F/SOP/FYDP02/06/00

UNDERGRADUATE FINAL YEAR PROJECT

Department of Computer Science & Information Technology

NED University of Engineering and Technology

Development of Mobile Application for Early Detection of Foot Ulcer in Diabetic Patients by employing predictive models.

Group Number: 21019
Batch: 2021– 2025

Group Member Names:

Nabaha Tahir	CT-21009
Tayyaba Zahra	CT-21013
Ureel Lal Muhammad	CT-21014
Shahzeb Abro	CT-21034

Approved by

……………………………………………………………………………………..
.

Dr. Usman Amjad

Assistant Professor

Project Advisor

…………………..

Author's Declaration

We declare that we are the sole authors of this project. It is the actual copy of the project that was accepted by our advisor(s) including any necessary revisions. We also grant NED University of Engineering and Technology permission to reproduce and distribute electronic or paper copies of this project.

Signature and Date	Signature and Date	Signature and Date	Signature and Date
…………………	…………………	…………………	…………………
Nabaha Tahir	Tayyaba Zahra	Ureel Lal Muhammad	Shahzeb Abro
CT-21009	CT-21013	CT-21014	CT-21034
tahir4406585@cloud.neduet.edu.pk	zahra4402005@cloud.neduet.edu.pk	muhammad4401086@cloud.neduet.edu.pk	shahzeb4401790@cloud.neduet.edu.pk

Statement of Contributions

CT-21009 & CT-21014 - Data Collection, Preprocessing, and Annotation

- CT-21009 and CT-21014 collaborated on data collection, preprocessing, and annotation. CT-21009 gathered 206 images from Bosch Pharma and 2,000 from the Kaggle DFU dataset for binary classification. CT-21014 collected and manually cleaned 18,000 images for multi-class classification (4 classes).

- Both CT-21009 and CT-21014 worked with medical experts to annotate the dataset using binary and multi-class labeling. Categories included: Immediately Treatable, Treatable Within 4 Weeks, Complex Wound, and No Ulcers. CT-21014 led model training and fine-tuning, ensuring high annotation quality and dataset accuracy.

CT-21013 & CT-21034 - Development and System Integration

- CT-21013 and CT-21034 played key roles in developing and integrating the mobile application.
- CT-21013 focused on designing and developing the app's user interface with React Native, ensuring it was responsive and user-friendly on mobile devices. They also incorporated features such as notifications and routine scan reminders to enhance the user experience.
  CT-21034 was responsible for the backend development, particularly in integrating TensorFlow Lite models to ensure real-time, efficient image analysis on mobile platforms. They facilitated smooth communication between the machine learning models and the app, ensuring a seamless performance.
- All authors contributed equally to write the project report.

Executive Summary

Diabetic neuropathy often leads to a loss of sensation in the feet, increasing the risk of foot ulcers. If undetected, these ulcers can cause severe complications, including infections and amputations. Traditional detection methods rely on physical examinations, which are often inaccessible and insufficient, especially in rural areas where healthcare services are limited. Pakistan's healthcare system, with over 33 million adults affected by diabetes, faces significant challenges in man aging this issue.

This report presents زخم ردیاب, an AI-driven mobile application designed to address these gaps. The app empowers diabetic patients to monitor their foot health independently by detecting early signs of ulcers through image analysis. By providing real-time alerts and enabling proactive care, the app aims to reduce complications, improve patient outcomes, and enhance healthcare accessibility, particularly in underserved areas.

The project was implemented in multiple phases:

1. **Data Collection and Preprocessing:** Foot images were collected and annotated to cover various ulcer stages and demographics. The images were standardized and augmented to ensure robustness.
2. **Model Development:** A convolutional neural network (CNN) was developed using transfer learning to detect early signs of ulcers, focusing on accuracy and generalizability.
3. **Application Development**: The app was built for Android and iOS platforms, featuring real-time analysis, user alerts, and a web interface for healthcare providers to manage patient data.
4. **Deployment and Security**: The app leverages cloud and edge computing for accessibility, with encryption and authentication protocols ensuring data security.

The findings demonstrate that زخم ردیاب improves the accuracy and timeliness of ulcer detection, enabling patients to take proactive measures and reducing the burden on healthcare facilities. By offering a scalable and cost-effective solution, the app addresses critical challenges in managing diabetic foot ulcers.

