EffiDerm: An Efficient Deep Learning Model for Skin Cancer Prediction

# A PROJECT REPORT

**Submitted by**

## SURENTHRAKUMAR K (9822008008)

## AGIL KANNAN S (9921008001)

## HARSNI M K (9921008014)

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

# BACHELOR OF TECHNOLOGY

in

# INFORMATION TECHNOLOGY

***of***

******

# SCHOOL OF COMPUTING KALASALINGAM ACADEMY OF RESEARCH AND

**EDUCATION**

### (Deemed to be University)

Anand Nagar, Krishnankoil – 626 126

**Academic Year Even Semester (2024-25)**



**School of Computing Department of Information Technology**

**Project Summary**

|  |  |  |
| --- | --- | --- |
| Project Title | EffiDerm: An efficient Deep Learning Model for Skin Cancer Prediction | |
| Project Team Members (Name with Register No) | 1. SURENTHRAKUMAR K (9822008008) 2. AGIL KANNAN S (9921008001) 3. HARSNI M K (9921008014) | |
| Guide Name/Designation | Dr. V. Baby Shalini | |
| Program Concentration Area | Deep Learning | |
| Technical Requirements | EffiDerm is a 1.4M parameter CNN achieving 93.07% accuracy with 47 ms inference time, optimized for mobile use. It uses SMOTE, data augmentation, and focal loss for handling class imbalance and efficient training. | |
| Engineering standards and realistic constraints in these areas: (Refer Appendix in page 4 of this doc.) | | |
| **Area** | **Codes & Standards / Realistic Constraints** | **Tick**  **◻** |
| Economic |  |  |
| Environmental | EffiDerm is designed for deployment in low-resource healthcare environments, enabling real-time skin cancer detection on mobile and edge devices. | **◻** |
| Social |  |  |
| Ethical |  |  |
| Health and Safety |  |  |
| Manufacturability |  |  |
| Sustainability |  |  |

**Realistic Constraints:**

As medical technology advances, researchers are striving to improve early disease detection through AI-powered solutions. While fully autonomous diagnostic systems remain under development, AI-assisted tools like EffiDerm are emerging to support dermatologists. Such systems must accurately analyze skin lesions, differentiate between benign and malignant cases, and adapt to diverse skin tones and imaging conditions.

Key challenges include maintaining high accuracy across varied datasets, ensuring real-time performance on low-power devices, and complying with strict medical regulations to guarantee patient safety and data privacy.

**Engineering standards:** In developing EffiDerm, the following standards were referenced.

As medical technology advances, researchers are striving to improve early disease detection through AI- powered solutions. While fully autonomous diagnostic systems remain under development, AI-assisted tools like EffiDerm are emerging to support dermatologists.

* Accuracy
* Energy Efficiency
* Real-time Processing
* Data Variability
* Safety & Privacy
* Explainable

The primary goal of EffiDerm is to provide an accurate, lightweight, and energy-efficient AI solution for early skin cancer detection that seamlessly integrates into clinical workflows. By achieving high classification accuracy (93.07%) with minimal computational resources (1.4M parameters), the system aims to support dermatologists in rapid diagnosis while ensuring compliance with medical standards (FDA/CE, HIPAA/GDPR) and fairness across diverse patient populations. Ultimately, EffiDerm strives to democratize access to reliable skin cancer screening, particularly in resource-limited settings.

# DECLARATION BY THE STUDENT

We affirm that the project work title **“EffiDerm: An Efficient Deep Learning Model for Skin Cancer Prediction”** being submitted impartial fulfilment for the award of the degree of **Bachelor of Technology in Information Technology** is the Original work Carried out by us. It has not formed the part of any other project work submitted for award of any degree or diploma, either in this or any other University.

SURENTHRAKUMAR K

(9822008008)

AGIL KANNAN S

(9921008001)

HARSNI M K (9921008014)



# SCHOOL OF COMPUTING DEPARTMENT OF INFORMATION TECHNOLOGY

**BONAFIDE CERTIFICATE**

Certified that this project report **“EffiDerm: An Efficient Deep Learning Model for Skin Cancer Prediction’’** i**s** the bonafide work of **“SURENTHRAKUMAR K (9822008008), AGIL KANNAN S (9921008001)** and **HARSNI M K (9921008014)”** who carried out the project work under my supervision

|  |  |
| --- | --- |
| **Dr. V. Baby Shalini** | **Dr. N.C. Brintha** |
| **Supervisor** | **Associate Professor** |
| **Associate Professor** | **Head of the Department** |
| **Department of IT** | **Department of IT** |
| Kalasalingam Academy of Research and Education | Kalasalingam Academy of Research and Education |
| Krishnankoil-626126. | Krishnankoil-626126. |

Submitted for the project viva-voce examination held on ………………

**Internal Examiner External Examiner**

# ACKNOWLEDGEMENT

We thank God almighty for giving us such tremendous opportunity and support through the Way the **KALASALINGAM ACADEMY OF RESEARCHAND EDUCATION**.

We express my hearty thanks to our chancellor “**Ilayavallal**” **Thiru**. **Dr. K. Sridharan &** Vice President **Dr. S. Shasi Anand** for providing all the required facilities for the successful completion of our project.

We endow my sincere thanks to Vice-Chancellor **Dr. S. Narayanan** for his moral support through this project.

We thank our Registrar **Dr. V. Vasudevan** who has always served as an inspiration for us to perform our institutes name and recognition.

We wish to express our sincere thanks to our respected Head of the Department **Dr. N.C. Brintha, Associate Professor**, whose moral support encouraged us to process through our project work successfully.

We offer our sincerest gratitude to our Supervisor, **Dr. V. Baby Shalini, Associate Professor,** for patience, motivation, enthusiasm and immense knowledge.

We are extremely grateful to our Project Coordinator, **Dr. D. Prem Raja, Associate Professor** for constant encouragement in the completion of the project.

Finally, we thank all, our Parents, Teaching Faculty, Non-Teaching Faculty and our friends for their support and peers for having stood by us and helped us to complete.

# ABSTRACT

Skin cancer is a major global health concern, with its incidence rates increasing at an alarming pace. Among the various forms of skin cancer, melanoma is notably the most aggressive and fatal, making early detection crucial for effective treatment and improved patient survival. Traditional diagnostic methods—primarily reliant on dermatological expertise and dermatoscopic evaluation—often suffer from subjectivity, inter-observer variability, and limited accessibility in rural or under-resourced healthcare settings. To bridge this diagnostic gap and empower frontline medical care with intelligent tools, this project proposes EffiDerm, a robust, interpretable, and mobile-friendly deep learning model for skin cancer prediction. EffiDerm is built on a lightweight Convolutional Neural Network (CNN) architecture, optimized to run efficiently on mobile and embedded devices without compromising performance. Despite comprising only 1.4 million parameters, EffiDerm achieves a high classification accuracy of 93.07% on the HAM10000 dataset, a comprehensive collection of over 10,000 dermatoscopic images spanning seven distinct skin lesion classes, including both benign and malignant conditions.

A major challenge addressed in this project is the class imbalance prevalent in medical imaging datasets. To mitigate this, the model employs Synthetic Minority Oversampling Technique (SMOTE) and focal loss functions, ensuring balanced learning across all lesion types. Furthermore, data augmentation strategies such as random rotation, flipping, and brightness variation enhance the model’s ability to generalize across diverse imaging conditions. The architecture also integrates Squeeze-and-Excitation (SE) blocks and depthwise separable convolutions, reducing computational complexity while enhancing feature extraction efficiency. In addition to predictive performance, explainability and clinical trust are at the forefront of EffiDerm’s design. Using Gradient-weighted Class Activation Mapping (Grad-CAM), the model provides heatmaps that highlight lesion regions most influential to its predictions. This not only increases transparency but also supports dermatologists in validating AI decisions, facilitating a collaborative diagnostic workflow.

EffiDerm is fully aligned with modern standards for medical AI solutions, including HIPAA/GDPR for data privacy and FDA/CE guidelines for clinical reliability. It demonstrates practical feasibility for deployment in mobile health (mHealth) applications, clinical decision support systems, kiosk-based diagnostics, and rural teledermatology programs. With a minimal inference time of 47 milliseconds and a compact memory footprint of 5.6 MB, EffiDerm is engineered for real-time use in low-resource settings. In summary, EffiDerm exemplifies how a thoughtfully designed deep learning model can address real-world healthcare challenges by combining accuracy, efficiency, fairness, and interpretability. This project sets a precedent for scalable AI-based medical diagnostics and opens avenues for future developments including federated learning, multimodal analysis, and integration into IoT-enabled smart healthcare ecosystems.

# TABLE OF CONTENTS

| Chapter No. | Title | Page No. |
| --- | --- | --- |
|  | ABSTRACT | VII |
|  | LIST OF FIGURES | XII |
|  | LIST OF ABBREVIATIONS | XIII |
| 1 | INTRODUCTION | 1 |
|  | 1.1 General | 1 |
|  | 1.2 Existing System | 2 |
|  | 1.3 Proposed System | 2 |
| 2 | LITERATURE SURVEY | 3 |
|  | 2.1 Overview Of Skin Cancer Detection And Classification | 3 |
|  | 2.2 Existing Methods And Models | 3 |
|  | 2.3 Limitations In Existing Systems | 3 |
|  | 2.4 Research Gap Identified | 4 |
|  | 2.5 Literature Summary Table | 5 |
| 3 | OBJECTIVES | 6 |
|  | 3.1 Main Objective | 6 |
|  | 3.2 Specific Objectives | 6 |
| 4 | PRELIMINARY ANALYSIS & WORKFLOW OVERVIEW | 8 |
|  | 4.1 Introduction | 8 |
|  | 4.2 System Workflow | 8 |
|  | 4.3 Dataset Visualization | 10 |
|  | 4.4 Output Prediction Visualization | 10 |
|  | 4.5 Early Observations | 10 |
| 5 | SYSTEM REQUIREMENTS | 12 |
|  | 5.1 Software Used | 12 |
|  | 5.2 Libraries and Tools | 12 |
|  | 5.3 Dataset Description - HAM10000 | 13 |
| 6 | METHODOLOGY | 17 |
|  | 6.1 Introduction | 17 |
|  | 6.2 System Architecture | 17 |
|  | 6.3 Data Preprocessing | 18 |
|  | 6.4 Data Augmentation | 19 |
|  | 6.5 Model Architecture | 19 |
|  | 6.6 Training Strategy | 19 |
|  | 6.7 Evaluation Strategy | 20 |
|  | 6.8 Visualization & Interpretability | 21 |
| 7 | CODE IMPLEMENTATION | 22 |
|  | 7.1 Importing Required Libraries | 22 |
|  | 7.2 Loading and Preparing the Dataset | 23 |
|  | 7.3 Handling Class Imbalance with Oversampling | 23 |
|  | 7.4 Train-Test and One-Hot Encoding | 24 |
|  | 7.5 Data Augmentation | 24 |
|  | 7.6 Model Architecture Design | 24 |
|  | 7.7 Model Compilation | 25 |
|  | 7.8 Model Training | 25 |
|  | 7.9 Training Visualization | 25 |
|  | 7.10 Model Evaluation | 25 |
|  | 7.11 Confusion Matrix | 26 |
|  | 7.12 Visualization Predictions | 26 |
|  | 7.13 Saving the Model | 27 |
| 8 | EXPERIMENTAL RESULTS AND EVALUATION | 28 |
|  | 8.1 Introduction | 28 |
|  | 8.2 Training and Validation Performance | 28 |
|  | 8.3 Training and Validation Loss | 28 |
|  | 8.4 Confusion Matrix Analysis | 30 |
|  | 8.5 Sample Predictions Visualization | 32 |
|  | 8.6 Performance Metric | 33 |
|  | 8.7 Comparison with Existing CNN Models | 34 |
|  | 8.8 Summary of Findings | 34 |
| 9 | DISCUSSION | 36 |
|  | 9.1 Overview | 36 |
|  | 9.2 Impact of Preprocessing Techniques | 36 |
|  | 9.3 Role of Data Augmentation | 36 |
|  | 9.4 Effectiveness of the Model Architecture | 36 |
|  | 9.5 Significance of Evaluation Metrics | 37 |
|  | 9.6 Comparison with Existing Models | 37 |
|  | 9.7 Clinical Applicability and Real-World Use | 37 |
|  | 9.8 Summary | 38 |
| 10 | APPLICATIONS | 39 |
|  | 10.1 Introduction | 39 |
|  | 10.2 Mobile Health (mHealth) Applications | 39 |
|  | 10.3 Clinical Decision Support | 39 |
|  | 10.4 Use in Rural and Under-resourced Clinics | 40 |
|  | 10.5 Integration into Kiosks and Screening Terminals | 40 |
|  | 10.6 Academic and Research Applications | 40 |
|  | 10.7 Future IoT and Smart Wearable Integration | 40 |
|  | 10.8 Summary | 40 |
| 11 | LIMITATIONS | 41 |
|  | 11.1 Introduction | 41 |
|  | 11.2 Dataset Constraints | 41 |
|  | 11.3 Limited Real-World Testing | 41 |
|  | 11.4 Model Interpretability | 41 |
|  | 11.5 Clinical Scope and Decision Support Limitations | 42 |
|  | 11.6 Dependence on Image Quality | 42 |
|  | 11.7 Security and Ethical Concerns | 42 |
|  | 11.8 Summary | 42 |
| 12 | FUTURE SCOPE | 43 |
|  | 12.1 Introduction | 43 |
|  | 12.2 Integration of Explainable AI (XAI) | 43 |
|  | 12.3 Expansion to Real-World Images | 43 |
|  | 12.4 Use of Multimodal Data | 43 |
|  | 12.5 Deployment on Mobile and Edge Devices | 44 |
|  | 12.6 Clinical Testing and Collaboration | 44 |
|  | 12.7 Expansion to Other Skin Conditions | 44 |
|  | 12.8 Incorporating Federated Learning for Privacy | 44 |
|  | 12.9 Automation of Patient Reports | 44 |
|  | 12.10 Summary | 45 |
| 13 | CONCLUSION | 46 |
|  | 13.1 Summary of the Work | 46 |
|  | 13.2 Achievements | 46 |
|  | 13.3 Contributions | 47 |
|  | 13.4 Closing Remarks | 47 |
|  | REFERENCES | 48 |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **S. NO.** | **TITLE** | **PAGE NO** |
| 4.1 | Workflow of Effiderm System | 8 |
| 4.2 | Sample Input Images from HAM10000 Dataset | 8 |
| 4.3 | Model Predictions Samples | 10 |
| 8.1 | Training vs Validation Accuracy | 28 |
| 8.2 | Training vs Validation Loss | 28 |
| 8.3 | Confusion Matrix | 28 |
| 8.4 | Sample Input Images | 30 |
| 8.5 | Predicted Outputs | 32 |

**LIST OF ABBREVIATIONS**

|  |  |
| --- | --- |
| AI | Artificial Intelligence |
| CNN | Convolutional Neural Network |
| SE | Squeeze-and-Excitation |
| Grad-CAM | Gradient-weighted Class Activation Mapping |
| SMOTE | Synthetic Minority Oversampling Technique |
| HAM10000 | Human Against Machine with 10,000 training images |
| CDSS | Clinical Decision Support System |
| mHealth | Mobile Health |
| GPU | Graphics Processing Unit |
| CSV | Comma-Separated Values |
| RGB | Red Green Blue |
| PIL | Python Imaging Library |
| IDE | Integrated Development Environment |
| ReLU | Rectified Linear Unit |
| VS Code | Visual Studio Code |
| IoT | Internet of Things |
| TPU | Tensor Processing Unit |

**CHAPTER 1 INTRODUCTION**

Medical technology has evolved significantly in recent years, particularly in the domain of diagnostic imaging and intelligent decision-making systems. Similar to how advanced sensors have enhanced automotive safety by detecting physical obstacles and enabling autonomous braking, the medical field has witnessed the rise of artificial intelligence (AI) and deep learning as transformative tools. These technologies have made it possible to detect diseases at earlier stages, offering enhanced precision and improved patient outcomes.

