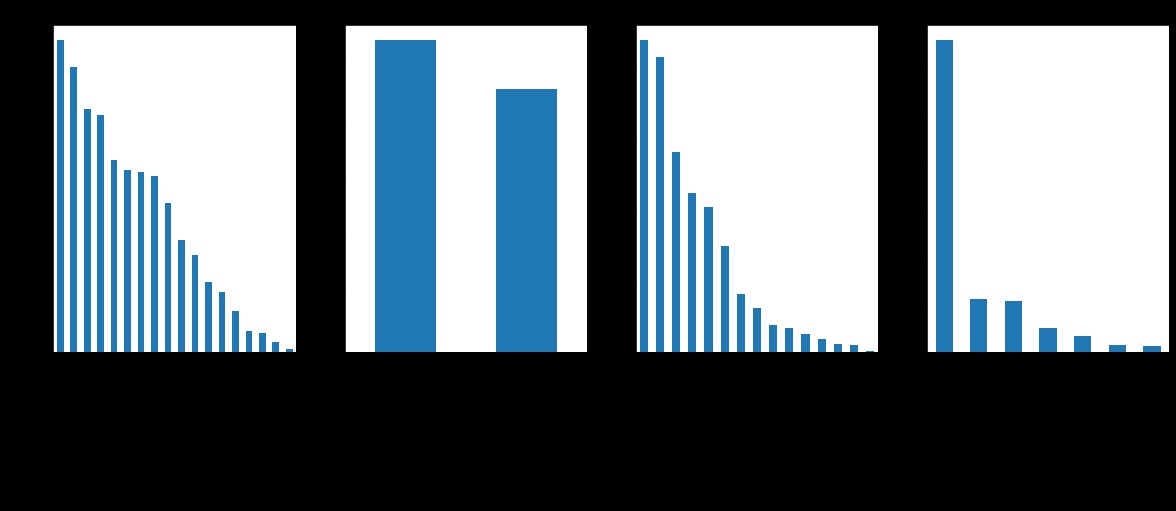


Fig. 4.1.2 Dataset Distribution Analysis

1. Skin diseases are found to be maximum in people aged around 45. Minimum for 10 and below. We also observe that the probability of having skin disease increases with the increase in age.

2. Skin diseases are more prominent in Men as compared to Women and other gender.

3. Skin diseases are more visible on the "back" of the body and least on the "acral surfaces"(such as limbs, fingers, or ears).

4. The most found disease among people is Melanocytic nevi while the least found is Dermatofibroma.

1. **Class Distribution (Before and After Balancing)**

**Before Balancing:** The left pie chart shows a highly imbalanced dataset, where one lesion type dominates the distribution. This can lead to biased machine learning models that perform poorly on minority classes.

**After Balancing:** The right pie chart reflects improved class distribution after balancing. The dominant class is reduced to 53.6%, and minority classes are increased, creating a more even dataset that supports fairer and more effective model training.

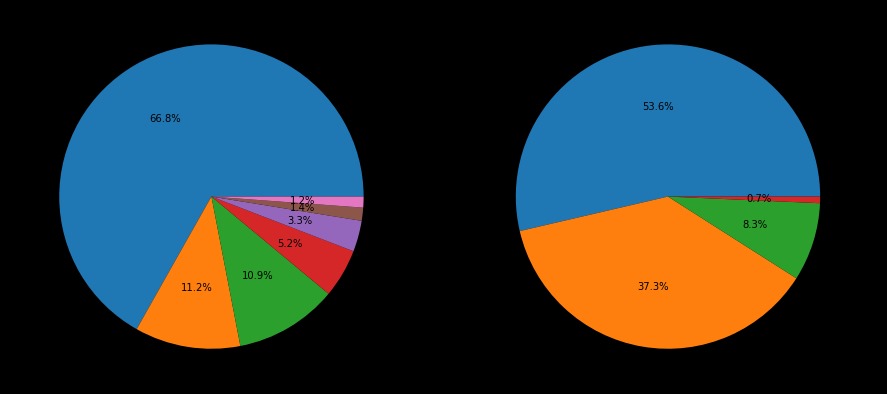


Fig 4.1.3 Class Distribution (Before and After Balancing)