Acknowledgments

We would like to thank Dr. Usman Amjad, our FYDP supervisor, for his invaluable guidance, mentorship, and support throughout this project. His expertise and encouragement were instrumental in shaping the methodology and ensuring the project's successful execution

Dedication

This work is dedicated to the indigenous and underprivileged communities who lack access to basic healthcare facilities, with the hope that this project can contribute to improving their well-being and healthcare access.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

CNN: Convolutional Neural Network

DFU: Diabetic Foot Ulcer

DFUC: Diabetic Foot Ulcer Challenge

AWS: Amazon Web Services

SDG: Sustainable Development Goal

JWT: JSON Web Token

MERN: MongoDB, Express.js, React.js, Node.js

UI/UX: User Interface/User Experience

OAuth: Open Authorization

United Nations Sustainable Development Goals

The Sustainable Development Goals (SDGs) are the blueprint to achieve a better and more sustainable future for all. They address the global challenges we face, including poverty, inequality, climate change, environmental degradation, peace and justice. There is a total of 17 SDGs as mentioned below. Check the appropriate SDGs related to the project.

□ No Poverty

□ Zero Hunger

□ Good Health and Wellbeing

□ Quality Education

□ Gender Equality

□ Clean Water and Sanitation

□ Affordable and Clean Energy

□ Decent Work and Economic Growth

□ Industry, Innovation and Infrastructure

□ Reduced Inequalities

□ Sustainable Cities and Communities

□ Responsible Consumption and Production

□ Climate Action

□ Life Below Water

□ Life on Land

□ Peace and Justice and Strong Institutions

□ Partnerships to Achieve the Goals

Similarity Index Report

Following students have compiled the final year report on the topic given below for partial fulfillment of the requirement for bachelor's degree in computer science & information technology.

Project Title: Development of Mobile Application for Early Detection of Foot Ulcers in Diabetic Patients by employing predictive models.

Nabaha Tahir	Tayyaba Zahra	Ureel Lal Muhammad	Shahzeb Abro
CT-21009	CT-21013	CT-21014	CT-21034

This is to certify that Plagiarism test was conducted on complete report, and overall similarity index was found to be less than 20%, with maximum 5% from single source, as required.

Signature and Date

……………………...

Dr. Usman Amjad

Assistant Professor

Chapter 1

Introduction

1.1 Background Information

Diabetes and its complications, such as diabetic neuropathy, pose a serious global health challenge. In Pakistan alone, over 33 million people live with diabetes, and many face limited access to timely and effective healthcare, especially in rural areas where traditional healthcare systems often fail to catch diabetic foot ulcers (DFUs) early. DFUs are particularly concerning as delays in detection can lead to severe outcomes, including amputations. To tackle this critical issue, this project aims to create a userfriendly mobile application powered by artificial intelligence and computer vision. By leveraging predictive models and image analysis, the app will enable early detection of foot ulcers, empowering patients, enhancing access to care, and helping alleviate the burden on overstretched healthcare systems.

1.2 Significance and Motivation

This project holds significant promise for transforming diabetes care by facilitating the early detection of foot ulcers. It is driven by a strong motivation to address the critical need for reducing complications caused by delayed diagnoses. Aligned with the Sustainable Development Goals (SDGs), the solution directly supports improved health and well-being (SDG 3), fosters innovation in healthcare (SDG 9), and works toward reducing healthcare disparities (SDG 10). To further enhance accessibility, especially in rural and underserved regions, the app is designed to be bilingual, offering support in both English and Urdu, making it highly relevant to Pakistan's healthcare landscape.

1.3 Aims and Objectives

The primary objective of this project is to develop a mobile application that can detect diabetic foot ulcers (DFUs) early using advanced image analytics. Project Objectives include:

- Build a robust deep learning model to identify visual features of ulcers in foot images.

- Provide user-friendly features such as real-time monitoring, notifications, and reminders for routine scans.
- Implement robust cybersecurity measures to safeguard sensitive medical data.
- Especially in remote areas, use a simple and simple platform that solves health problems.

1.4 Methodology

Data collection and preliminary processing: Normalization of various leg images for collecting and analyzing various leg images.

Model Development: Train Convolutional Neural Networks (CNNs) using frameworks such as TensorFlow and Keras, with a focus on lightweight architectures such as MobileNet and EfficientNet-Lite for optimal mobile performance.