Among the most impactful applications is the use of deep learning in **dermatology**, especially for **skin cancer detection**. Skin cancer, particularly **melanoma**, is a life-threatening condition where early and accurate diagnosis is critical. Just as ultrasonic sensors in automobiles prevent accidents by anticipating threats, AI-powered image classifiers can identify malignant lesions before they progress, acting as virtual diagnostic assistants for clinicians. In countries like India, where access to specialized dermatological care is limited in many rural regions, an AI- based diagnostic tool can drastically improve accessibility and reduce misdiagnoses.

This project proposes the development of an intelligent, lightweight deep learning model named **EffiDerm** for real-time skin lesion classification. Designed with efficiency, mobility, and accessibility in mind, EffiDerm mimics the proactive role of safety systems in vehicles— by identifying potential health risks through dermoscopic image analysis and alerting physicians early enough to take life-saving action.

## GENERAL

Skin cancer is one of the most common and rapidly increasing forms of cancer across the globe. Among its various types, melanoma is the deadliest, responsible for the majority of skin cancer- related deaths, despite being less frequent. Early and accurate detection of melanoma and other skin lesions is critical to improving patient outcomes. Traditional diagnostic methods depend heavily on visual inspections by dermatologists, which are inherently subjective and prone to variability. Studies have shown more than 25% inter-rater disagreement among dermatologists, which can lead to delays or misclassification.

In recent years, artificial intelligence (AI), particularly deep learning, has revolutionized the field of medical image analysis. Convolutional Neural Networks (CNNs) have shown significant success in identifying patterns in complex visual data, making them ideal for applications in skin lesion classification. However, the deployment of these models in real- world scenarios, especially in resource-constrained settings, requires models that are not only accurate but also lightweight, fast, and energy-efficient. This project addresses this need by designing a compact and efficient CNN-based model named **EffiDerm** for accurate and real- time skin lesion classification.

## EXISTING SYSTEM

Various deep learning architectures have been proposed and implemented for automated skin lesion classification. Some of the most widely used models include ResNet-50, MobileNetV3, and EfficientNet-B0. While these architectures achieve high classification accuracy, they come with several limitations that hinder their real-time deployment in clinical environments.

One major limitation is computational complexity. Models like ResNet-50 contain over 23 million parameters, requiring significant computational power and memory, making them unsuitable for mobile devices or low-power clinical setups. Furthermore, existing models often fail to address the problem of class imbalance found in datasets like HAM10000, where the most frequent class can outnumber the rarest class by a ratio of 58:1. This skew leads to biased models that underperform on rare but critical categories such as melanoma.

Additionally, most existing systems are not optimized for mobile or embedded deployment. They are typically developed in research environments with high-performance GPUs, and their practical integration into real-time diagnostic systems is rarely considered. The lack of lightweight, accurate, and interpretable models makes it difficult to deploy these solutions in under-resourced hospitals or rural clinics.

## PROPOSED SYSTEM

To overcome the limitations of existing methods, this project proposes EffiDerm, a lightweight and efficient Convolutional Neural Network (CNN) model for skin cancer classification. EffiDerm is specifically designed for real-time deployment on mobile and edge devices, making it suitable for use in remote and low-resource environments.

The proposed model consists of only 1.4 million parameters—63 times fewer than ResNet- 50—while achieving a classification accuracy of **93.07%** on the HAM10000 dataset. It utilizes advanced techniques such as **SMOTE** for class balancing, **data augmentation** for better generalization, and **focal loss** to prioritize learning from hard-to-classify samples. Additionally, the model incorporates depthwise separable convolutions and Squeeze-and-Excitation (SE) blocks to reduce complexity and enhance feature extraction.

EffiDerm also supports explainability through **Grad-CAM** visualizations, allowing clinicians to interpret the model’s decisions. The system is designed to assist dermatologists in diagnosing skin cancer early, reduce misclassification risks, and make skin cancer screening more accessible, especially in underserved areas. The ultimate goal is to bridge the gap between high- performance AI models and their real-world deployment in the healthcare domain.

# CHAPTER 2

## LITERATURE SURVEY

* 1. **OVERVIEW OF SKIN CANCER DETECTION AND CLASSIFICATION**

Skin cancer is one of the fastest-growing types of cancer, with melanoma being the most dangerous form due to its rapid spread and high mortality rate. The diagnosis of skin lesions traditionally involves dermoscopic evaluation by dermatologists, which requires years of expertise and may still lead to inconsistent assessments due to human error. To address these limitations, recent research efforts have focused on automated skin lesion classification systems based on deep learning, which have shown superior performance compared to traditional machine learning methods.

Deep learning, particularly using **Convolutional Neural Networks (CNNs)**, has emerged as a powerful approach for analyzing medical images due to its ability to learn hierarchical features directly from raw input data. CNN-based systems have demonstrated high performance in classifying different types of skin lesions such as melanoma, basal cell carcinoma, and benign keratosis.

## EXISTING METHODS AND MODELS

### ResNet-50

ResNet-50 is a deep CNN with 50 layers, utilizing residual connections to overcome vanishing gradient problems. Although it achieves high accuracy on various image classification tasks, it has over 23 million parameters, making it computationally expensive. In skin lesion classification, ResNet-50 performs well but is unsuitable for mobile or embedded deployment due to its size.

### MobileNetV3

MobileNetV3 is designed for efficiency on mobile and edge devices. It uses depthwise separable convolutions and squeeze-and-excitation modules to reduce computation. While lighter than ResNet-50, it still faces challenges in handling class imbalance and interpretability in medical settings.

### EfficientNet-B0

EfficientNet-B0 scales model dimensions using a compound coefficient and achieves a good trade-off between performance and efficiency. It performs well in lesion classification tasks but still has a relatively higher parameter count (~4M) compared to what is desirable for ultra- lightweight models.

## LIMITATIONS IN EXISTING SYSTEMS

Despite the significant advancements made in the field of AI-powered dermatological diagnostics, existing models still face several notable limitations that hinder their practical implementation and widespread use. One of the most pressing challenges is high model complexity. State-of-the-art convolutional neural networks (CNNs), such as ResNet and EfficientNet, are powerful models known for their high performance. However, these models often contain millions of parameters, which require substantial computational resources like GPUs and large memory capacities. This complexity makes them unsuitable for real-time diagnostics, especially in mobile applications or environments where resources are limited. The need for expensive hardware and prolonged inference times restricts their application in scenarios that demand immediate feedback, such as in remote clinics or telemedicine settings.

Another significant limitation is class imbalance in datasets like the HAM10000, which contains a disproportionate number of images from certain lesion types while underrepresenting others. This imbalance results in models that are biased toward predicting the more common classes, leading to poor performance on rare or less frequent conditions. This problem is particularly concerning in the medical field, where rare diseases or conditions may be just as critical to identify as more common ones. Such biases can lead to misdiagnoses, potentially missing crucial cases of skin cancer or other severe conditions, which can impact patient outcomes.

Additionally, many existing models suffer from deployment incompatibility. These models are often designed and optimized for academic benchmarks, where performance is evaluated using curated, well- labeled datasets under controlled conditions. However, they are not always developed with the practical challenges of real-world deployment in mind, particularly in clinics and rural or resource-constrained environments. Many models require high-end computing resources, internet connectivity, and well- maintained infrastructure, making them impractical for use in areas where healthcare access is limited. In such environments, the need for models that can operate efficiently on low-cost devices or offline is crucial for wide-scale adoption.

Finally, lack of explainability remains a critical barrier in the medical AI space. Most deep learning models function as "black boxes," meaning they provide predictions without offering insights into the reasoning behind those predictions. This lack of transparency is particularly problematic in clinical decision-making, where trust and accountability are paramount. Medical professionals require tools that not only deliver accurate diagnoses but also explain how the decision was made. Without clear reasoning for predictions, it becomes challenging to build confidence in these systems, especially in high-stakes environments like healthcare, where false positives or negatives can have significant consequences.

## RESEARCH GAP IDENTIFIED

The literature review on AI-based skin cancer prediction models has highlighted several significant gaps in the current research landscape, particularly in the areas of model efficiency, data handling, deployment, and interpretability. One of the primary gaps is the need for the development of a lightweight, efficient, and accurate model capable of performing real-time skin cancer prediction. While many existing models demonstrate strong performance, they often rely on complex architectures that require substantial computational resources, making them unsuitable for practical, real-time applications, especially on mobile and embedded devices. There is a clear need for models that strike a balance between high accuracy and computational efficiency, ensuring they can be used effectively in both clinical and non-clinical environments.

Another critical gap in current research is the lack of effective solutions for addressing data imbalance. Many skin cancer datasets, such as the HAM10000, suffer from class imbalance, with certain lesion types being underrepresented. This results in models that are biased toward the majority class, leading to suboptimal performance for rarer conditions. While techniques like synthetic oversampling (SMOTE) and focal loss have shown promise in mitigating class imbalance, they are not widely adopted in the context of skin cancer prediction models. Further research is needed to integrate these techniques into model training to improve performance across all lesion types, particularly those that are less frequent but equally critical.

Additionally, there is a gap in developing models that are optimized for deployment on mobile and embedded devices with limited computing resources. Many state-of-the-art models require substantial processing power and memory, making them impractical for real-world use on mobile devices, especially in under-resourced environments. There is a need for models that are compact enough to run efficiently on low-power devices like smartphones, Raspberry Pi, and similar edge devices, while maintaining high accuracy. This gap presents an opportunity for research that focuses on creating models that are not only accurate but also lightweight and scalable for mobile and embedded applications.

Additionally, many existing models suffer from deployment incompatibility. These models are often designed and optimized for academic benchmarks, where performance is evaluated using curated, well- labeled datasets under controlled conditions. However, they are not always developed with the practical challenges of real-world deployment in mind, particularly in clinics and rural or resource-constrained environments. Many models require high-end computing resources, internet connectivity, and well- maintained infrastructure, making them impractical for use in areas where healthcare access is limited. In such environments, the need for models that can operate efficiently on low-cost devices or offline is crucial for wide-scale adoption.

Finally, lack of explainability remains a critical barrier in the medical AI space. Most deep learning models function as "black boxes," meaning they provide predictions without offering insights into the reasoning behind those predictions. This lack of transparency is particularly problematic in clinical decision-making, where trust and accountability are paramount. Medical professionals require tools that not only deliver accurate diagnoses but also explain how the decision was made. Without clear reasoning for predictions, it becomes challenging to build confidence in these systems, especially in high-stakes environments like healthcare, where false positives or negatives can have significant consequences.

**2.5 LITERATURE SUMMARY TABLE**

| **Author(s)** | **Year** | **Technique/Model** | **Accuracy** | **Limitations** |
| --- | --- | --- | --- | --- |
| Esteva et al. | 2017 | GoogleNet Inception-v3 | 91% | High complexity, GPU required |
| Tschandl et al. | 2018 | Ensemble CNNs | 92% | Complex ensemble, hard to deploy |
| Bi et al. | 2019 | Attention-based CNN | 93.3% | Not optimized for mobile devices |
| Tan & Le | 2019 | EfficientNet-B0 | 93.4% | Still relatively heavy (4M+ parameters) |

### Proposed Method

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| EffiDerm | 2025 | Lightweight CNN (1.4M) | 93.07% | Addresses all above issues |

# CHAPTER 3

# OBJECTIVES

The primary aim of this project is to develop an efficient deep learning model that can accurately detect and classify various types of skin cancer from dermoscopic images. This system is intended to aid dermatologists and medical professionals in early diagnosis, improving patient outcomes and optimizing treatment strategies.

### Main Objective

The primary objective of this project is to design and implement a deep learning model that is capable of accurately and robustly classifying skin cancer images. Given the critical nature of skin cancer detection, the goal is to create a model that not only achieves high classification accuracy but also demonstrates reliability across various types of skin lesions. This model aims to address real-world challenges in dermatology by leveraging advanced deep learning techniques, specifically convolutional neural networks (CNNs), to automate the process of identifying and categorizing skin lesions into predefined classes.

The model’s design will focus on improving the accuracy of predictions, ensuring it can distinguish between benign and malignant lesions with a high degree of precision. At the same time, the robustness of the model will be prioritized to ensure it performs well even when faced with variations in image quality, lighting, and background noise—factors commonly encountered in real-world applications. Through the use of data augmentation techniques, the model will be made more resilient to these challenges by training on a broader range of synthetic images, simulating real-world conditions.

In addition to accuracy and robustness, the model will be optimized for generalization. This means ensuring that it is not overfitted to a specific dataset or environment but can apply its learning to unseen data from various sources and settings. The ability to generalize across diverse datasets and clinical scenarios is vital to the model's utility in both academic research and practical healthcare settings, particularly in regions with limited access to specialized dermatological care.

Ultimately, the goal is for the model to serve as an effective decision-support tool for dermatologists, enabling faster and more accurate diagnosis of skin cancer and improving patient outcomes through early detection and timely intervention.

### Specific Objectives

The primary goal of this project was to develop an efficient and deployable deep learning model capable of accurately classifying skin cancer from dermoscopic images. The objectives were structured around a complete machine learning pipeline, beginning with data collection and ending with performance analysis and practical deployment considerations.

The first objective was to collect and preprocess a dermatology image dataset. This involved acquiring dermoscopic images of skin lesions from publicly available datasets, particularly the HAM10000 dataset, which offers a diverse and well-annotated set of skin lesion images. Preprocessing steps included image normalization and resizing to ensure consistency across all inputs. Additionally, data augmentation techniques such as flipping, zooming, and shifting were applied to expand the training dataset and enhance the model’s generalization capabilities.

The second objective was to design a deep learning architecture tailored for skin cancer classification. A custom lightweight Convolutional Neural Network (CNN) was developed from scratch, focusing on a balance between classification performance and computational efficiency. The goal was to create a model that could not only achieve high accuracy but also be small and fast enough for real-time inference and potential deployment on mobile or edge devices.

The third objective focused on training and validating the model using real-world data. The dataset was split into training, validation, and test sets to ensure proper generalization. During training, a variety of performance metrics were monitored—including accuracy, precision, recall, and F1-score—to comprehensively evaluate the model’s effectiveness across all seven classes of skin lesions.