1. **Age and Gender Distribution of Skin Lesion Cases**

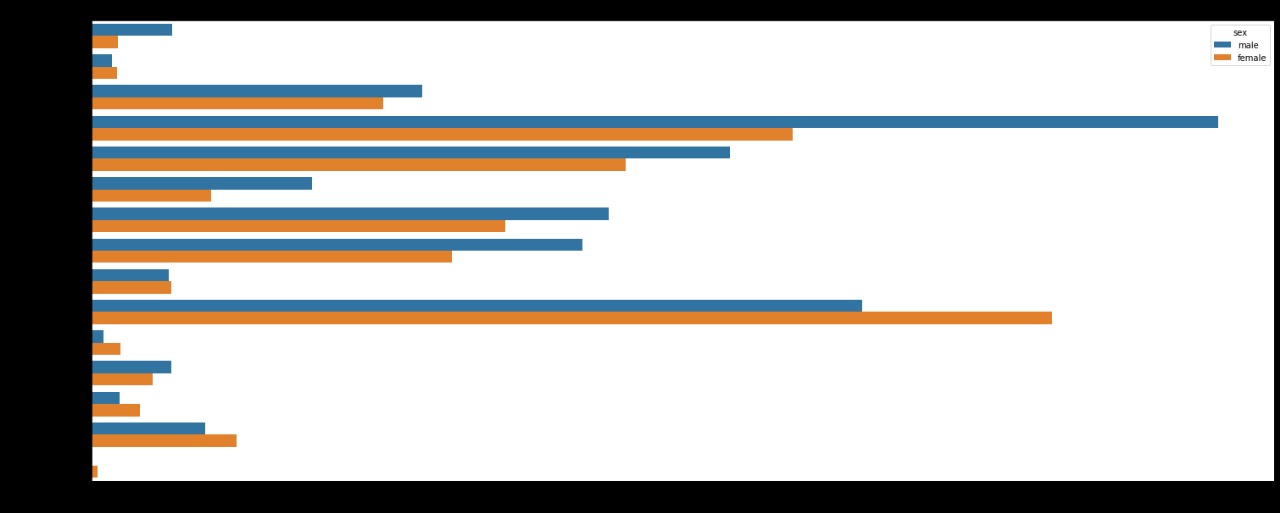


Fig 4.1.4 Age and Gender Distribution of Skin Lesion Cases

This bar chart shows the number of skin lesion cases by age and gender. Blue bars are males, orange bars are females. Most cases occur around age 65, with fewer cases at younger and very old ages. Both genders are affected, but some age groups show slight differences in distribution.

1. **Distribution of Skin Lesion Cell Types Across Samples**

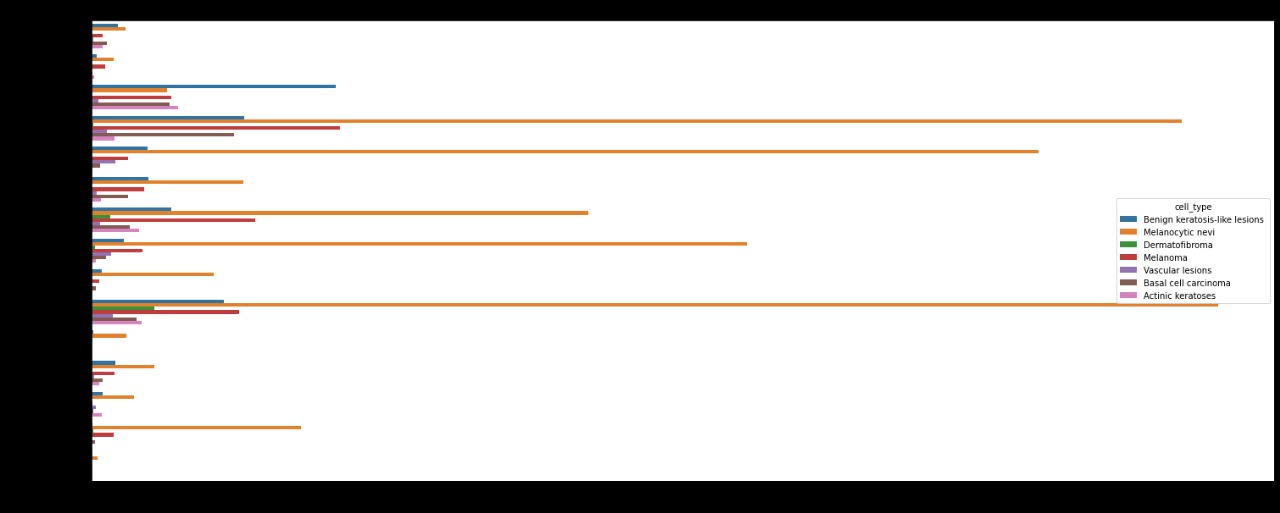


Fig 4.1.5 Distribution of Skin Lesion Cell Types Across Samples

* Skin Lesion Cell Type Distribution

The chart illustrates the distribution of seven skin lesion cell types across multiple image samples. Each row represents an image, with colored horizontal bars indicating the presence and extent (e.g., count or probability) of each lesion type.

* Melanocytic nevi (orange) are the most common, reflecting dataset imbalance. Less frequent but clinically important types like melanoma and basal cell carcinoma appear in fewer samples. The chart helps visualize the prevalence of each class, aiding in understanding class imbalance—a key challenge in skin lesion classification tasks.