App Development: Build cross-platform mobile apps with real-time analytics, leveraging frameworks such as React Native and Flutter, integrating TensorFlow Lite for smooth performance.

Integration and Deployment: Hosting the system on cloud platforms such as AWS, providing offline functionality using edge computing, and creating a secure web portal for healthcare providers to access and monitor data.

Table 1: Methodology Breakdown

Step	**Description**
1. Data Collection and Preliminary Processing	Collection and normalization of various leg images for analysis. Ensuring data consistency and quality for model training.
2. Model Development	Train Convolutional Neural Networks (CNNs) using frameworks like TensorFlow and Keras. Focus on lightweight architectures such as MobileNet and EfficientNet-Lite for mobile optimization.

3. App Development	Develop cross-platform mobile apps using React Native or Flutter. Integrate TensorFlow Lite for on-device model inference, ensuring smooth performance on mobile devices.
4. Integration and Deployment	Host the system on cloud platforms (e.g., AWS) and implement offline functionality using edge computing. Develop a secure web portal for healthcare providers to monitor patient data.

1.5 Report Outline

This project aims to revolutionize diabetic healthcare management, offering an innovative, accessible, and efficient solution to a critical healthcare issue. The report is meticulously organized into a series of comprehensive chapters, each of which delves deeply into the critical components and stages of the project development lifecycle

Chapter 1: Introduction: Provides an overview of the project including background, significance, main objectives, and methodology used.

Chapter 2: Literature Review: Examines the existing technologies and methods currently used to detect diabetic foot ulcers (DFUs).

Chapter 3: Methodology: Define the technical framework, tools and methods used during project development.

Chapter 4: Implementation and Results: Focus on the application development process and emphasize the main test results.

Chapter 5: Discussion and Future Work: Reflect on the problems encountered, recognize limitations, and identify areas for future improvement.

This project aims to reinvent diabetes care by introducing innovative, affordable, and effective solutions that meet immediate medical needs.

Chapter 2

Literature Review

2.1 Introduction

This chapter introduces a detailed review of existing research on the detection of diabetic suspension ulcers using image processing and automated learning methods. To solve the DFU detection problem, various approaches are considered, such as implementing deep learning models, pre-treatment strategies, and the use of diverse datasets. The review focuses on identifying the strengths, weaknesses, and gaps in previous research. The chapter begins with a review of the problem statement in the literature, followed by an analysis of the datasets and pre-processing methods. It then evaluates machine learning models and their performance metrics and summarizes the main findings. By synthesizing these insights, this chapter establishes the foundations of the innovative method proposed in this work and situates its contribution in the broader field of DFU detection research**.**

Table 2: Summary of Reviewed Literature on DFU Detection

Parameter	**Count**
Total Number of Papers Reviewed	25
Total Preprocessing Techniques Discussed	19
Total Machine Learning Models Evaluated	12
Most Common Dataset Used	DFUC (2020 & 2021)
Performance Metrics Analysed	6 (Accuracy, Precision, Recall, F1-Score, ROC-AUC, Sensitivity)

This table provides a concise summary of the literature review, giving the reader a clear overview of the scope and depth of the studies reviewed. It can be referenced in the text as follows:

Table 2 provides a quantitative overview of the studies reviewed, highlighting the different pretreatment techniques and designs used, as well as the performance metrics typically assessed in DFU detection studies.

2.2 Problem Statements in Existing Research

Many studies have examined the challenges faced by diabetic patients when diabetic foot disease (DFD) is detected at an early stage. According to [1], diabetes mellitus (DM) often leads to chronic foot ulcers, which, if left untreated, can develop into severe complications such as infection and amputation. This study highlights the urgent need for accurate and timely detection using advanced technologies such as deep learning models. Shata and Abdulaziz [2] discuss the difficulties diabetic patients face when diagnosed with DFU without clinical testing.

The research of Muthuraja et al. [3] and Giridhar et al. [4] further highlights the significant burden DFUs place on healthcare systems and underscores the importance of automated, scalable solutions. Studies by Nguyen et al. [5] and Gupta et al. [6] emphasize that early detection is essential to minimize complications and improve patient outcomes. However, despite the progress, there is still a demand for not only accurate but also easy and reliable solutions for a wide range of settings. Additionally, Zhao et al. [7] discuss the socio-economic impacts of untreated DFUs, which further accentuate the importance of accessible automated solutions.