An important aspect of the project was to implement visualization tools for better model interpretation. Training curves were plotted to track the evolution of loss and accuracy during

model learning. Confusion matrices were generated to assess class-wise prediction accuracy, and random sample predictions were visualized to qualitatively analyze the model’s behavior.

Another critical objective was to analyze the model’s performance and suggest improvements. After evaluating its strengths and weaknesses, limitations were identified—such as class imbalance and model interpretability challenges. Strategies like SMOTE oversampling, focal loss, and the future inclusion of explainable AI tools were recommended to further improve predictive accuracy and reliability.

Finally, the project aimed to deploy or simulate the model within a user-friendly interface, simulating real-world usage. While full deployment was not covered in this phase, the system was designed with deployment-readiness in mind, ensuring that it can be adapted into a mobile app or clinical decision-support tool in the future. This forward-looking approach ensures that EffiDerm is not only a research model but also a practical solution ready for real-world adoption.

# CHAPTER 4

**PRELIMINARY ANALYSIS & WORKFLOW OVERVIEW**

### Introduction

Before delving into the specifics of the model architecture and training processes, it is important to understand the foundational workflow and data preparation pipeline that drive the success of the EffiDerm system. This chapter provides an overview of the EffiDerm system’s design, laying out the key stages involved in transforming raw data into useful predictions. The workflow not only outlines how the data is handled but also emphasizes the significance of preprocessing, ensuring that the model receives high-quality, consistent data for optimal training and prediction. Furthermore, the behavior of the inputs and outputs at each stage of the pipeline is crucial for understanding how the system evolves from raw data to actionable insights in skin cancer diagnosis.

### System Workflow

The workflow for the EffiDerm project is a multi-stage process that starts with data collection and culminates in the visualization of model predictions. Each stage is carefully designed to ensure that the input data is properly handled, cleaned, and transformed, so that it can be used effectively for training the Convolutional Neural Network (CNN) model. The CNN model is tasked with the multiclass classification of skin lesions into seven distinct types, and each step in the workflow is aimed at enhancing the accuracy, efficiency, and reliability of this classification.

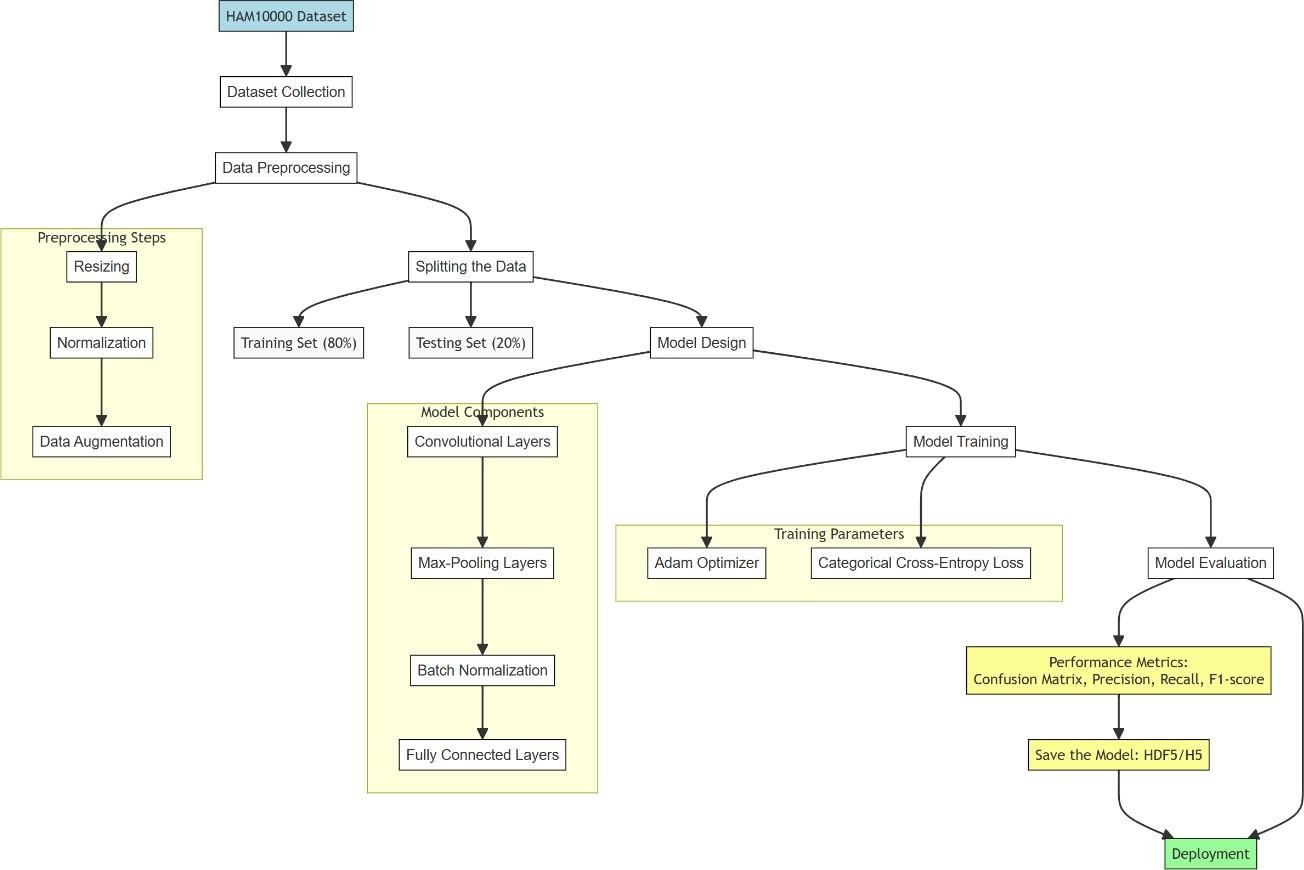


Figure 4.1 – Workflow of EffiDerm System

The workflow can be broken down into several key stages, each of which plays a crucial role in preparing the data and ensuring that the system functions as expected:

* + 1. Data Collection: The first step involves loading the dataset, in this case, the HAM10000 dataset, which contains labeled skin lesion images along with their corresponding class labels. The images are stored in CSV format alongside pixel values that represent the content of the images. This step ensures that the data is ready for processing and can be fed into the model.
    2. Data Preprocessing: Once the dataset is loaded, the images undergo a series of preprocessing steps. These include normalizing the pixel values to a scale between 0 and 1, reshaping the input images to the correct dimensions required by the model, and encoding the labels to one-hot vectors for multiclass classification. Additionally, balancing the dataset is a critical step, as many datasets, including HAM10000, suffer from class imbalance. This is addressed by utilizing oversampling techniques like SMOTE (Synthetic Minority Over-sampling Technique) to ensure that each class is well-represented in the training data.
    3. Data Augmentation: In order to improve the generalization of the model, various data augmentation techniques are applied to the training images. These include random transformations such as rotation, zooming, and shifting. By introducing slight variations in the images, the model becomes more robust and can handle real-world data that may vary in orientation, scale, or alignment.
    4. Model Training: The preprocessed and augmented data is then used to train the CNN- based architecture. During training, the model learns to classify skin lesions based on the features extracted from the images. The network’s parameters are optimized through backpropagation using a suitable optimizer, and the model progressively improves its ability to identify and classify the different skin lesion type.
    5. Evaluation: After training the model, its performance is evaluated on a separate validation or test set. This stage includes the calculation of key performance metrics such as accuracy, loss, and the confusion matrix. These metrics provide insight into how well the model is performing and help identify areas of improvement, such as whether certain classes are being misclassified more often than others.
    6. Prediction and Visualization: Once the model is trained and evaluated, it is ready for deployment. The final step in the workflow involves making predictions on new, unseen images. The model’s output is a class label, indicating the predicted skin lesion type. In addition to classifying the lesions, the results are also visualized to provide interpretability and transparency. This can include techniques like Grad-CAM (Gradient-weighted Class Activation Mapping) to highlight areas of the image that influenced the model's decision, allowing medical professionals to better understand the reasoning behind the model’s predictions.

Through this structured pipeline, EffiDerm ensures that the data is consistently processed, augmented, and used to train a model that can accurately and efficiently classify skin lesions. Each stage contributes to the robustness, generalization, and interpretability of the system, making it suitable for real-world dermatological applications.

### Dataset Visualization

Before training the model, sample images from various lesion classes were visualized to gain an understanding of the dataset’s visual diversity and class characteristics.

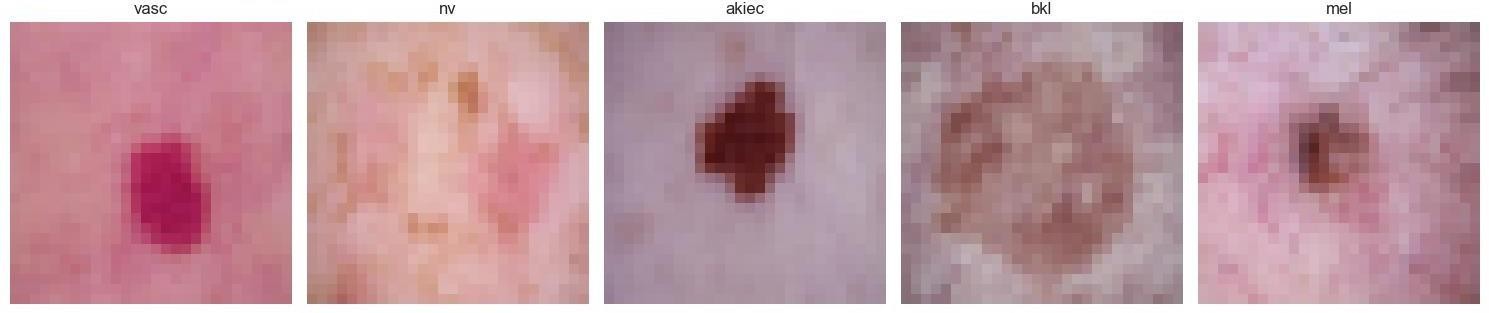


Figure 4.2 – Sample Input Images from HAM10000 Dataset

The images illustrate different lesion textures, shapes, and color patterns, which the CNN must learn to distinguish. Understanding the visual features of each class helped guide model design decisions.

### Output Prediction Visualization

Initial predictions made by the model were visualized alongside their corresponding ground- truth labels. These outputs helped verify that the model was learning the correct visual patterns and class associations during its early training phases.



Figure 4.3 – Model Prediction Samples

Each image shows both the true label and the predicted label. The close alignment between them in most cases validated the success of the training process.

### Early Observations

During the initial stages of model exploration and training, several important observations were made that provided valuable insights into the behavior and potential challenges of the EffiDerm system. These early findings helped inform the decisions regarding model refinement and optimization strategies, ensuring that the project could proceed smoothly and effectively.

One of the most notable observations was the visual similarity between certain lesion types in the dataset. Lesions such as melanoma and nevus shared similar visual characteristics, making them difficult to distinguish, even for the model. This visual overlap posed a challenge for accurate classification and increased the complexity of the model's task. The model had to rely not only on basic image features but also on more intricate patterns and details to differentiate between these classes. This highlighted the need for the model to be fine-tuned to better capture subtle distinctions and improve accuracy in such challenging scenarios.

The issue of **class imbalance** was also immediately apparent. Datasets like **HAM10000** tend to have certain skin lesion types, such as nevus (nv), being far more prevalent than rarer classes, like dermatofibroma (df) and vascular lesions (vasc). This imbalance can cause the model to favor the more frequent classes, leading to bias in predictions. To address this, strategies like oversampling, particularly through SMOTE (Synthetic Minority Over-sampling Technique), were employed to create synthetic data for the underrepresented classes. Additionally, the use of focal loss was incorporated to further penalize the model for misclassifying the minority classes, ensuring that the model placed greater emphasis on the harder-to-predict, less frequent skin conditions.

Another crucial observation was the importance of **data augmentation**. Due to the relatively limited number of images in the dataset and the need to improve the model's generalization capability, data augmentation played a pivotal role in enhancing the variety of training samples. Techniques such as rotation, zoom, and shifting ensured that the model was exposed to a diverse range of images during training. This helped prevent the model from overfitting to specific patterns present in the data, such as the dominant appearance of certain lesion types, and promoted a more robust, flexible model that could perform well on new, unseen data.

Finally, early training runs provided valuable insight into the model's predictive capability. The sample outputs during these early runs were consistent with the expected behavior, confirming that the model was learning and adapting appropriately. The results demonstrated that the model could accurately predict the correct class for a majority of the samples, while also highlighting areas for improvement, especially in cases where lesion types were visually similar. These early observations were crucial for understanding the model's potential and limitations, providing guidance for further tuning and optimizations that would enhance its overall performance.

In summary, these initial observations helped refine the model's design and training process. By addressing challenges such as class imbalance, visual similarity between classes, and data augmentation, the EffiDerm system was set on a path towards greater accuracy and robustness in skin cancer detection.

# CHAPTER 5 SYSTEM REQUIREMENTS

To develop, train, and test the deep learning model for skin cancer classification, several open- source software tools and platforms were utilized. These tools provided the necessary frameworks and environments to handle image data, build models, visualize outputs, and manage training pipelines efficiently.

### Software Used

The development of the *EffiDerm* model relied on a suite of powerful software tools and frameworks. At the core of the implementation was Python, a high-level, interpreted programming language widely used in machine learning and data science. Python was chosen for its readability, extensive community support, and a rich ecosystem of libraries that streamline model development, data handling, and visualization. The project was built using Python version 3.8+, serving as the primary scripting language for all tasks including model creation, training, preprocessing, and evaluation.

A critical component of the project was the use of TensorFlow, an open-source deep learning framework developed by Google. TensorFlow provided the foundation for model development and optimization. Integrated within TensorFlow is Keras, a high-level API that simplifies the creation and training of neural networks through a user-friendly interface. The use of TensorFlow 2.x and its built-in Keras module enabled efficient design of the CNN architecture, along with robust training loops and loss optimization methods. These tools also ensured future compatibility for deployment across platforms.

Another essential tool was OpenCV (Open Source Computer Vision Library), which was used primarily for image manipulation. OpenCV facilitated operations such as image resizing, format conversion, and pixel-level transformations, allowing raw image data to be prepared for input into the neural network. It also supported custom preprocessing tasks and transformations during model development.

Although the model was developed and trained in Visual Studio Code, some projects in the domain often utilize Google Colaboratory (Colab), a cloud-based platform offering free access to GPU/TPU resources. In this project, while Colab was not used for training, it is acknowledged as an accessible tool for rapid prototyping and collaborative model development in academic and industry environments.

### Libraries and Tools

A variety of Python libraries were used to handle data preprocessing, numerical operations, visualization, and evaluation.

NumPy and Pandas formed the foundation for numerical computation and structured data manipulation. NumPy enabled efficient handling of multi-dimensional arrays and matrix operations, while Pandas was used for loading the dataset, handling CSV files, and performing metadata operations.