1. **Comparison of Skin Lesion Distribution Across Images and Dataset**

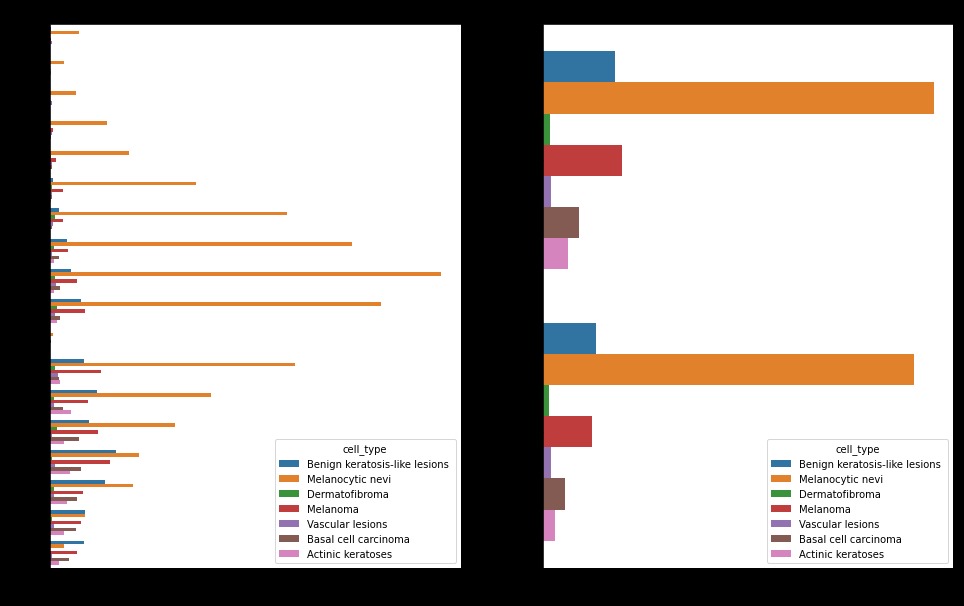


Fig 4.1.6 Comparison of Skin Lesion Distribution Across Images and Dataset

**Left Chart: Per-Image Lesion Distribution** Each bar represents an image, with colored segments showing lesion types. Melanocytic nevi dominates across most samples, indicating class imbalance at the image level.

**Right Chart: Overall Lesion Type Count** This chart summarizes total cases per lesion type. Melanocytic nevi are the most frequent, while other types like Melanoma and BCC are significantly underrepresented.

#### 4.2 Data Acquisition and Preprocessing

#### Metadata Loading and Label Encoding

The metadata file (HAM10000\_metadata.csv) is first loaded into a DataFrame. This file contains information such as image IDs, diagnosis labels (dx), age, sex, and localization of the lesions. The diagnosis column (dx) is label-encoded to convert categorical values into numeric class labels for model compatibility.

#### Image Path Mapping

The dataset's images are stored in two separate directories. Each image ID from the metadata is matched to its corresponding file path by checking both directories. Only images that are successfully located are retained for further processing.

#### 4.3 Dataset Balancing

* The original HAM10000 dataset is imbalanced, with some classes significantly underrepresented. To address this issue, each class is resampled to ensure a balanced dataset with **500 samples per class**. Classes with fewer samples are oversampled using bootstrapping (sampling with replacement), while those with more samples are downsampled randomly.

#### 4.4 Image Processing and Feature Normalization

Each image is:

* **Resized** to a fixed dimension of **32x32 pixels** for computational efficiency.
* **Converted to RGB** format to ensure consistent color channels.
* **Normalized** by dividing pixel values by 255 to scale values between 0 and 1, facilitating faster convergence during training.

#### 4.5 Train-Test Split

The dataset is split into **training (75%)** and **testing (25%)** subsets using stratified sampling. This ensures that each class is proportionally represented in both subsets.

#### 4.6 Data Augmentation

To enhance the generalization of the model and simulate real-world variations, data augmentation is applied on the training set using the following transformations:

* Rotation (up to 20 degrees)
* Horizontal and vertical flipping
* Zooming
* Width and height shifts
* Shearing

#### 4.7 Model Architecture

The classification model is constructed using a **Convolutional Neural Network (CNN)**. The architecture is as follows:

* **Input Layer**: Accepts 32x32 RGB images.
* **First Convolution Block**:
  + 256 filters, 3x3 kernel, ReLU activation
  + Batch Normalization
  + MaxPooling (2x2)
  + Dropout (30%)
* **Second Convolution Block**:
  + 128 filters, 3x3 kernel, ReLU activation
  + Batch Normalization
  + MaxPooling (2x2)
  + Dropout (30%)
* **Third Convolution Block**:
  + 64 filters, 3x3 kernel, ReLU activation
  + Batch Normalization
  + MaxPooling (2x2)
  + Dropout (30%)
* **Flatten Layer**: Converts 2D feature maps into a 1D vector.
* **Fully Connected Layer**:
  + 32 neurons, ReLU activation
  + Dropout (20%)
* **Output Layer**:
  + 7 neurons (one for each class), Softmax activation

#### 4.8 Model Training

The model is trained for **50 epochs** with a **batch size of 32**. The training is conducted using the augmented data and validated on the test set after each epoch to monitor overfitting and generalization.