2.3 Datasets and Preprocessing Approaches

Datasets are essential for developing and testing models designed to detect diabetic foot ulcers (DFUs). Among the most widely used are the Diabetic Foot Ulcer Challenges (DFUC) datasets from 2020 and 2021. For example, Sarmun et al. [1] used over 2,000 images from the DFUC 2020 dataset, which contains a mix of ulcerative and nonulcerative cases. Similarly, Shata and Abdulaziz worked on a denoised subset of the DFUC dataset using MATLAB's denoising CNN to improve image quality, while Muthuraja et al. utilized the Kaggle dataset comprising 2,674 images.

Pre-treatment is an important step to clarify the input data for automated learning models. Common techniques include image resizing to standard resolutions, noise removal, and data augmentation to improve data quality and diversity. For example, Giridhar et al. [4] enhanced their dataset by applying noise removal, image scaling, and augmentation to improve variability and generalization. Muthuraja et al. [3] standardized the images to a resolution of 224x224 pixels to ensure compatibility with models such as EfficientNet and ResNet. Other researchers, such as Ali et al. [8], employed advanced contrast enhancement techniques to improve model performance, while Hassan et al. [9] focused on removing artifacts to reduce noise during training.

2.4 Machine Learning Models and Performance Metrics

Deep learning models have become the first choice for diabetic foot ulcer (DFU) detection due to their ability to analyze complex patterns in medical images. Convolutional Neural Networks (CNNs) are particularly popular, with architectures such as EfficientNet, ResNet, VGG-19, and DenseNet often used. Sarmun et al. [1] explored various object detection networks to achieve precise DFU identification. Muthuraja et al. [3] demonstrated an impressive result using EfficientNet, achieving 94.2% accuracy. Similarly, Giridhar et al. [4] highlighted the effectiveness of DenseNet201, achieving high F1 scores and accuracy across multiple datasets. Feature extraction plays a key role in these models, and transfer learning techniques are often used. For example, Muthuraja et al. [3] utilized pre-trained weights from EfficientNet and ResNet for DFU detection.

Other researchers, such as Nguyen et al. [5] and Zhao et al. [7], explored ensemble methods, combining multiple CNN architectures to enhance the reliability of predictions. Khan et al. [10] integrated attention mechanisms to improve localization accuracy for DFUs in medical images. Recent studies, such as Gupta et al. [6] and Hassan et al. [9], emphasized hybrid techniques that blend traditional image processing with deep learning to reduce computational costs.

Model performance is usually evaluated using metrics such as accuracy, precision, recall, F1 score, and ROC-AUC. Shata and Abdulaziz [2] reported an accuracy of 82.4% and a sensitivity of 69.2% for a CNN-based model, reflecting a balance between true and false positives. Despite these achievements, challenges such as overfitting, limited generalization between datasets, and insufficient testing in real-world scenarios persist, as highlighted by Ali et al. [8] and Khan et al. [10].