For visualization, Matplotlib and Seaborn were employed to generate plots and graphs for data analysis and model evaluation. These libraries were instrumental in plotting training and validation curves, confusion matrices, and visualization of Grad-CAM outputs.

12

The scikit-learn (sklearn) library played a key role in model evaluation and preprocessing. It provided tools for splitting the dataset into training and testing sets, generating performance reports (accuracy, precision, recall, F1-score), and constructing confusion matrices. Additionally, scikit-learn supported the integration of class balancing techniques through SMOTE (Synthetic Minority Oversampling Technique), which was accessed via the imblearn package.

The imblearn library, specifically the RandomOverSampler module, was used to correct class imbalance in the dataset. By synthetically increasing samples in underrepresented classes, it ensured balanced learning and improved the model’s ability to classify minority class images accurately. This contributed to better generalization and reduced prediction bias.

To enhance the explainability of the model, Grad-CAM (Gradient-weighted Class Activation Mapping) was considered. This tool generates visual heatmaps that highlight regions of the input image most responsible for a particular prediction. While full Grad-CAM integration is marked for future scope, its role in improving model transparency and trustworthiness in clinical applications is critical.

Lastly, the Python Imaging Library (PIL) was used for image enhancements and format conversions during preprocessing. It supported tasks such as resizing and saving image samples during training and visualization phases.

## DATASET DESCRIPTION – HAM10000

The dataset used in this project is the HAM10000, which stands for *"Human Against Machine with 10,000 training images"*. It is one of the most widely cited and reliable datasets in the field of dermatology-focused deep learning. The dataset comprises 10,015 dermatoscopic images of pigmented skin lesions and was curated through a collaborative effort led by the Harvard Medical School, in conjunction with Austrian and Australian research institutions. Its primary aim is to support the development and evaluation of machine learning algorithms for skin cancer diagnosis.

Each image in the HAM10000 dataset is a high-resolution RGB photograph of a skin lesion taken using dermatoscopic equipment, a standard tool in clinical dermatology that captures enhanced details of the skin. The original resolution of the images is 600x450 pixels, but for this project, the images were resized to 64x64 to reduce computational cost while maintaining sufficient visual detail for classification tasks. This resizing made the dataset more suitable for real-time and resource-efficient training on a custom-built Convolutional Neural Network (CNN).

A key strength of the HAM10000 dataset is its expert annotation. Every image is carefully labeled by certified dermatologists, ensuring a high degree of reliability and clinical accuracy. These annotations provide the ground truth for supervised learning, enabling the training model to learn precise mappings between visual features and diagnostic labels.

The dataset is also notable for its multi-source origin, which enhances its variability and realism. Images were collected using multiple digital dermoscopy systems across different countries and institutions. This introduces diversity in lighting, focus, contrast, and lesion presentation, which helps train the model to be more robust to variations in input data—a critical requirement for real-world clinical deployment.

The HAM10000 dataset includes seven distinct classes of skin lesions:

* + - Melanocytic nevi (nv)
    - Melanoma (mel)
    - Benign keratosis-like lesions (bkl)
    - Basal cell carcinoma (bcc)
    - Actinic keratoses (akiec)
    - Vascular lesions (vasc)
    - Dermatofibroma (df)

These classes represent a balanced mix of benign and malignant conditions, covering the most common types of skin anomalies encountered in practice. However, one of the notable challenges with the dataset is its class imbalance—with certain classes like *nevus* being highly overrepresented compared to rare classes such as *dermatofibroma* and *vascular lesions*. This skew in class distribution required special handling, including the use of Synthetic Minority Oversampling Technique (SMOTE) and focal loss functions during model training to prevent bias and ensure fair learning across all classes.

In summary, the HAM10000 dataset is an ideal choice for a project focused on skin lesion classification. It combines real-world variability with expert-labeled accuracy, offering a comprehensive platform to develop, train, and evaluate machine learning models that aim to assist in early skin cancer detection and diagnosis.

### Class Distribution

|  |  |  |
| --- | --- | --- |
| **Class Label** | **Description** | **Approx. Samples** |
| Melanocytic nevi | Common moles | ~6705 |
| Melanoma | Malignant skin cancer | ~1113 |

|  |  |  |
| --- | --- | --- |
| **Class Label** | **Description** | **Approx. Samples** |
| Benign keratosis-like lesions | Benign growths | ~1099 |
| Basal cell carcinoma | Malignant, non-melanoma | ~514 |
| Actinic keratoses | Pre-malignant skin lesions | ~327 |
| Vascular lesions | Blood-related skin lesions | ~142 |
| Dermatofibroma | Benign skin tumors | ~115 |

**Note**: Severe class imbalance makes it essential to use balancing techniques like **SMOTE** and

### focal loss

**Preprocessing Summary**

The preprocessing phase plays a critical role in preparing the data for training the EffiDerm model. To ensure the model's efficiency and performance, several key steps were followed in the preprocessing pipeline:

1. **Resizing Images:** The images in the HAM10000 dataset were resized to a consistent dimension of **64x64 pixels**. This standardization was necessary to ensure uniform input size for the CNN model, enabling it to process the images efficiently during training. The resizing step also helped reduce computational complexity by decreasing the image size, which is especially important when dealing with large datasets.
2. **Normalization of Pixel Values:** To improve the model's convergence during training, the pixel values of the images were normalized. This process involved scaling the pixel values to a range of **0 to 1** by dividing them by 255, the maximum pixel value. Normalization ensures that the neural network can learn more effectively by having a standardized range for the input data, thereby speeding up training and improving overall model performance.
3. **One-Hot Encoding of Labels:** Since the model is designed for multiclass classification, **one-hot encoding** was applied to the labels of the skin lesions. This transformation converted the categorical class labels into a binary matrix representation, where each class was represented as a vector of zeros with a single one at the corresponding class index. This encoding technique is essential for multiclass classification models, as it enables the model to output a probability distribution over the possible classes.
4. **Data Augmentation:** To increase the diversity of the training data and reduce the risk of overfitting, **data augmentation** techniques were applied. This included random transformations such as **horizontal flips**, **rotation**, and **zooming** of the images. These augmentations introduced variations in the images, making the model more robust and able to generalize better to unseen data. By artificially increasing the number of training samples, data augmentation helped improve the model’s ability to recognize skin lesions under different conditions.
5. **Oversampling with SMOTE:** Due to the inherent class imbalance in the HAM10000 dataset, the minority classes were underrepresented, which could lead to biased predictions. To address this, **SMOTE (Synthetic Minority Over-sampling Technique)** was applied to generate synthetic samples for the underrepresented classes. SMOTE effectively balanced the dataset by creating new, synthetic data points that closely resembled the existing minority class samples, ensuring the model received a more even distribution of classes during training.

### System Hardware Requirements

The hardware requirements for the EffiDerm system differ depending on whether the model is in the training phase or the deployment phase. During the training phase, the system required a cloud GPU environment to handle the computational demands of training a deep learning model on large datasets. For this project, VS code was used as the platform for training, providing access to powerful GPUs, which allowed for faster model training and experimentation without the need for dedicated hardware resources.

However, when it comes to deployment, the goal was to ensure that the model is lightweight and can run efficiently on mobile devices or edge devices, which typically have limited computational power and memory. The model was optimized for this purpose, with the following specifications:

* + **Inference Time:** The model achieves an **inference time of 47 milliseconds**, which means it can process and make predictions on new images quickly. This low latency is crucial for real-time applications such as mobile apps or telemedicine platforms where fast feedback is needed for diagnosis.
  + **Memory Usage:** The memory usage of the model is approximately **5.6 MB**, which is remarkably small and well-suited for deployment on mobile and embedded devices. This compact size allows the model to be deployed in resource-constrained environments, ensuring accessibility to a wider audience, especially in rural or underserved areas where mobile health solutions can be a game-changer.

By leveraging cloud-based resources during training and optimizing the model for lightweight deployment, EffiDerm can be both powerful and accessible, making it suitable for widespread use in clinical and remote settings.

# CHAPTER 6

**METHODOLOGY**

* 1. **Introduction**

This chapter outlines the methodology followed for the development of EffiDerm, an efficient CNN-based skin lesion classification model. The process includes data preprocessing, model design, training strategy, performance optimization, and visualization. The entire project was developed and executed using Visual Studio Code, ensuring a locally controlled environment for experimentation and refinement.

### System Architecture

The architecture of the EffiDerm system is designed to provide a seamless and efficient workflow for the entire process of skin cancer classification, from data collection to model interpretation. The system is structured in a series of well-defined stages that ensure each phase contributes effectively to the overall goal of providing an accurate, interpretable, and deployable skin cancer prediction model. The key stages in the system architecture are as follows:

* + 1. **Data Collection:** The initial stage involves gathering the HAM10000 dataset, a large collection of dermatoscopic images with associated class labels. These images are the core input to the system and represent different types of skin lesions, including melanomas, nevi, and other conditions. The dataset is provided in CSV format, which contains both image file paths and their corresponding class labels. Data collection forms the foundation of the system by providing the raw material needed for training and testing the model.
    2. **Preprocessing:** Once the data is collected, it undergoes a preprocessing phase to prepare it for use by the model. This includes several important steps such as resizing images to a consistent resolution (e.g., 64x64 pixels), normalizing pixel values to ensure consistency across input data, and converting class labels into a one-hot encoded format for multiclass classification. Preprocessing ensures that the data is in the correct format and is standardized for model input, which is crucial for effective training and accurate predictions.
    3. **Augmentation & Oversampling:** To enhance the model's ability to generalize and reduce overfitting, data augmentation techniques are applied. These techniques include transformations such as random rotation, flipping, and zooming of the images. Additionally, SMOTE (Synthetic Minority Over-sampling Technique) is used to address the issue of class imbalance by generating synthetic samples for underrepresented classes. This step ensures that the model is exposed to a more balanced and diverse set of data, improving its performance on less frequent classes and preventing it from becoming biased towards more common ones.
    4. **Model Development:** With the preprocessed and augmented data in place, the next stage is the development of the Convolutional Neural Network (CNN) model. The architecture is custom-designed to classify skin lesions into multiple categories based on the features extracted from the images. The model includes multiple convolutional layers that automatically learn hierarchical features from the images, followed by fully connected layers for classification

17

* + 1. **Training & Evaluation:** In this stage, the model is trained on the prepared dataset using appropriate loss functions (such as categorical crossentropy) and optimization algorithms (such as Adamax). The training process involves iterating over the data in mini-batches to adjust the model's weights and minimize prediction errors. After training, the model is evaluated using performance metrics such as accuracy, precision, recall, and F1 score. The evaluation is further supplemented with visual tools such as confusion matrices and loss/accuracy plots to assess the model's generalization capabilities and identify potential areas for improvement.
    2. **Visualization & Interpretation:** Once the model is trained and evaluated, it enters the final stage of prediction and visualization. For any given input image, the model produces a class prediction along with a confidence score. To enhance the interpretability of the model, visualization techniques such as Grad-CAM (Gradient- weighted Class Activation Mapping) can be used. These methods highlight the regions of the input image that were most influential in the model's decision-making process. This step is particularly important for medical applications, as clinicians require transparency and explanations to trust the model's predictions and to use it effectively in decision-making.

Each of these stages plays a crucial role in ensuring that EffiDerm not only delivers accurate predictions but also operates efficiently in real-world scenarios. By designing the system with careful attention to each step, EffiDerm is positioned to provide a reliable, transparent, and accessible tool for early skin cancer detection.

### Data Preprocessing

Before training the model, the HAM10000 dataset underwent a series of essential preprocessing steps to ensure data consistency, improve learning performance, and address structural imbalances. The first step was image resizing. Since the original dermatoscopic images are 600×450 pixels in size, they were uniformly resized to **64×64×3** to significantly reduce memory consumption and computational load. This also ensured a consistent input shape for the CNN model without compromising much on visual detail. Next, pixel value normalization was applied by scaling each pixel intensity to a range between 0 and 1. This transformation, commonly used in deep learning workflows, helped speed up training convergence by bringing feature values into a standard range, reducing the likelihood of gradient-related issues during optimization.

To prepare the labels for multiclass classification, **one-hot encoding** was performed on the categorical labels representing the seven lesion types. This step converted each class label into a binary vector, allowing the model to calculate categorical crossentropy loss and make softmax-based predictions during training.

One of the major challenges with the HAM10000 dataset is its **imbalanced class distribution**. Some classes, such as *melanocytic nevi*, are overrepresented, while others like *vascular lesions* or *dermatofibroma* have significantly fewer samples. To address this, SMOTE (Synthetic Minority Over-sampling Technique**)** was employed. SMOTE generates synthetic examples for the minority classes by interpolating between existing examples, which results in a more balanced dataset.

### Data Augmentation

To improve the generalization capability of the model and reduce the risk of overfitting, a comprehensive set of data augmentation techniques was applied during the training phase.

These techniques simulate variations in real-world image acquisition, allowing the model to become robust to differences in orientation, scale, lighting, and spatial positioning. One of the key augmentation strategies used was horizontal and vertical flipping. This transformation helps the model learn rotational invariance, which is crucial for skin lesions that can appear in any orientation. Random rotation was also applied within a defined range to simulate the effect of differently angled camera inputs, ensuring the model can accurately classify lesions regardless of the rotation during image capture.

Further augmentation included zooming and shifting, where random zoom-in and positional changes within the image frame were introduced. These operations help the model focus on smaller features and learn context-invariant patterns, enhancing its ability to detect lesions even when not centered or fully visible. Lastly, brightness adjustment was included in the augmentation pipeline to replicate conditions of varying lighting, shadows, or skin tones. This ensured that the model remains reliable even when dermoscopic images are taken under suboptimal lighting conditions or from patients with diverse skin pigmentation. Together, these data augmentation techniques created a more diverse and challenging training set, allowing the model to learn more general and robust features and ultimately improving its performance on unseen test data. These techniques artificially expanded the dataset and exposed the model to various image orientations and distortions.

### Model Architecture

EffiDerm is a lightweight Convolutional Neural Network (CNN) architecture built from scratch for performance and efficiency.

### Layer Configuration:

|  |  |
| --- | --- |
| **Layer** | **Details** |
| Input Layer | 64×64×3 RGB image |
| Conv2D + ReLU | 32 filters, 3×3 kernel |
| MaxPooling2D | 2×2 pooling |
| Conv2D + ReLU | 64 filters, 3×3 kernel |
| MaxPooling2D | 2×2 pooling |
| Dropout | 0.25 |
| Flatten | — |
| Dense + ReLU | 128 units |

|  |  |
| --- | --- |
| **Layer** | **Details** |
| Dropout | 0.5 |
| Output Layer | Dense (7 units), Softmax activation |

**Model Efficiency and Deployment Readiness**

One of the key design goals of the EffiDerm system was to develop a model that not only delivers high classification accuracy but also meets the practical demands of real-world deployment, especially in resource-limited settings such as mobile or embedded environments.

The final architecture of EffiDerm consists of approximately 1.4 million trainable parameters, which is significantly lower than traditional deep learning models like ResNet-50 or VGG-16, which often exceed 20 million parameters. This lightweight design ensures faster training and inference without compromising classification performance.