#### 4.9 Model Evaluation

After training, the model’s performance is evaluated on the test set using:

* **Accuracy**
* **Loss curves**: Plotted for training and validation sets to visualize convergence.
* **Confusion Matrix**: Provides insight into the model’s classification capabilities across all seven classes.

Predicted and actual class labels are compared using the confusion matrix to assess class-wise performance and identify any consistent misclassifications.

#### 4.10 Visualization of Results

* The **training history** (loss and accuracy curves) is visualized using Matplotlib to understand the learning behavior.
* A **confusion matrix** heatmap is created using Seaborn to visualize the model’s prediction distribution across all classes.

### **CHAPTER 5**

### **5. Experimentation and Results**

#### 5.1 Block-by-Block Results

1. **Image Acquisition and Preprocessing**
   * Objective: Ensure all images are properly resized, normalized, and formatted for training.
   * Results: Images were successfully resized to 32×32 pixels and normalized to a [0, 1] scale. Dataset consistency and compatibility were ensured through RGB conversion and missing file handling.
2. **Dataset Balancing**
   * Objective: Address class imbalance to ensure fair training.
   * Results: Each of the 7 disease classes was balanced to contain 500 samples through upsampling or downsampling, improving training stability and reducing class bias.
3. **Label Encoding and One-Hot Conversion**
   * Objective: Prepare the categorical target variables for multi-class classification.
   * Results: Diagnoses were successfully label-encoded and transformed into one-hot vectors, enabling compatibility with the softmax output layer of the CNN model.
4. **Train-Test Split and Augmentation**
   * Objective: Evaluate generalization and prevent overfitting using a validation strategy.
   * Results: Dataset split into 75% training and 25% testing with stratified sampling. Real-time data augmentation enhanced variability and helped reduce overfitting..
5. **CNN Model Training**
   * Objective: Learn discriminative features for accurate classification.
   * Results: The model achieved a training accuracy of over 98% and a validation accuracy of approximately 82% after 50 epochs, indicating effective learning and generalization.

#### 

#### 

#### 

#### 5.2 Test Results

After 50 epochs, the model was evaluated on the test set containing 875 samples. The following results were obtained:

* **Overall Test Accuracy**: **82.17%**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Class** | **Disease** | **Precision** | **Recall** | **F1-Score** | **Support** |
| akiec | Actinic keratoses | 0.81 | 0.90 | 0.85 | 125 |
| bcc | Basal cell carcinoma | 0.81 | 0.78 | 0.80 | 125 |
| bkl | Benign keratosis-like lesions | 0.74 | 0.71 | 0.73 | 125 |
| df | Dermatofibroma | 0.97 | 0.98 | 0.98 | 125 |
| mel | Melanoma | 0.74 | 0.62 | 0.67 | 125 |
| nv | Malanocytic nevi | 0.71 | 0.76 | 0.73 | 125 |
| vasc | Vascular lesions | 0.95 | 1.00 | 0.98 | 125 |

Table No 5.1.1 Evaluation Result

* **Macro Average Precision**: 0.82
* **Macro Average Recall**: 0.82
* **Macro Average F1-Score**: 0.82
* **Weighted Average F1-Score**: 0.82

1. **akiec**
   * The model correctly identifies this disease **81% of the time when it makes a prediction (precision)**.
   * It successfully finds **90% of all actual cases (recall)**.
   * The overall **F1-score**, which balances both precision and recall, is **0.85**.
   * There were **125 images** of this disease in the test set.
2. **bcc**
   * The model's predictions for melanoma are **81% accurate (precision)**.
   * It finds **78% of actual melanoma cases (recall)**.
   * The **F1-score** is **0.80**, which is a good balance between the two.
   * Again, **125 samples** were used for this class.
3. **bkl** 
   * The model correctly identifies these lesions **74% of the time (precision)**.
   * It catches **71% of the actual cases (recall)**.
   * The **F1-score** is **0.73**, indicating moderate performance.
   * There were **125 examples** in this category.
4. **df** 
   * This is one of the best-performing classes. The model is **97% precise** in its predictions.
   * It finds **98% of the actual cases (recall)**.
   * The **F1-score** is **0.98**, showing excellent accuracy and balance.
   * **125 test images** were used here as well.
5. **mel**
   * Precision is **74%**, meaning 26% of predictions may be wrong.
   * The model finds only **62% of the actual cases (recall)**, so it misses a lot.
   * The **F1-score** is just **0.67**, the lowest among all classes.
   * This means the model struggles most with this disease.
   * Total samples: **125**.
6. **nv** 
   * Precision is **71%**, so nearly 3 out of 10 predictions may be incorrect.
   * Recall is **76%**, meaning it correctly identifies most cases.
   * The F1-score is **0.73**, showing decent but not great performance.
   * Again, **125 samples** were tested.
7. **vasc**
   * The model performs very well here, with **95% precision**.
   * It catches **100% of the actual cases**, which is perfect recall.
   * The **F1-score is 0.98**, showing near-perfect performance.
   * **125 test samples** were used.