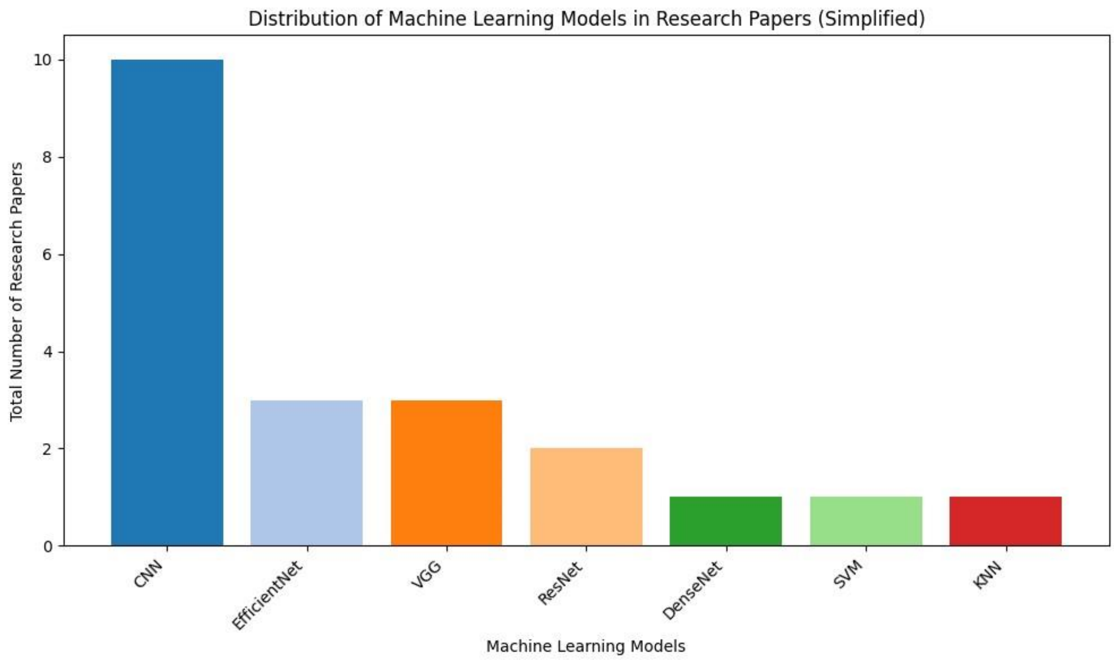

Figure 1:ML Model Usage in Research Papers: A Simplified Analysis

2.5 Summary

This chapter provides a detailed overview of current research on DFU detection, covering key areas such as problem formulation, datasets, preprocessing methods, machine learning models, and performance metrics. Although deep learning has shown promise for DFU detection, issues such as dataset variability, inconsistent preprocessing, and poor generalization to real-world contexts remain. These problems emphasize the need for innovative solutions focusing on the ability, reliability, and actual time. The results of this review are the basics of the proposed research, aiming to create an evolved and effective foundation for detecting DFUs.

Chapter 3

Methodology

3.1 Introduction

The methodology section of this report outlines the comprehensive approach employed to prepare, annotate, and preprocess datasets for the study. Two key datasets were integrated: high-quality clinical images from Bosch Pharma and a diverse collection from the Kaggle DFU dataset. Together, these datasets provided a robust foundation for training, with expert annotations ensuring reliability. Preprocessing techniques, including noise removal, image scaling, normalization, and data augmentation, were systematically applied to enhance image quality, address class imbalances, and standardize inputs. This meticulous process enabled the development of a deep learning model capable of accurate binary and multi-class classification of diabetic foot ulcers.

3.2 Data Collection

The data collection phase was pivotal in developing model. This section details the data sources used for binary and multi-class classification tasks, the challenges encountered during collection and curation, and the ethical considerations addressed to ensure data integrity and patient privacy. The datasets were carefully selected to support the project's objectives of building robust deep learning models for accurate DFU detection, with a focus on accessibility and scalability, particularly for underserved populations in Pakistan. The data collection strategy was tailored to enable both binary classification (ulcer vs. no ulcer) and multi-class, aligning with the project's aim to provide precise and actionable diagnostic insights.

3.2.1 Data Sources

The project utilized distinct datasets tailored to the requirements of binary and multiclass classification tasks, ensuring robust training of convolutional neural networks (CNNs) for diabetic foot ulcer (DFU) detection. The binary classification task, aimed at distinguishing between the presence and absence of ulcers, leveraged datasets from Bosch Pharma and the Kaggle DFU dataset. In contrast, the multi-class classification task, focused on categorizing ulcers by severity and treatability, relied exclusively on a curated dataset from Octdaily.

3.2.1.1 Binary Classification

For binary classification, the dataset comprised images from two sources: Bosch Pharma and the Kaggle Diabetic Foot Ulcer (DFU) dataset, totaling 2,206 images. The Bosch Pharma dataset provided 206 high-quality clinical images of diabetic feet, initially categorized by medical experts into three severity levels: severe, mild, and moderate. To simplify the initial detection task, these images were re-annotated into two classes “ulcer” (encompassing all severity levels) and “no ulcer” ensuring a clear distinction for model training. The Kaggle DFU dataset contributed 2,000 images, offering a diverse collection of foot images captured under varied conditions, such as different lighting, angles, and skin textures. This dataset was also annotated for binary classification, maintaining consistency with the Bosch Pharma data. The combined dataset provided a robust foundation for training models to detect the presence or absence of ulcers, The diversity and volume of these images enabled the model to generalize across different imaging scenarios, aligning with the project’s objective of early and accurate DFU detection.