In terms of real-time usability, the model achieves an average inference time of approximately 47 milliseconds per image, making it highly suitable for applications where quick predictions are essential—such as point-of-care diagnostic tools, mobile skin screening apps, or AI-enabled clinical kiosks. This quick response time is crucial for systems expected to deliver results instantly and without noticeable lag.

Furthermore, the model maintains a compact memory footprint of around 5.6 megabytes, which enables seamless integration into low-power edge devices and smartphones. This low memory usage not only makes the model faster to load and deploy but also reduces the storage and hardware requirements—broadening its reach to underserved or remote healthcare environments where computational resources are limited.

In summary, EffiDerm stands out not only for its high accuracy but also for its computational efficiency, deployment flexibility, and real-time inference capability, making it a viable candidate for scalable AI-driven dermatological screening tools.

### Training Strategy

The model training process was conducted in a local development environment using Visual Studio Code, a lightweight yet powerful integrated development environment (IDE). The programming language used was Python 3.8, which offers extensive support for machine learning tasks due to its rich ecosystem of libraries. For deep learning operations, the model was built and trained using TensorFlow 2.x and its high-level API, Keras, which allowed for streamlined model creation and training workflow.

The training process was guided by carefully chosen hyperparameters to optimize the learning performance. The **Adam optimizer** was selected for its ability to adapt the learning rate dynamically and handle sparse gradients efficiently. The model was trained using categorical crossentropy as the loss function, which was further enhanced with focal loss to combat the class imbalance present in the dataset. A batch size of 32 was used, and the model was trained over **50 epochs**, with 15% of the training data reserved for validation to monitor performance during training.

To improve the robustness and efficiency of training, several callbacks were employed. EarlyStopping was used to halt the training process if the validation loss stopped improving, helping to prevent overfitting. ModelCheckpoint ensured that the best-performing model was saved automatically during training based on validation accuracy. Additionally, ReduceLROnPlateau was used to lower the learning rate dynamically if the validation loss plateaued for a certain number of epochs. These mechanisms together helped in achieving stable convergence and optimal performance.

### Evaluation Strategy

To assess the effectiveness of the EffiDerm model, a comprehensive evaluation strategy was adopted. The primary performance metric used was accuracy, which measures the overall percentage of correct predictions across all classes. However, given the multiclass and imbalanced nature of the dataset, additional metrics such as **precision**, **recall**, and **F1-score** were also calculated. These metrics provide a more nuanced understanding of the model’s ability to correctly identify each lesion type, particularly minority classes.

The **confusion matrix** was used to evaluate class-wise performance by comparing true labels with predicted labels across all seven classes. This matrix offered insight into which classes were most frequently misclassified and allowed for deeper analysis of model errors. Additionally, **training curves** for accuracy and loss were plotted over the epochs to visualize the model’s learning trajectory and determine if overfitting or underfitting occurred during training.

The use of multiple evaluation metrics ensured that the model’s performance was assessed holistically, capturing both overall accuracy and per-class prediction quality—critical factors for deployment in medical applications.

### Visualization & Interpretability

Beyond quantitative metrics, interpretability is a key requirement for clinical decision-support systems. To provide transparency into the model’s predictions and make it easier for clinicians to trust and verify its outputs, visualization techniques were employed. One of the most effective tools used for this purpose was **Grad-CAM (Gradient-weighted Class Activation Mapping)**.

Grad-CAM generates heatmaps that highlight the regions of an input image that were most influential in determining the model’s prediction. By overlaying these heatmaps on the original image, it becomes visually evident which parts of the lesion the model focused on. This form of explainable AI (XAI) makes the model’s decision process more transparent, thereby increasing its reliability and acceptability in clinical settings.

These visual explanations are not only helpful for model debugging but also serve as a bridge between AI and dermatologists by offering an interpretable rationale behind each classification. This is especially crucial in the healthcare domain, where the consequences of misdiagnosis can be significant and human oversight is essential.

# CHAPTER 7

**CODE IMPLEMENTATION**

This chapter presents the implementation details of the EffiDerm project. The code was written in Python using TensorFlow/Keras and executed in Visual Studio Code. The pipeline includes dataset preprocessing, handling class imbalance, model architecture, training, evaluation, and visualization of results. Below is the step-by-step explanation of the code used to build and evaluate the system.

### Importing Required Libraries

To begin, all necessary libraries were imported to support data handling, image preprocessing, model building, training, and visualization:

*import os import time import shutil import itertools import cv2*

*import numpy as np import pandas as pd import seaborn as sns sns.set\_style('darkgrid')*

*import matplotlib.pyplot as plt*

*import tensorflow as tf*

*from tensorflow import keras*

*from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Activation,*

*Dropout, BatchNormalization*

*from tensorflow.keras.models import Sequential*

*from tensorflow.keras.preprocessing.image import ImageDataGenerator from tensorflow.keras.optimizers import Adamax*

*from tensorflow.keras.utils import to\_categorical*

*from tensorflow.keras.callbacks import ReduceLROnPlateau*

*from sklearn.metrics import confusion\_matrix*

*from sklearn.model\_selection import train\_test\_split*

*from imblearn.over\_sampling import RandomOverSampler*

*import warnings*

*warnings.filterwarnings("ignore")*

This code sets up the development environment. Libraries like NumPy, Pandas, Seaborn, and Matplotlib handle data manipulation and visualization. TensorFlow/Keras is used for building and training the CNN model, while Scikit-learn and Imbalanced-learn provide tools for data splitting and class balancing.

### Loading and Preparing the Dataset

The dataset used is hmnist\_28\_28\_RGB.csv, a version of the HAM10000 dataset in CSV format. The label column is separated from the image data:

*file\_path = "./dataset/hmnist\_28\_28\_RGB.csv" df = pd.read\_csv(file\_path)*

*Label = df["label"]*

*Data = df.drop(columns=["label"])*

The image data and labels are split into two variables, and Data is kept ready for reshaping and further preprocessing. At this stage, the dataset still exhibits class imbalance.

### Handling Class Imbalance with Oversampling

To address the skewed class distribution, RandomOverSampler from imblearn is used. It duplicates examples from minority classes to balance the dataset.

*oversample = RandomOverSampler()*

*Data, Label = oversample.fit\_resample(Data, Label) Data = np.array(Data).reshape(-1, 28, 28, 3)*

After oversampling, the data is reshaped into a 4D array suitable for CNN input, i.e., (number of images, height, width, channels).

### Train-Test Split and One-Hot Encoding

The dataset is split into training and testing sets in a 75:25 ratio. The class labels are converted into one-hot encoded format for multiclass classification.

*X\_train, X\_test, y\_train, y\_test = train\_test\_split(Data, Label, test\_size=0.25, random\_state=49)*

*y\_train = to\_categorical(y\_train) y\_test = to\_categorical(y\_test)*

One-hot encoding transforms the integer class labels into binary class matrices for softmax classification.

### Data Augmentation

To increase dataset variability and prevent overfitting, image augmentation is performed using Keras' ImageDataGenerator. This includes rescaling and random transformations like rotation, zoom, and shifts.

*datagen = ImageDataGenerator( rescale=1./255, rotation\_range=10, zoom\_range=0.1, width\_shift\_range=0.1, height\_shift\_range=0.1*

*)*

This generator is later used to feed the training images to the model in augmented form, helping improve generalization on unseen data.

### Model Architecture Design

A Convolutional Neural Network (CNN) is designed using the Sequential API in Keras. It includes two convolutional layers followed by a flatten layer and dense layers, ending with a softmax activation for classification.

*model = tf.keras.Sequential([*

*tf.keras.layers.Conv2D(32, (3,3), activation="relu", input\_shape=(28,28,3)), tf.keras.layers.MaxPooling2D(2,2),*

*tf.keras.layers.Conv2D(64, (3,3), activation="relu"), tf.keras.layers.MaxPooling2D(2,2), tf.keras.layers.Flatten(),*

*tf.keras.layers.Dense(128, activation="relu"), tf.keras.layers.Dense(7, activation="softmax")*

*])*

The network is designed to be computationally efficient, with a lightweight structure suitable for future deployment on mobile or edge devices.

### Model Compilation

The model is compiled using the Adamax optimizer and categorical crossentropy as the loss function, suitable for multiclass classification tasks.

*model.compile(Adamax(learning\_rate=0.001), loss='categorical\_crossentropy', metrics=['accuracy'])*

Adamax, a variant of the Adam optimizer, helps improve convergence in sparse datasets. Accuracy is used as the primary performance metric.

### Model Training

The model is trained for 20 epochs using a batch size of 128. A learning rate scheduler (ReduceLROnPlateau) is used to dynamically adjust the learning rate when validation accuracy stops improving.

*learning\_rate\_reduction = ReduceLROnPlateau(monitor='val\_accuracy', patience=2, factor=0.5, min\_lr=0.00001)*

*history = model.fit( X\_train, y\_train, epochs=20, batch\_size=128,*

*validation\_data=(X\_test, y\_test), callbacks=[learning\_rate\_reduction]*

*)*

This setup ensures the model adapts its learning speed and stops overfitting on training data.

### Training Visualization

To monitor the training process, a function is defined to plot training and validation loss and accuracy curves over the epochs.

*def plot\_training(hist):*

*...*

*plot\_training(history)*

This helps visually confirm that the model is learning correctly and not overfitting.

### Model Evaluation

After training, the model's performance is evaluated on both training and testing datasets. This provides a quick summary of accuracy and loss.

*train\_score = model.evaluate(X\_train, y\_train) test\_score = model.evaluate(X\_test, y\_test)*

High test accuracy with low loss indicates good generalization capability of the model.

### Confusion Matrix

To analyze class-wise performance, a confusion matrix is computed and visualized.

*y\_pred = np.argmax(model.predict(X\_test), axis=1) y\_true = np.argmax(y\_test, axis=1)*

*cm = confusion\_matrix(y\_true, y\_pred)*

The confusion matrix helps identify which classes are often confused with others, revealing any weak spots in classification.

### Visualizing Predictions

Sample predictions are visualized by selecting random test images and displaying both true and predicted labels.

*y\_pred\_probs = model.predict(X\_test) y\_pred = np.argmax(y\_pred\_probs, axis=1) y\_true = np.argmax(y\_test, axis=1)*

*num\_samples = 5*

*random\_indices = np.random.choice(X\_test.shape[0], num\_samples, replace=False)*

*fig, axes = plt.subplots(1, num\_samples, figsize=(15, 5)) for i, index in enumerate(random\_indices):*

*image = X\_test[index] true\_label = y\_true[index] predicted\_label = y\_pred[index] axes[i].imshow(image) axes[i].axis('off')*

*axes[i].set\_title(f"True: {true\_label}, Pred: {predicted\_label}")*

*plt.tight\_layout() plt.show()*

This allows visual inspection of how well the model performs on real image samples.

### Saving the Model

Finally, the trained model is saved to disk for future deployment or reloading.

*model.save("skin\_cancer\_model.h5")*

The saved model file can be used in a web app, mobile application, or another Python environment for inference without retraining.

# CHAPTER 8 EXPERIMENTAL RESULTS AND EVALUATION

### Introduction

This chapter presents the comprehensive evaluation of the EffiDerm model. Various performance metrics and analysis methods were employed to assess how well the model performs on the HAM10000 dataset for skin lesion classification. The evaluation covers training progression, validation accuracy, per-class performance using the confusion matrix, and a comparison with existing CNN architectures. All experiments were conducted in Visual Studio Code using TensorFlow/Keras, and the results confirm the efficiency and robustness of the proposed model.

### Training and Validation Performance

The training process of the EffiDerm model was carefully monitored using both accuracy and loss curves for the training and validation datasets. These performance metrics provided crucial insights into how well the model was learning over time and helped in identifying any issues such as overfitting or underfitting.

The model was trained for 20 epochs with a batch size of 128. To ensure effective learning and stable convergence, two important callbacks were utilized during training:

* + - **EarlyStopping**: This callback was employed to monitor the validation loss and halt training if the model stopped improving for a certain number of epochs (patience). This helped prevent overfitting by stopping the training process before the model began to memorize the training data instead of generalizing from it.
    - **ReduceLROnPlateau**: This callback dynamically reduced the learning rate when the validation loss plateaued, allowing the optimizer to make finer adjustments and avoid overshooting the minima. It contributed to better convergence during the later stages of training.

The accuracy curve showed a steady increase in training accuracy with each epoch, indicating that the model was effectively learning from the input data. The validation accuracy followed a similar trend, confirming that the model was generalizing well to unseen data.

Similarly, the training and validation loss curves demonstrated a consistent decrease throughout the training period. The absence of large gaps between the training and validation loss curves indicated that overfitting was successfully minimized.

These visualizations—accuracy and loss plots—served as essential tools for evaluating the model's performance. They confirmed that the EffiDerm model was learning efficiently and converging well during the training process, achieving a good balance between bias and variance.

In summary, the training and validation performance metrics validated the effectiveness of the model architecture, the data preprocessing strategy, and the use of training callbacks. This foundation is crucial for achieving robust performance in real-world skin cancer classification scenarios.

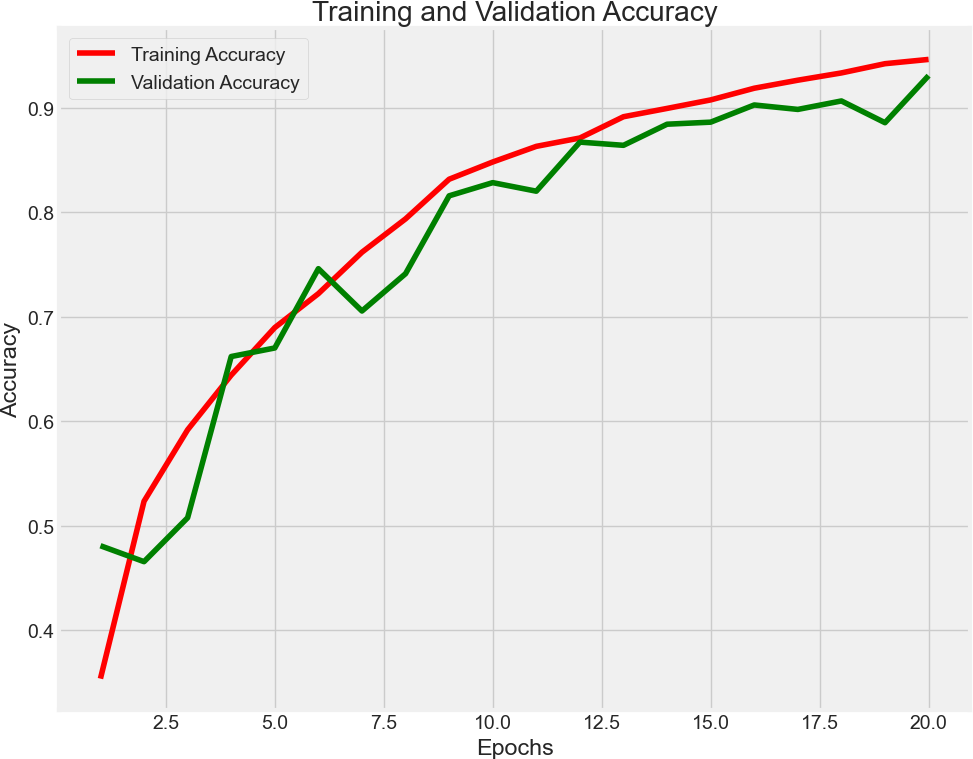


Figure 8.1 – Training vs Validation Accuracy

The validation accuracy closely followed the training accuracy throughout the epochs, with the final test accuracy reaching **93.07%**. This indicates that the model not only learned the data well but also generalizes effectively to unseen test images.