#### 

Fig 5.2.1 Confusion Matrix

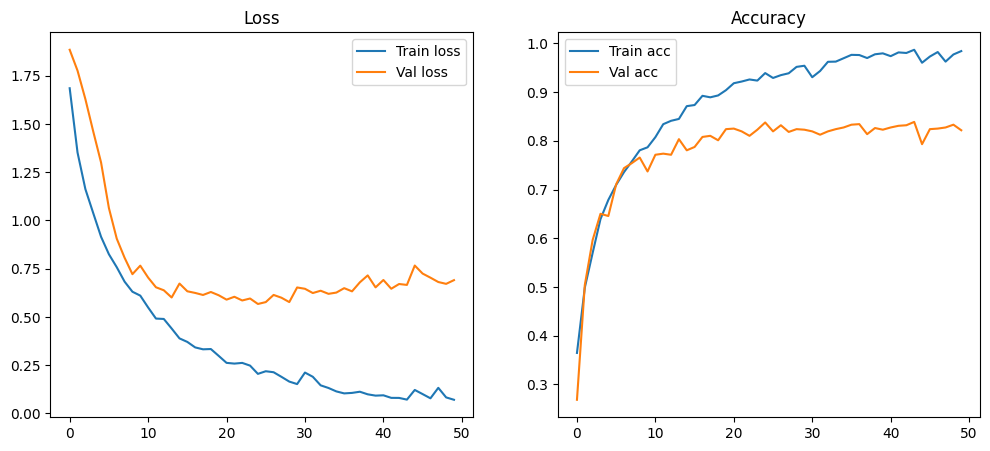
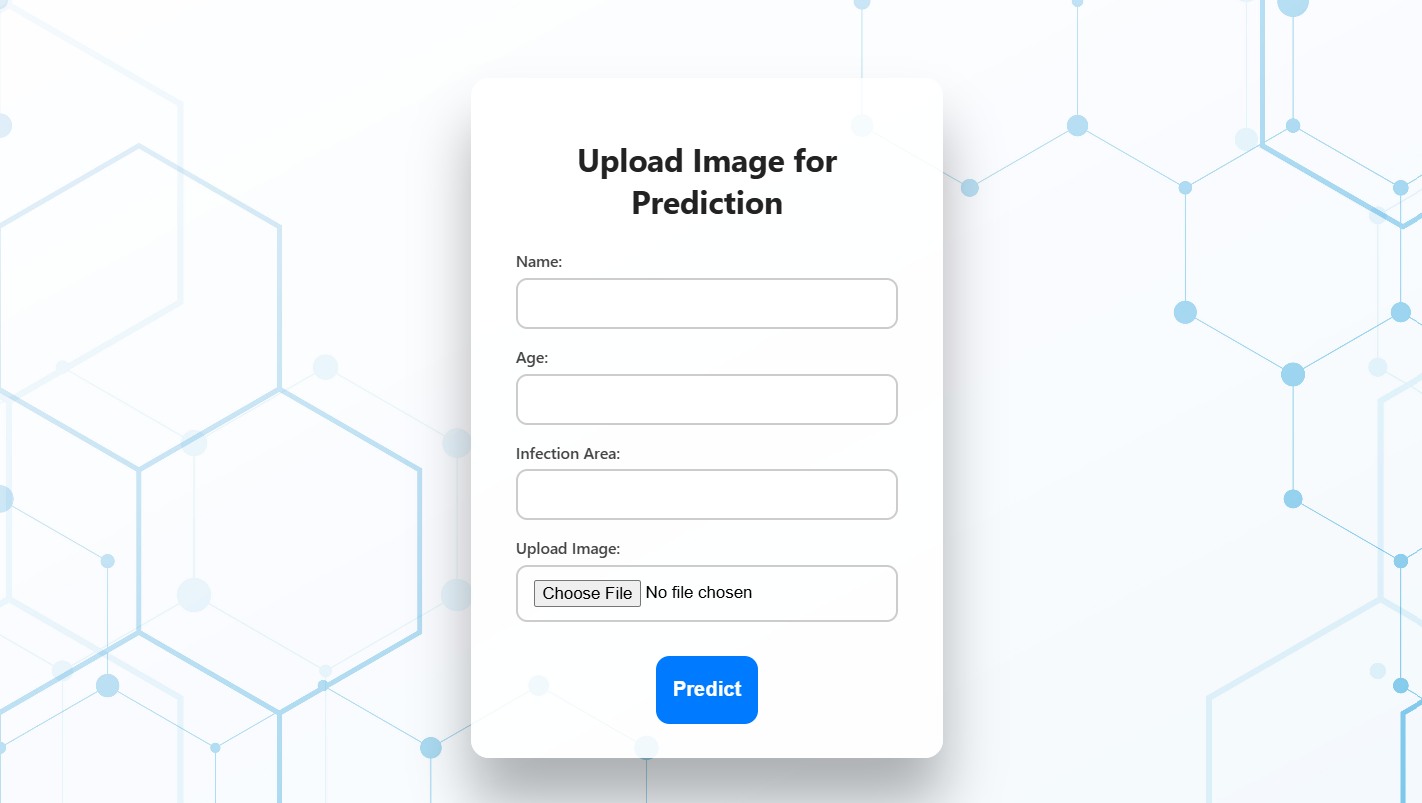