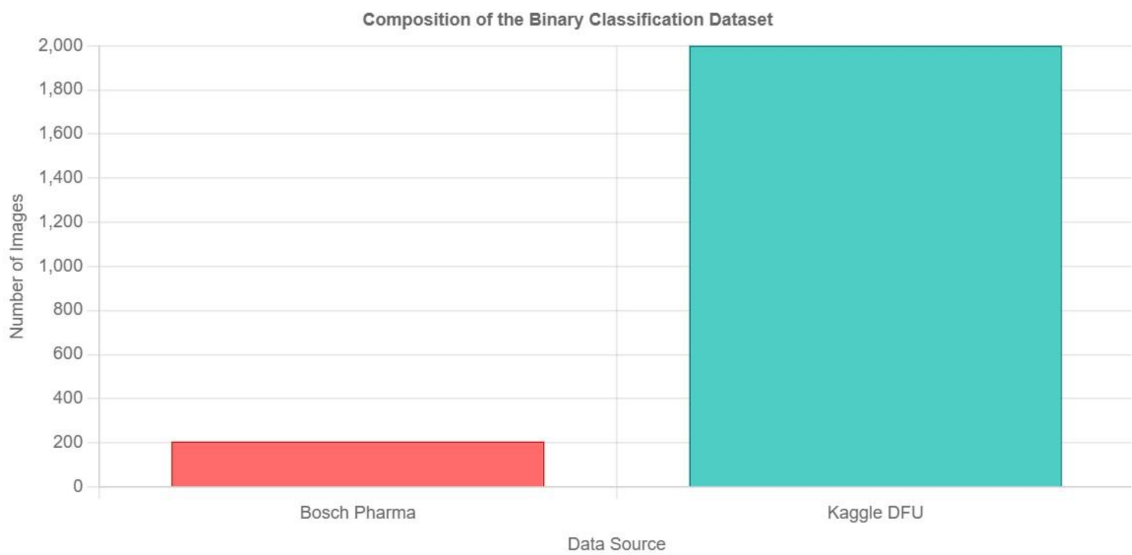

Figure 2: Composition of Binary Classification Dataset

3.2.1.2 Multi-class Classification

For multi-class classification, the project relied exclusively on a dataset provided by Octdaily, a medical data provider. The Octdaily dataset initially contained 18,000 raw images of various wound types from different body parts, classified into four categories: Complex wounds, Immediately treatable, No ulcer, and Treatable within 4 weeks. Due to the dataset's broad scope, which included non-foot wounds, a manual cleaning process was conducted in collaboration with medical experts to extract foot-specific images relevant to DFU detection. This process reduced the dataset to 4,916 images, distributed as follows: Complex wounds (225 images), Immediately treatable (1,827 images), No ulcer (924 images), and Treatable within 4 weeks (1,940 images), as visualized in Fig 9 (Class Distribution of the Octdaily Dataset). These images were annotated by medical professionals to ensure clinical accuracy, supporting the project's goal of classifying ulcers by severity and treatability for more nuanced diagnostic outcomes. The multi-class dataset was designed to enable the model to distinguish between different ulcer stages, enhancing its utility for both patients and healthcare providers in Pakistan's underserved regions.

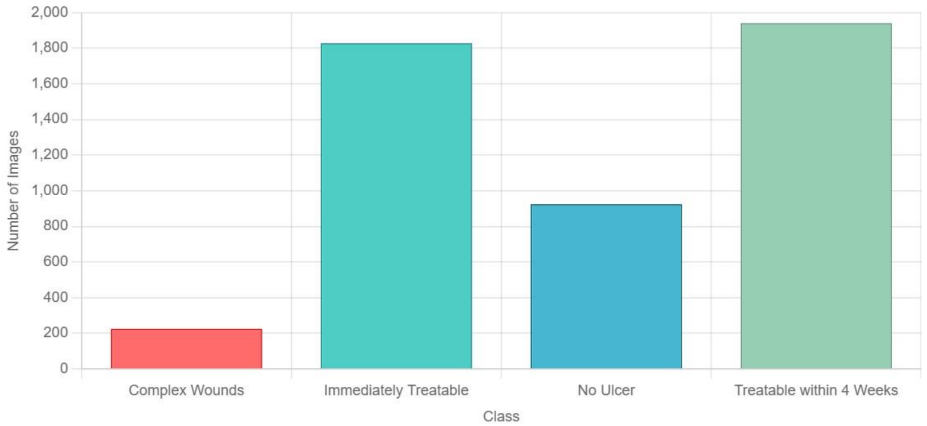

Figure 3:Composition of Binary Classification Dataset

Table 3 below summarizes the composition of the datasets used for each classification task, highlighting the sources, image counts, and annotation classes. The separation of datasets ensured that the binary classification model could focus on initial detection, while the multi-class model addressed detailed severity assessment.