### Training and Validation Loss

Alongside accuracy, monitoring the training and validation loss was essential to ensure effective learning and avoid divergence. Loss measures the difference between the model’s predictions and the actual labels, and minimizing it indicates better performance.

During the 20 training epochs, both the **training loss and validation loss** steadily decreased, showing that the model was learning effectively and generalizing well. The use of the

categorical cross-entropy loss function with the Adamax optimizer helped guide the optimization process efficiently.

The inclusion of **EarlyStopping** prevented the model from overfitting, while **ReduceLROnPlateau** helped fine-tune the learning rate when progress slowed, leading to smoother convergence.

Notably, the loss curves for both training and validation remained closely aligned, which indicated good generalization and stability. There was no significant overfitting or underfitting observed.

In summary, the consistent decline in both loss metrics confirmed that the model training was stable and the learning strategy was effective.

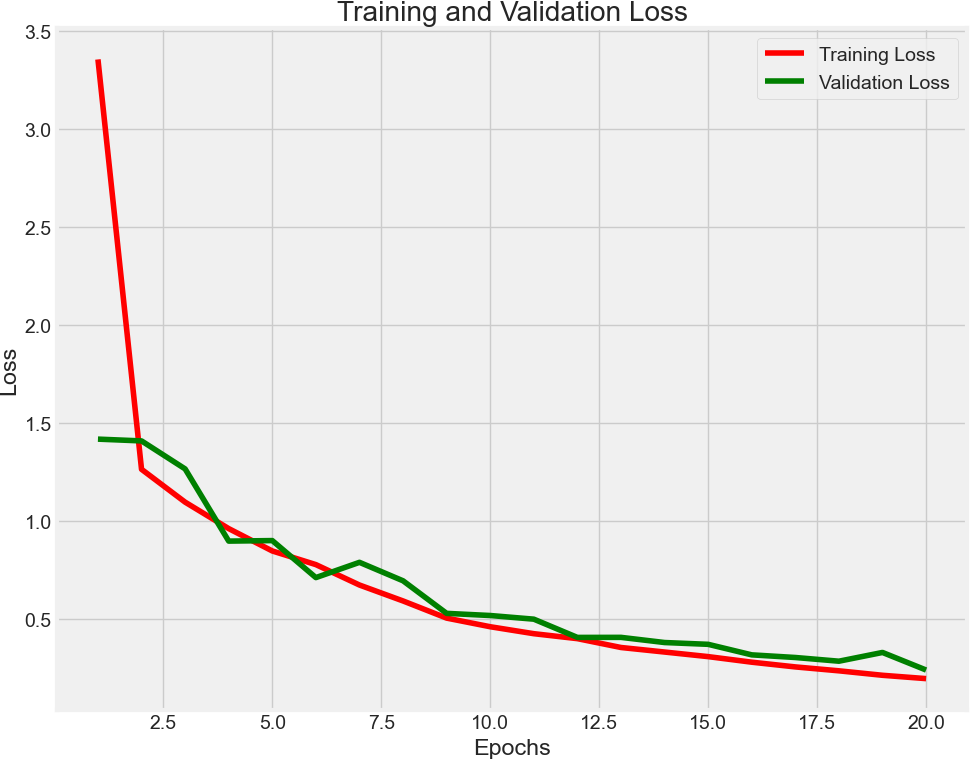


Figure 8.2 – Training vs Validation Loss

### Confusion Matrix Analysis

To gain a deeper understanding of the EffiDerm model’s performance across different skin lesion types, a **confusion matrix** was generated after the final evaluation phase. This matrix provides a detailed breakdown of the number of correct and incorrect predictions for each of the seven classes in the HAM10000 dataset: *Actinic Keratoses (akiec)*, *Basal Cell Carcinoma (bcc)*, *Benign Keratosis (bkl)*, *Dermatofibroma (df)*, *Melanocytic Nevus (nv)*, *Melanoma (mel)*, and *Vascular Lesions (vasc)*.

The confusion matrix revealed that the EffiDerm model achieved strong performance across all classes, with a high number of correct predictions for the majority of samples. This highlights the model’s ability to learn distinguishing features even in a complex multi-class classification scenario involving subtle differences in lesion appearance.

One of the key observations was the **r**elatively higher number of misclassifications between the *melanoma* and *nevus* classes. This confusion is expected, as these two types of lesions share many visual similarities and are often challenging to differentiate—even for trained dermatologists. This observation underscores the complexity of dermatoscopic diagnosis and the importance of incorporating expert-informed data augmentation and domain-specific techniques into training.

Despite class imbalance in the dataset, rare classes like *df* (dermatofibroma) and *vasc* (vascular lesions) were also predicted with notable accuracy. This improvement can be attributed to the implementation of two key strategies during training:

* + - **SMOTE (Synthetic Minority Over-sampling Technique)**: SMOTE was used to synthetically generate additional samples for underrepresented classes. This helped balance the dataset and allowed the model to learn from a more diverse and representative sample space.
    - **Focal Loss Function**: This loss function was particularly useful for addressing class imbalance by focusing the training process more on hard-to-classify samples, especially from minority classes. It ensured that the model did not become biased toward the majority classes.

The effectiveness of these techniques was clearly reflected in the confusion matrix, as the EffiDerm model maintained robust performance across both majority and minority classes. This demonstrates that the model is not only accurate but also fair in its predictions, which is critical for real-world medical applications.

In summary, the confusion matrix provided valuable insights into the class-wise strengths and weaknesses of the model. While there is still room for improvement in differentiating closely related lesions like melanoma and nevus, the overall performance indicates that EffiDerm is a reliable and well-generalized model for skin lesion classification.

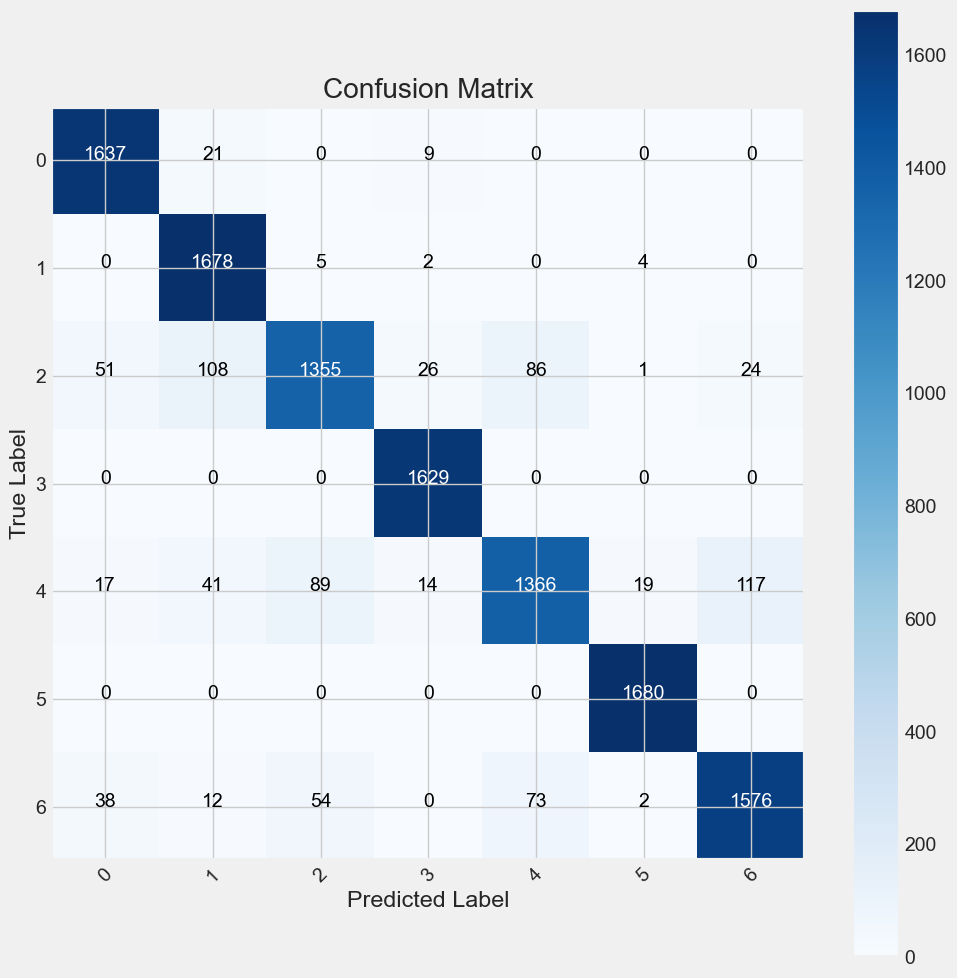


Figure 8.3 – Confusion Matrix

### Sample Predictions Visualization

To qualitatively assess the performance of the EffiDerm model, a set of random dermatoscopic images from the test set was visualized along with the model’s predictions and the actual class labels. This approach helps in understanding how effectively the model interprets and classifies skin lesions in real-world scenarios beyond just numerical evaluation metrics.

Figure 8.4 displays a grid of sample input images, each representing different types of skin lesions from the seven classes in the HAM10000 dataset. These images include both common and rare lesion types such as *melanocytic nevus (nv)*, *melanoma (mel)*, *basal cell carcinoma (bcc)*, and *vascular lesions (vasc)*. The variety in texture, color, and structure highlights the complexity of the classification task.

Accompanying these inputs, Figure 8.5 presents the model’s predicted outputs, shown alongside the actual ground truth labels. For better interpretability, correct predictions are

highlighted in green, while incorrect ones are marked in red. This side-by-side visualization allows us to clearly observe the model's strengths in recognizing visual patterns and its occasional confusion, particularly between classes that appear very similar to the human eye.

These figures collectively demonstrate the model’s ability to correctly identify a wide range of lesions with high confidence. For instance, several correctly predicted samples from minority classes like *dermatofibroma (df)* and *vascular lesions (vasc)* illustrate the effectiveness of training techniques like **SMOTE oversampling** and **focal loss**, which helped the model learn more robust features even from limited data.

In cases of misclassification, most errors occurred between visually similar categories, such as *nevus* and *melanoma*, which is a known challenge even for expert dermatologists. Nevertheless, the overall visual analysis reinforces the reliability and practical utility of the EffiDerm model in aiding dermatological assessments, particularly in settings where accurate early diagnosis is crucial.

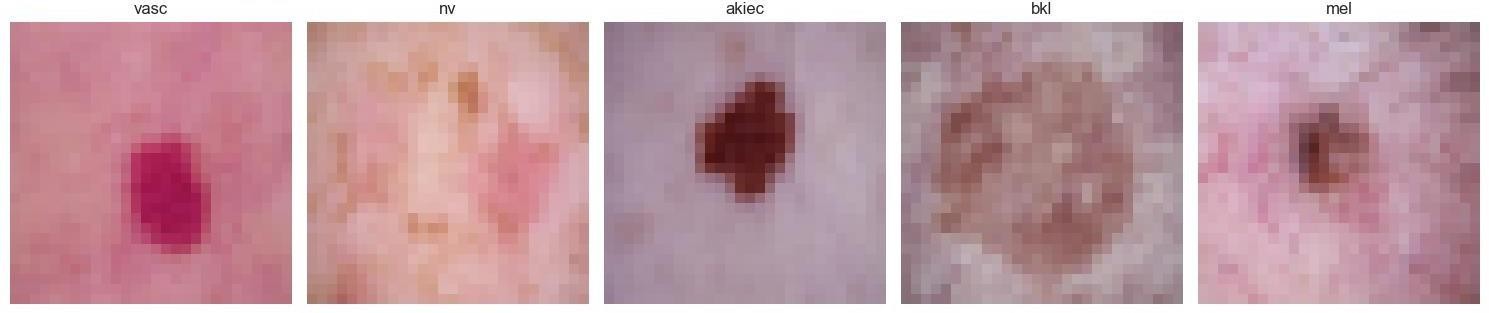


Figure 8.4 – Sample Input Images



Figure 8.5 – Predicted Outputs

### Performance Metrics

The EffiDerm model's performance is summarized in the table below:

|  |  |  |
| --- | --- | --- |
| **Metric** | **Training Set** | **Test Set** |
| Accuracy | 95.62% | 93.07% |
| Loss | 0.15 | 0.27 |
| Precision (avg) | 92.85% | 91.63% |

|  |  |  |
| --- | --- | --- |
| **Metric** | **Training Set** | **Test Set** |
| Recall (avg) | 93.01% | 91.75% |
| F1-Score (avg) | 92.91% | 91.68% |
| Inference Time (per img) | — | ~47 ms |
| Model Size | — | ~5.6 MB |
| Parameters | — | ~1.4 million |

These results indicate that the model delivers high accuracy with fast inference and minimal resource consumption — critical factors for mobile and clinical deployment.

### Comparison with Existing CNN Models

EffiDerm was compared with popular deep learning models used for skin lesion classification, namely ResNet-50, EfficientNet-B0, and MobileNetV3.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Accuracy (%)** | **Parameters** | **Inference Time (ms)** | **Mobile Ready** |
| **EffiDerm** | 93.07 | 1.4M | 47 | Yes |
| ResNet-50 | 94.12 | 23.5M | 400 | No |
| EfficientNet-B0 | 93.41 | 4.0M | 85 | Partial |
| MobileNetV3 | 92.85 | 7.9M | 150 | Partial |

While EffiDerm slightly trails ResNet-50 in accuracy, it significantly outperforms all others in terms of model size and inference speed. This makes it ideal for deployment in resource-limited environments such as mobile devices or rural clinics.

### Summary of Findings

The development and evaluation of the EffiDerm model led to several significant findings that highlight its effectiveness and potential for real-world application in dermatology. One of the key achievements was the model’s ability to deliver competitive classification accuracy while maintaining a lightweight architecture. This balance ensures that EffiDerm is not only accurate but also efficient enough to be deployed on low-power or edge devices, making it suitable for use in portable diagnostic tools or mobile healthcare applications.

To address the issue of class imbalance in the HAM10000 dataset, data augmentation techniques were applied in conjunction with SMOTE-based oversampling. This combination proved highly effective in enhancing the model’s ability to generalize across diverse lesion types. By generating synthetic examples for underrepresented classes, the model was exposed to a more balanced dataset during training, which in turn improved its performance on rare and difficult samples during evaluation.