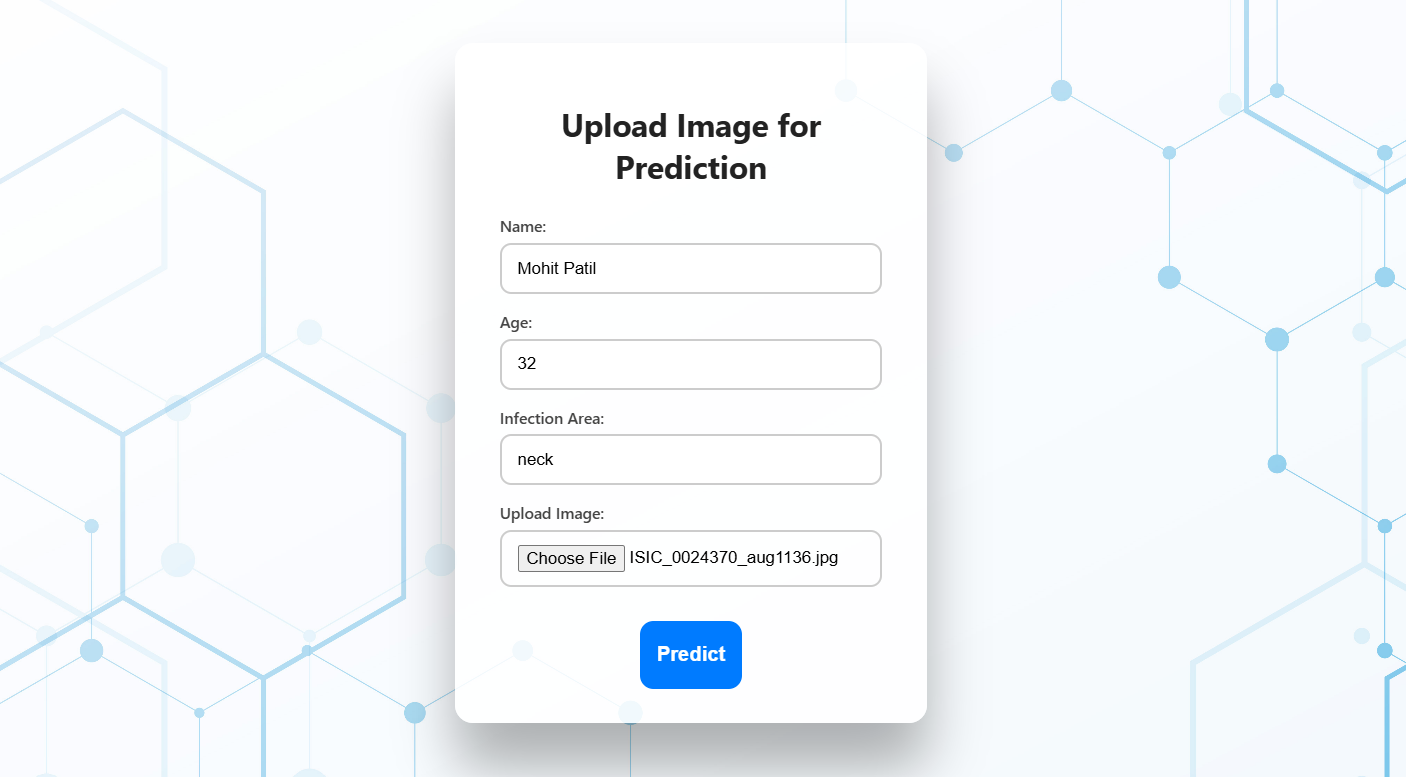
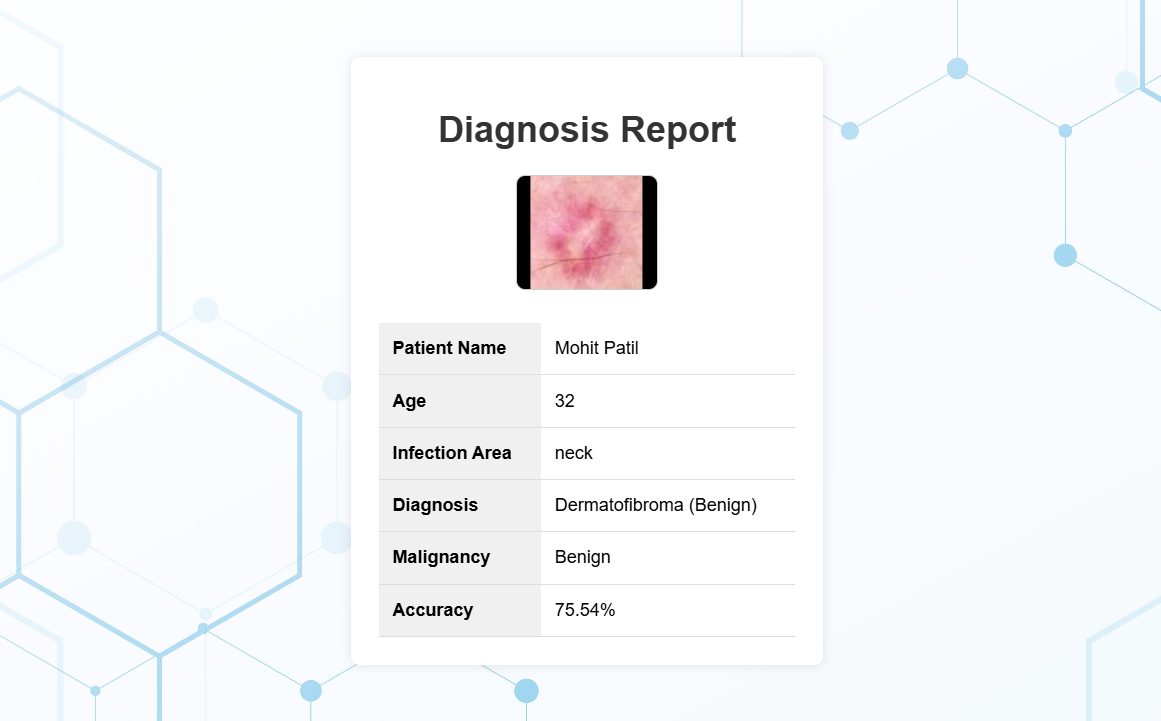