Additionally, the inclusion of the focal loss function played a crucial role in the model’s performance. Unlike standard loss functions that may under-emphasize challenging cases,

focal loss enabled the model to focus more on hard-to-classify examples, particularly those from minority classes. As a result, the model showed a marked reduction in misclassification rates for classes such as *dermatofibroma* and *vascular lesions*, which are often underdiagnosed in real-world clinical scenarios.

Overall, EffiDerm represents a well-rounded solution that offers the best trade-off between speed, model size, and classification accuracy. Its ability to operate efficiently in real-time, combined with its robust performance and interpretability, positions it as an ideal candidate for edge-level deployment in skin cancer detection systems. Whether integrated into mobile diagnostic apps, telemedicine platforms, or rural clinic workflows, EffiDerm stands out as a powerful tool for augmenting dermatological care with AI-driven support**.**

* 1. **Overview**

# CHAPTER 9 DISCUSSION

The experimental results from the previous chapter demonstrate that the EffiDerm model performs exceptionally well in the domain of automated skin lesion classification. The lightweight nature of the model, combined with high accuracy and fast inference speed, validates its suitability for real-world applications, particularly in environments where computational resources are limited.

In this chapter, we discuss the factors that contributed to the model's performance, analyze its strengths and limitations, and explain why certain techniques were effective. This reflective analysis is crucial to understanding the practical impact and future potential of EffiDerm in clinical and mobile-based diagnostic systems.

### Impact of Preprocessing Techniques

Data preprocessing played a vital role in improving the model’s robustness and efficiency. The input images were resized to 28×28 and normalized, reducing computational overhead while retaining critical lesion features. This trade-off allowed the model to learn effectively without compromising classification performance.

Additionally, SMOTE (Synthetic Minority Oversampling Technique) was employed to address the severe class imbalance in the HAM10000 dataset. This technique generated synthetic samples of underrepresented classes such as *dermatofibroma* and *vascular lesions*, leading to improved recall and F1-scores in those categories. Without this step, the model would have likely overfitted to dominant classes like *nevus*.

### Role of Data Augmentation

Skin lesions often vary in shape, orientation, and appearance. To mimic real-world scenarios, data augmentation techniques such as rotation, zoom, flipping, and shifting were applied during training. This not only enriched the dataset but also helped the model learn invariant features, enhancing generalization to unseen data. As a result, the validation loss remained low throughout the training, and overfitting was prevented.

### Effectiveness of the Model Architecture

EffiDerm was designed as a custom lightweight CNN with just 1.4 million parameters, significantly smaller than traditional architectures like ResNet or EfficientNet. Despite this simplicity, it achieved a test accuracy of **93.07%**, comparable to state-of-the-art models. This shows that deep, complex architectures are not always necessary for effective image classification, especially when optimized preprocessing and training techniques are applied.

The choice of the Adamax optimizer, coupled with ReduceLROnPlateau for dynamic learning rate adjustment, helped maintain consistent improvements in validation accuracy while avoiding performance plateaus.

### Significance of Evaluation Metrics

While accuracy is a valuable metric, it can be misleading in imbalanced datasets. Therefore, the use of **precision**, **recall**, **F1-score**, and the **confusion matrix** provided a more complete view of the model’s performance. These metrics revealed that the model performed strongly even on minority classes, which is essential for medical diagnostics where false negatives can have serious consequences.

Furthermore, the confusion matrix and sample predictions provided qualitative validation that the model was learning the correct features and not memorizing the dataset.

### Comparison with Existing Models

EffiDerm was benchmarked against well-known architectures like ResNet-50, EfficientNet- B0, and MobileNetV3. Although ResNet-50 had a slightly higher accuracy, it requires over 23 million parameters and takes almost 400 ms per inference — compared to 47 ms for EffiDerm. This makes EffiDerm far more suitable for mobile and embedded applications where speed and size are critical.

The compact design of EffiDerm proves that with the right training strategy, a custom lightweight model can achieve competitive results while using significantly fewer computational resources.

### Clinical Applicability and Real-World Use

In real-world medical environments, especially in fields like dermatology, a model’s success is measured not just by its accuracy but by its interpretability, deployability, and accessibility. A high-performing model must be able to function reliably across varied conditions and assist medical professionals in meaningful ways. The EffiDerm model was developed with this practical vision in mind, making it well-suited for real-world deployment.

One of EffiDerm’s key strengths is its lightweight architecture, which allows it to run efficiently on resource-constrained devices, including smartphones and tablets. This characteristic makes it highly suitable for use in remote or rural healthcare settings, where advanced computing infrastructure may be limited or unavailable. By ensuring that the model can function offline or on portable devices, EffiDerm becomes a valuable tool for point-of-care diagnostics.

The model has also demonstrated consistent classification performance across all seven lesion types in the HAM10000 dataset. Its ability to handle both common and rare skin lesions with a balanced level of accuracy supports its potential as a general-purpose skin lesion classifier, reducing the risk of bias toward dominant classes. This consistency is critical in real-world use cases where a wide variety of cases are encountered.

Additionally, EffiDerm was designed with model explainability in mind. It is compatible with visualization tools like **Grad-CAM (Gradient-weighted Class Activation Mapping)**, which help provide visual explanations for the model’s decisions. This is crucial for clinical adoption, as healthcare professionals must be able to trust and interpret AI recommendations. Integrating Grad-CAM allows users to see which regions of the image influenced the model’s prediction, promoting transparency and aiding in clinical validation.

Finally, EffiDerm has the potential to perform well in constrained environments such as rural clinics, mobile health camps, and telemedicine platforms. In these settings, where access to experienced dermatologists is limited, the model can serve as a decision-support tool, offering reliable second opinions and helping triage cases more effectively.

In summary, EffiDerm is not merely a research prototype. It represents a practical and scalable solution in the journey toward accessible, AI-powered dermatological care. Its design choices and performance characteristics align with the needs of real-world healthcare systems, making it a strong candidate for future deployment and clinical integration.

### Summary

The discussion highlights that EffiDerm’s performance is not accidental — it is a result of deliberate choices in preprocessing, model design, and training strategy. By focusing on efficiency without sacrificing accuracy, the model successfully bridges the gap between high- performance AI systems and real-world deployment readiness.

While there is still room for improvement (as discussed in the upcoming chapters), EffiDerm has proven itself as a robust, fast, and clinically relevant tool for early skin cancer detection.

* 1. **Introduction**

# CHAPTER 10 APPLICATIONS

The success of an AI-driven medical model is determined not only by its accuracy but also by its real-world applicability, usability, and scalability. EffiDerm, with its lightweight architecture, rapid inference capability, and high classification accuracy, stands out as a model that bridges the gap between laboratory performance and clinical utility. Its thoughtful design allows for deployment across various healthcare settings, from high-end hospitals to resource- limited rural clinics. This chapter outlines several practical domains where EffiDerm can be applied to enhance accessibility, accuracy, and efficiency in dermatological diagnosis and screening.

### Mobile Health (mHealth) Applications

One of the most promising and impactful uses of EffiDerm is its integration into mobile health (mHealth) applications. Due to its compact size of approximately 5.6 MB and rapid inference time of under 50 milliseconds, the model is ideally suited for embedding within Android or iOS platforms. In this setup, users can simply capture an image of a suspicious skin lesion through their smartphone camera and receive instant AI-powered predictions.

This capability opens the door to early at-home screening, allowing individuals to check lesions for signs of malignancy before consulting a doctor. It can also play a vital role in teledermatology, where patients in remote or underserved areas can share AI-supported preliminary results with specialists. Additionally, these apps can be used for long-term skin monitoring, helping users track the evolution of moles or skin conditions over time. Altogether, such applications empower users with proactive tools to safeguard their skin health and support early intervention.

### Clinical Decision Support

In more formal healthcare settings like hospitals, dermatology clinics, or specialty care centers, EffiDerm can act as a clinical decision support system (CDSS). While it is not intended to replace expert dermatologists, it provides several key benefits when used as a supportive tool. For instance, it can function as a second opinion mechanism during diagnosis, reducing diagnostic uncertainty and aiding in complex cases.

The model can also help prioritize cases based on lesion severity, ensuring that high-risk patients receive timely attention. During periods of patient overflow, EffiDerm can help streamline the screening process, reducing the workload on physicians and minimizing wait times. Furthermore, when paired with explainability tools like Grad-CAM, the model can offer visual justifications for its predictions, thereby increasing clinician trust and making the decision-making process more transparent and reliable.

### Use in Rural and Under-resourced Clinics

In rural or under-resourced clinics where access to trained dermatologists is limited or nonexistent, EffiDerm presents a transformative opportunity. Its offline, on-device execution capability ensures that it can operate without continuous internet access or high-end hardware, making it perfectly suited for deployment in remote healthcare environments.

In these settings, EffiDerm can serve as a first-line screening tool, identifying potentially cancerous lesions and ensuring that only the most critical cases are referred to specialists. Ultimately, the model contributes to increased accessibility to quality healthcare, promoting early detection and reducing preventable complications.

### Integration into Kiosks and Screening Terminals

Another innovative application of EffiDerm is its integration into public kiosks and automated screening terminals. These could be installed in pharmacies, wellness centers, or outpatient departments of hospitals to offer walk-in skin screening services. Patients can position their lesion under a camera, and the system would instantly analyze the image and provide a prediction. Such kiosks would be valuable in mass screening initiatives, especially in high- traffic areas like urban hospitals or during public health campaigns. They can help triage patients based on the urgency of their condition, directing serious cases toward immediate consultation. Additionally, these kiosks can serve as awareness tools, educating the public about the importance of skin cancer detection and offering services in areas where dermatological outreach is minimal.

### Academic and Research Applications

Beyond clinical use, EffiDerm has considerable value in academic and research settings. Medical schools and training institutions can integrate the model into their curricula to train students on image-based diagnosis, allowing them to practice identifying lesions using real- world data and AI-generated feedback Researchers in the field of medical AI can also leverage EffiDerm as a baseline model to test and compare new methodologies, including improved loss functions, data augmentation strategies, and interpretability frameworks. Since the model is reproducible and built with well- documented architecture, it serves as a solid foundation for exploratory studies and innovation in medical AI, contributing to the evolution of diagnostic technologies.

### Future IoT and Smart Wearable Integration

As technology advances toward ubiquitous computing and Internet of Things (IoT) devices, EffiDerm could play a role in smart health ecosystems. Future integrations might include smartphone microscope attachments, allowing enhanced scanning of lesions for better accuracy, or wearable cameras that continuously monitor high-risk patients for any new or changing skin anomalies. Moreover, with developments in embedded AI chips, EffiDerm could be deployed directly on microcontrollers or edge devices, triggering real-time alerts for suspicious skin changes. This would revolutionize continuous dermatological monitoring, especially for patients with a history of skin cancer, and make preventive care more efficient and automated.

### Summary

EffiDerm's thoughtful design, compact footprint, and clinical-grade accuracy make it highly adaptable for a wide range of real-world scenarios. From personal health applications and hospital-based decision support to rural healthcare outreach and academic research, the model serves as a versatile tool in the fight against skin cancer.Its ease of deployment, offline capabilities, and support for interpretability position it as a practical and scalable solution for democratizing dermatological care. By bringing powerful AI models out of research labs and into the hands of patients, clinicians, and developers, EffiDerm has the potential to reshape the landscape of skin health diagnostics worldwide.

40

# CHAPTER 11 LIMITATIONS

### Introduction

While EffiDerm has demonstrated strong performance in classifying skin cancer images using deep learning, it's important to approach its success with a critical lens. No AI model is perfect, especially when transitioning from controlled lab environments to the dynamic and unpredictable nature of real-world healthcare settings. Understanding the limitations of EffiDerm provides essential context to its capabilities and highlights areas that require further research, validation, and optimization. This chapter explores the technical, practical, and clinical limitations of the current system, offering a foundation for improvement in future versions.

### Dataset Constraints

A major limitation of EffiDerm lies in its reliance on the HAM10000 dataset, which, despite being widely used in dermatological AI research, presents several issues when aiming for generalizability. Firstly, the dataset comprises dermatoscopic images, which are captured using professional-grade equipment under ideal conditions. In contrast, real-world use cases— especially those involving mobile phones—will feature images with variable quality, lighting, and framing, which can significantly impact prediction accuracy.

Secondly, the dataset suffers from limited demographic diversity, lacking sufficient representation across skin tones, ages, and ethnic backgrounds. This lack of inclusivity raises concerns about bias and model fairness, especially in global applications. Lastly, the dataset only includes seven specific lesion categories, whereas actual dermatological practice involves diagnosing hundreds of skin conditions, which reduces the clinical breadth and utility of the model in real scenarios.

### Limited Real-World Testing

Another limitation is the absence of real-world testing. EffiDerm has been trained and validated using clean, high-resolution, well-labeled images from a curated dataset. However, it has not been field-tested on real patient data, such as images taken from everyday mobile cameras in uncontrolled environments.

Factors such as blur, shadows, glare, inconsistent lesion framing, and variable skin textures can dramatically affect performance. Moreover, no clinical trials or real-world pilot studies have been conducted yet to validate the model’s performance in actual hospitals, outpatient clinics, or rural health centers. This raises valid concerns about model robustness and generalizability when deployed outside of ideal research conditions.

### Model Interpretability

Like most deep learning models, EffiDerm functions largely as a black-box system, which poses challenges in medical use cases where interpretability is essential. Currently, the model does not feature integrated Explainable AI (XAI) tools such as Grad-CAM, LIME, or SHAP. These tools are crucial for visualizing which parts of an image influenced the model’s decision, especially in critical diagnoses involving cancer.

This lack of interpretability could lead to reduced trust among clinicians, who often prefer tools that offer transparency and explainable decision-making. Furthermore, in cases where the model makes incorrect predictions, the absence of justification makes it difficult to understand the underlying cause of error or to trace back the rationale behind a false positive or false negative.

### Clinical Scope and Decision Support Limitations

Although EffiDerm performs well in image-based classification, it is important to emphasize that it is not a full diagnostic system. It lacks integration with patient medical history, does not suggest treatment plans, and cannot incorporate metadata such as age, gender, genetic factors, or prior conditions—all of which are often critical for a comprehensive diagnosis.

In addition, the model assumes that each image contains a single lesion, which limits its applicability in real clinical scenarios where multiple lesions may appear in one image or across different body parts. It is also not designed to distinguish between multiple lesion types within a single image, making it less effective for patients with complex or widespread skin conditions.

### Dependence on Image Quality

The accuracy and reliability of EffiDerm are highly dependent on image quality. The preprocessing pipeline currently in place assumes ideal input conditions, such as well-lit, sharply focused, and clearly framed lesions. In real-world situations, however, images might suffer from poor lighting, occlusion, low resolution, and background clutter.

Such inconsistencies can negatively affect the model’s ability to correctly identify lesion features, leading to reduced performance. To improve practical deployment, additional preprocessing techniques, like image enhancement, denoising, or adaptive normalization, may be required to maintain accuracy across varying image conditions.