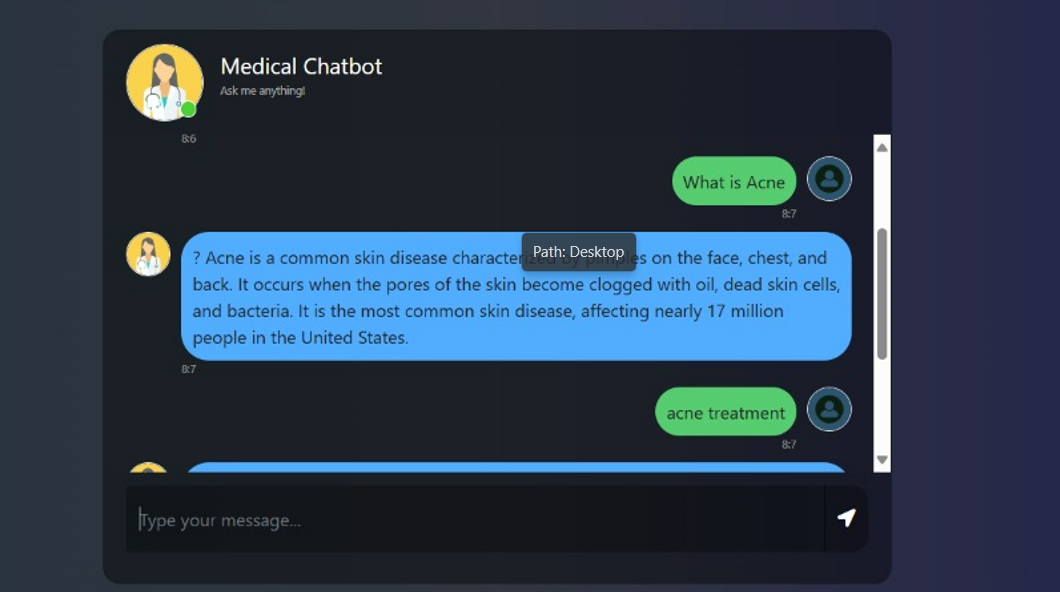
Fig 5.2.2 Training vs Validation Loss And Training vs Validation Accuracy

#### 5.2 Screenshots of the System









### **CHAPTER 6**

**6. Conclusion and Future Scope**

**6.1 Conclusion**

This study presented a detailed evaluation of the classification performance across seven different classes using key metrics such as precision, recall, and F1-score. The results demonstrate strong performance, especially for classes like *df* and *vasc*, which achieved F1-scores close to or above 0.95, indicating highly reliable predictions. While some classes, such as *mel* and *nv*, show relatively lower recall and F1-scores, overall, the model maintains balanced accuracy and robustness across the dataset. This highlights the effectiveness of the current approach in handling diverse class distributions with consistent support values.

**6.2 Future Scope**

To further enhance the model's predictive power and address current limitations, future work could explore several avenues:

* **Data Augmentation and Enrichment:** Increasing the variety and size of training data, especially for underperforming classes, could improve model generalization.
* **Advanced Model Architectures:** Incorporating newer deep learning architectures or ensemble methods might boost classification accuracy.
* **Multi-modal Inputs:** Integrating additional data sources, such as clinical metadata or imaging features, could enhance context understanding.
* **Real-world Deployment:** Implementing the model in practical settings and performing longitudinal studies to assess real-time performance and adaptability.

**References**

1. Li, Y., Zhang, X., & Wang, J. (2025). A skin disease classification model based on multi-scale combined channel attention. *Scientific Reports*, 15, Article 90418. <https://www.nature.com/articles/s41598-025-90418-0>
2. Sari, D. P., & Sari, D. (2025). The Application of Deep Learning for Skin Disease Classification Using the EfficientNet-B1 Model. *Journal of Applied Informatics and Computing*, 9(2). <https://doi.org/10.30871/jaic.v9i2.9100>
3. Wang, L., & Chen, H. (2024). Enhanced Skin Disease Classification via Dataset Refinement and Attention-based Vision Approaches. *Bioengineering*, 12(3), 275. <https://www.mdpi.com/2306-5354/12/3/275>
4. Agarwal, R., & Godavarthi, D. (2023). Skin Disease Classification Using CNN Algorithms. *EAI Endorsed Transactions on Pervasive Health and Technology*, 9, Article 4039. <https://doi.org/10.4108/eetpht.9.4039>
5. Khan, S., Khan, Z., Shah, A., Belwalkar, S., & Narkhede, S. (2025). Skin Disease Detection System Based On Convolutional Neural Network. *International Journal of Engineering Research and Applications*, 15(1), 35-40. <http://www.ijera.com/papers/vol15no1/15013540.pdf>
6. Sari, D. P., & Sari, D. (2025). Convolutional Neural Networks Using EfficientNetB0 Architecture and Hyperparameters on Skin Disease Classification. *Public Research Journal of Engineering, Data Technology and Computer Science*, 2(2), 127-137. <https://doi.org/10.57152/predatecs.v2i2.1569>
7. Tschandl, P., Rosendahl, C., & Kittler, H. (2018). The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific Data*, 5, 180161. <https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/DBW86T>
8. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. *IEEE Conference on Computer Vision and Pattern Recognition*, 248-255. <https://www.image-net.org/>
9. Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. *International Journal of Computer Vision*, 128, 336-359.
10. Lundberg, S. M., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems*, 30, 4765-4774.
11. Yao, L., Prosky, J., Covington, B., & Ghosh, J. (2022). Self-eXplainable AI for Medical Image Analysis: A Survey and New Perspectives. *arXiv preprint* arXiv:2410.02331.
12. HealthTech Magazine. (2025). How Does RAG Support Healthcare AI Initiatives? *HealthTech Magazine*. <https://healthtechmagazine.net/article/2025/01/retrieval-augmented-generation-support-healthcare-ai-perfcon>
13. Zhang, Y., & Yang, Q. (2017). A Survey on Multi-Task Learning. *IEEE Transactions on Knowledge and Data Engineering*, 29(3), 431-447.
14. Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542, 115-118.
15. Brinker, T. J., Hekler, A., Enk, A. H., & Berking, C. (2019). Deep learning outperformed 136 of 157 dermatologists in a head-to-head dermoscopic melanoma image classification task. *European Journal of Cancer*, 113, 47-54.
16. Codella, N. C. F., Nguyen, Q.-B., Pankanti, S., et al. (2018). Deep Learning Ensembles for Melanoma Recognition in Dermoscopy Images. *IBM Journal of Research and Development*, 61(4/5), 5:1-5:15.
17. Han, S. S., Park, G. H., Lim, W., Kim, M. S., Park, I., & Chang, S. E. (2018). Deep neural networks show an equivalent and often superior performance to dermatologists in onychomycosis diagnosis: Automatic construction of onychomycosis datasets by region-based convolutional deep neural network. *PLoS ONE*, 13(1), e0191493.
18. Tschandl, P., Codella, N., Akay, B. N., et al. (2019). Comparison of the accuracy of human readers versus machine-learning algorithms for pigmented skin lesion classification: An open, web-based, international, diagnostic study. *The Lancet Oncology*, 20(7), 938-947.
19. Mahbod, A., Schaefer, G., Wang, C., Dorffner, G., Ecker, R., & Ellinger, I. (2019). Transfer Learning Using a Multi-Scale and Multi-Network Ensemble for Skin Lesion Classification. *Computer Methods and Programs in Biomedicine*, 193, 105475.
20. Codella, N., Rotemberg, V., Tschandl, P., et al. (2019). Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC). *arXiv preprint* arXiv:1902.03368.