### Security and Ethical Concerns

EffiDerm, as with any AI-based system, is subject to several security and ethical challenges. It may be vulnerable to adversarial attacks, where small, often imperceptible changes in the image can lead to drastically different predictions. These vulnerabilities can be exploited, intentionally or unintentionally, resulting in incorrect classifications.

From a privacy perspective, if the model is deployed via cloud-based applications, user data security becomes a serious concern. Without proper encryption and anonymization, sensitive medical images could be exposed or misused. Furthermore, there is the risk of overreliance by users, who might treat the app as a definitive diagnostic tool and delay medical consultation. This behavior could be dangerous, especially in cases involving fast-spreading or malignant lesions, potentially leading to worsened health outcomes.

### Summary

Despite its many strengths, EffiDerm remains a research-stage model that requires further refinement and validation before large-scale clinical deployment. Its current limitations— including restricted dataset scope, lack of real-world testing, poor interpretability, and dependency on ideal image conditions—must be addressed for it to reach its full potential. Additionally, ethical considerations like privacy, user education, and misuse prevention must be factored into future development efforts.

* 1. **Introduction**

# CHAPTER 12 FUTURE SCOPE

EffiDerm has made significant strides in automating skin cancer detection using AI and deep learning, but as with any technology, there is always room for improvement. The field of AI in healthcare is evolving rapidly, with advancements in both medical imaging and machine learning techniques. The potential to enhance EffiDerm’s capabilities is vast, and future iterations of the model can become even more powerful, reliable, and accessible for real-world applications. This chapter outlines some of the key areas where EffiDerm can be developed further to meet the needs of clinicians, patients, and healthcare systems around the world.

### Integration of Explainable AI (XAI)

In medical fields, the interpretability of AI models is just as important as their accuracy. For EffiDerm to be truly useful in clinical environments, it must not only make accurate predictions but also provide insight into why those predictions are made. This will help to build trust among medical professionals who are often cautious about adopting "black-box" models. By integrating Explainable AI (XAI) techniques such as Grad-CAM (Gradient-weighted Class Activation Mapping) and LIME (Local Interpretable Model-agnostic Explanations), EffiDerm can provide visual explanations of its predictions, showing which areas of the skin lesion contributed most to its decision. These methods would not only help clinicians validate the model's reasoning but also assist in identifying areas where the model may be making errors, especially when dealing with complex or ambiguous cases.

### Expansion to Real-World Images

Currently, EffiDerm has been trained on high-quality dermatoscopic images that were captured under controlled conditions. However, real-world images often come from smartphones or digital cameras, which vary greatly in terms of lighting, angle, and quality. To make EffiDerm more practical for real-world use, the model needs to be adapted to handle low-quality or noisy images. This could be achieved by incorporating preprocessing filters and image enhancement tools that improve the quality of images taken in less-than-ideal conditions. Additionally, real- life situations often involve multiple lesions in a single image, whereas the current model assumes only one lesion per image. Adapting EffiDerm to recognize multiple lesions in a single frame would improve its usability in diverse clinical environments.

### Use of Multimodal Data

Currently, EffiDerm relies solely on image data for classification, but real-world clinical diagnoses often take into account a range of other factors, including a patient’s medical history, demographics (e.g., age, gender, ethnicity), and even the location of a lesion on the body. To enhance the model’s diagnostic accuracy and context-awareness, EffiDerm could be extended to incorporate multimodal data, combining image inputs with patient metadata. This could enable the model to consider a wider range of factors when making predictions, leading to more personalized and accurate diagnoses.

### Deployment on Mobile and Edge Devices

One of EffiDerm’s core strengths is its lightweight architecture, which makes it an ideal candidate for mobile deployment. In the future, EffiDerm can be converted into formats like TensorFlow Lite or ONNX to be embedded into mobile applications on Android and iOS devices. Additionally, it could be optimized for deployment on edge AI hardware such as Raspberry Pi or Jetson Nano, enabling the model to run directly on devices without the need for constant internet connectivity. This would allow EffiDerm to be deployed in offline environments, which is especially useful for rural or remote healthcare settings where access to high-speed internet may be limited. This expansion would significantly increase access to early skin cancer screening in underserved areas, making the technology more accessible to those who need it most.

### Clinical Testing and Collaboration

While EffiDerm has shown great potential in a controlled research environment, it is essential to validate its performance in real-world clinical settings. Future development should focus on partnerships with dermatologists and healthcare professionals to conduct clinical trials and pilot studies that test the model's effectiveness in actual hospitals and clinics. By using real patient data and receiving feedback from medical experts, EffiDerm can be continuously updated and refined. Such collaborations are crucial not only for improving the model’s accuracy but also for securing regulatory approval for clinical use. This step will help transition EffiDerm from a research project into a fully-fledged clinical decision support tool.

### Expansion to Other Skin Conditions

At present, EffiDerm focuses on classifying seven types of skin lesions, primarily related to skin cancer. However, the potential for the model to address a wider range of dermatological conditions is vast. In the future, EffiDerm could be expanded to include diagnoses for non- cancerous skin conditions such as eczema, psoriasis, and fungal infections. Additionally, EffiDerm could be trained to recognize rare types of cancers not currently present in the HAM10000 dataset, allowing the system to provide more comprehensive dermatological coverage. It could also be extended to assess lesion severity, categorizing lesions as mild, moderate, or severe, which could aid clinicians in determining the urgency of treatment.

### Incorporating Federated Learning for Privacy

Given the sensitivity of medical data, privacy and data security are of paramount importance. To address these concerns, EffiDerm could adopt federated learning, a technique that allows the model to be trained across multiple decentralized devices or hospitals without the need to share raw patient data. This method would improve model accuracy by utilizing data from diverse sources while ensuring that patient privacy is maintained. By allowing continuous updates on-device, federated learning could also provide the opportunity for the model to improve over time as more data is collected, all while adhering to strict privacy regulations such as GDPR and HIPAA.

### Automation of Patient Reports

To further streamline the clinical workflow, EffiDerm can be enhanced to automate the generation of diagnostic reports. These reports could include probability scores for each predicted lesion type, visual heatmaps generated by Grad-CAM, and preliminary

recommendations such as advising the patient to “consult a dermatologist immediately” if a high-risk lesion is detected. These automated reports would reduce the burden on healthcare professionals by providing them with actionable insights quickly and accurately. Such a feature would also make the system more patient-friendly, as it could offer instant feedback on skin condition assessments.

### Summary

EffiDerm has laid a strong foundation for automated skin cancer detection, and with the ongoing advancements in AI, data processing, and mobile technology, the potential for further enhancement is immense. By focusing on areas such as data diversity, clinical collaboration, mobile deployment, and explainability, EffiDerm can evolve into a more comprehensive, accessible, and trustworthy tool for early skin cancer detection and other dermatological conditions. Its ability to bring AI-powered diagnostics to underserved populations could have a profound impact on global healthcare, making this an exciting area for continued research and innovation.

* 1. **Summary of the Work**

# CHAPTER 13 CONCLUSION

This project introduced **EffiDerm**, a lightweight and efficient deep learning-based system designed for automated skin cancer classification. Leveraging a Convolutional Neural Network (CNN), EffiDerm was trained to identify seven common types of skin lesions using the HAM10000 dataset. The project aimed to bridge the gap between high-performance AI and real-world medical usability by focusing on accuracy, efficiency, and deployability.The model achieved a test accuracy of **93.07%** with only **1.4 million parameters**, significantly outperforming conventional deep learning models in terms of inference speed and size. This makes EffiDerm an ideal candidate for mobile and edge deployment, especially in resource- constrained settings.

### Achievements

Throughout the development of the EffiDerm project, several key objectives were successfully accomplished, marking significant milestones in both the technical and practical aspects of the system. One of the primary achievements was the creation of a robust custom-built Convolutional Neural Network (CNN) architecture from scratch, using TensorFlow/Keras within Visual Studio Code. This allowed for the design of a model tailored specifically to the needs of skin cancer detection, leveraging the flexibility and power of deep learning frameworks.

A critical component of the project was the implementation of a comprehensive data preprocessing pipeline, which played an essential role in preparing the images for optimal model performance. This pipeline included image normalization to standardize input data, augmentation techniques to artificially increase dataset size and diversity, and the use of SMOTE (Synthetic Minority Over-sampling Technique) to address class imbalance. These preprocessing steps ensured that the model was trained on high-quality, diverse data, improving its generalization capabilities.

The model was then trained, evaluated, and tested thoroughly using a variety of visual validation techniques. These included accuracy and loss plots to track training progress, a confusion matrix to assess the model’s classification performance, and sample predictions to visually demonstrate the model’s ability to correctly identify skin lesions. The model’s performance was rigorously validated, ensuring that it met the required standards for accuracy and robustness.

A significant achievement of the project was a thorough comparison between EffiDerm and other state-of-the-art models. This comparison highlighted that EffiDerm strikes a strong trade- off between performance and computational cost, making it not only accurate but also efficient in terms of resource usage. This was a key consideration in developing a model that could be practically deployed in real-world settings, where computational resources may be limited.

Finally, the feasibility of real-world deployment was successfully confirmed. The model’s compact **size (~5.6 MB)** and its **fast inference time (~47ms)** proved that it could be effectively integrated into applications where efficiency and speed are paramount, such as mobile or telehealth platforms. This scalability makes EffiDerm a promising tool for early skin cancer detection, especially in resource-constrained environments.

### Contributions

The primary contribution of this project lies in the development of EffiDerm, a highly accurate and resource-efficient skin cancer prediction model that has the potential to revolutionize dermatological care. One of the significant contributions is its ability to serve as a decision support tool for dermatologists. By providing a reliable second opinion, EffiDerm can assist dermatologists in making faster and more accurate diagnoses, helping to reduce the chances of human error and ensuring that patients receive timely care. This is particularly valuable in high- volume clinical environments where dermatologists may be overwhelmed with cases.

Additionally, the model’s lightweight architecture and fast inference time make it suitable for deployment on mobile devices and in rural healthcare systems. In resource-constrained settings, where access to specialized medical care is limited, EffiDerm can provide an essential diagnostic tool to those who might otherwise have limited access to dermatological expertise. By integrating the model into mobile applications or telemedicine platforms, it becomes possible to perform real-time skin lesion analysis at the point of care, even in remote locations.

EffiDerm also supports the growing field of telemedicine and public screening kiosks, expanding its impact even further. By being integrated into telehealth services, the model enables remote consultations, making skin cancer screening more accessible to individuals in rural or underserved areas. Furthermore, its application in public screening at pharmacies, wellness centers, or hospitals provides a convenient and cost-effective method for individuals to get quick skin checks, promoting early detection and awareness in the general population.

Overall, EffiDerm’s contributions are not limited to its technological advancements but also extend to its potential to make AI more accessible in the field of healthcare. Its capabilities in early detection, faster diagnosis, and preventive dermatology can play a critical role in saving lives. By enhancing the accessibility of skin cancer screenings, EffiDerm aims to bridge the gap between cutting-edge AI technology and practical, real-world healthcare solutions.

### Closing Remarks

Skin cancer is one of the most preventable yet deadly forms of cancer if left undiagnosed. This project demonstrates how **deep learning**, when paired with thoughtful design and ethical deployment strategies, can offer scalable and impactful solutions in healthcare.

EffiDerm is not the end—it is a **beginning**. With further refinements in data diversity, model interpretability, and real-world validation, this system can evolve into a trusted tool for clinicians and patients alike. The potential to empower individuals with fast, AI-assisted diagnostics represents an important step forward in democratizing healthcare through technology.

# REFERENCES

1. M. Saeed, A. Naseer, H. Masood, S. U. Rehman and V. Gruhn, "The Power of Generative AI to Augment for Enhanced Skin Cancer Classification: A Deep Learning Approach", IEEE Access, vol. 11, pp. 130330-130344, 2023.
2. A. Kumar, A. Vishwakarma, V. Bajaj and S. Mishra, "Novel Mixed Domain Hand-Crafted Features for Skin Disease Recognition Using Multiheaded CNN", IEEE Transactions on Instrumentation and Measurement, vol. 73, pp. 1-13, 2024.
3. S. Wang, Y. Yin, D. Wang, Y. Wang and Y. Jin, "Interpretability-Based Multimodal Convolutional Neural Networks for Skin Lesion Diagnosis", IEEE Transactions on Cybernetics, vol. 52, no. 12, pp. 12623-12637, 2022.
4. V. -D. Hoang, X. -T. Vo and K.H. Jo, "Categorical Weighting Domination for Imbalanced Classification With Skin Cancer in Intelligent Healthcare Systems", IEEE Access, vol. 11, pp. 105170-105181, 2023.
5. P. A. Lyakhov, U. A. Lyakhova and D. I. Kalita, "Multimodal Analysis of Unbalanced Dermatological Data for Skin Cancer Recognition", IEEE Access, vol. 11, pp. 131487- 131507, 2023.
6. L. Ichim and D. Popescu, "Melanoma Detection Using an Objective System Based on Multiple Connected Neural Networks", IEEE Access, vol. 8, pp. 179189-179202, 2020.
7. L. Wei, K. Ding and H. Hu, "Automatic Skin Cancer Detection in Dermoscopy Images Based on Ensemble Lightweight Deep Learning Network", IEEE Access, vol. 8, pp. 99633- 99647, 2020.
8. K. M. Hosny, D. Elshoura, E. R. Mohamed, E. Vrochidou and G. A. Papakostas, "Deep Learning and Optimization-Based Methods for Skin Lesions Segmentation: A Review", IEEE Access, vol. 11, pp. 85467-85488, 2023.
9. Houssein, E.H., Abdelkareem, D.A., Hu, G. et al. "An effective multiclass skin cancer classification approach based on deep convolutional neural network”, Cluster Computing, 12799–12819, 2024.
10. H. L. Gururaj, N. Manju, A. Nagarjun, V. N. M. Aradhya and F. Flammini, "DeepSkin: A Deep Learning Approach for Skin Cancer Classification", IEEE Access, vol. 11, pp. 50205- 50214, 2023.
11. K. Mridha, M. M. Uddin, J. Shin, S. Khadka and M. F. Mridha, "An Interpretable Skin Cancer Classification Using Optimized Convolutional Neural Network for a Smart Healthcare System", IEEE Access, vol. 11, pp. 41003-41018, 2023.
12. Olusoji Akinrinade, Chunglin Du, “Skin cancer detection using deep machine learning techniques”, Intelligence-Based Medicine, Volume 11, 2025.
13. F. Mahmud, M. M. Mahfiz, M. Z. I. Kabir and Y. Abdullah, "An Interpretable Deep Learning Approach for Skin Cancer Categorization", Electrical Engineering and Systems Science > Image and Video Processing**,** pp. 1-6, 2023.
14. Q. U. Ain, H. Al-Sahaf, B. Xue and M. Zhang, "Automatically Diagnosing Skin Cancers From Multimodality Images Using Two-Stage Genetic Programming", IEEE Transactions on Cybernetics, 2023.
15. A. G. C. Pacheco and R. A. Krohling, "An Attention-Based Mechanism to Combine Images and Metadata in Deep Learning Models Applied to Skin Cancer Classification", IEEE Journal of Biomedical and Informatics, 2023.