Table 3: Overview of Dataset Composition for Binary and Multi-class Classification

Task	Source	Number of Images	Annotation Classes	Description
Binary Classification	Bosch Pharma	206	Ulcer, No Ulcer	High-quality clinical images re-annotated for binary classification.
	Kaggle DFU Dataset	2,000	Ulcer, No Ulcer	Publicly available dataset with diverse foot images, annotated for binary task.
	Total	2,206	Ulcer, No Ulcer	Combined dataset for detecting presence or absence of ulcers.
Multi-class Classification	Octdaily Dataset	6,247	Complex Wounds , Immediately Treatable , No Ulcer , Treatable within 4 Weeks	Filtered foot images from 18,000 raw wound images, categorized by severity and treatability.

3.2.2 Challenges in Data Collection

The data collection process presented significant challenges that required innovative solutions to ensure the datasets were suitable for training robust deep learning models. For the binary classification dataset, the Bosch Pharma images, while clinically validated, were limited in quantity (206 images), posing a risk of overfitting. The small dataset size necessitated the inclusion of the Kaggle DFU dataset to increase volume and diversity. However, the Kaggle dataset introduced challenges due to its heterogeneous nature, with variations in image quality, resolution, and lighting conditions. Harmonizing the Bosch Pharma and Kaggle datasets required extensive preprocessing, including resizing to a uniform 165×165-pixel resolution and normalization to ensure compatibility with convolutional neural networks (CNNs). The Octdaily dataset, used for multi-class classification, posed even greater challenges due to its raw and unrefined state. The initial 18,000 images included wounds from various body parts, such as arms and torsos, which were irrelevant to DFU detection.

Manual cleaning to isolate **6,247** foot-specific images was a labor-intensive and time consuming process, involving visual inspection by team members and validation by medical experts. This process was prone to human error, particularly in distinguishing foot wounds from other wound types with similar visual characteristics. Additionally, the dataset exhibited class imbalance, with categories like Complex wounds (225 images) significantly underrepresented compared to Treatable within 4 weeks (1,940 images). This imbalance risked biased model performance, necessitating data augmentation techniques, such as rotation and flipping.

The annotation process for both datasets also presented challenges. For the Bosch Pharma dataset, initial annotations for severity levels were inconsistent due to limited access to medical experts and variations in clinical interpretation, prompting reannotation for binary classification. Similarly, the Octdaily dataset required extensive collaboration with medical professionals to ensure accurate classification into the four categories, a process complicated by the subjective nature of assessing wound treatability. Despite these challenges, the curated datasets provided a solid foundation for model training, enabling the project to address diverse ulcer presentations and imaging conditions.

3.2.3 Ethical Considerations

Ethical considerations were a cornerstone of the data collection process, given the sensitive nature of medical images and their implications for patient privacy and clinical reliability. For the Bosch Pharma dataset, images were obtained with explicit permission from the provider, ensuring compliance with ethical standards for medical research. All images were de-identified to remove personally identifiable information (PII), such as patient names or medical record numbers, protecting patient confidentiality. The Kaggle DFU dataset, being publicly available, was verified to adhere to ethical guidelines, with images sourced from anonymized medical records, as documented in the dataset's public repository.

The Octdaily dataset required additional ethical scrutiny due to its raw and unprocessed nature. The team collaborated with Octdaily to confirm that the images were collected with informed consent from patients or their legal representatives, particularly for clinical images of visible wounds. During the manual cleaning process, any images containing identifiable features, such as faces or tattoos, were excluded to further safeguard patient privacy. The annotation process, conducted with medical experts, followed strict protocols to ensure clinical accuracy and avoid misclassification, which could lead to biased or harmful model predictions. This commitment to ethical data handling aligns with the project's dedication to SDG 3 (Good Health and Well-being) and SDG 10 (Reduced Inequalities).

To address potential dataset bias, the team prioritized diversity in the datasets, capturing a range of skin tones, ulcer severities, and demographic backgrounds. The Octdaily dataset's large initial size facilitated the inclusion of varied wound presentations, while the Kaggle dataset added diversity in imaging conditions.

3.3 Data Cleaning and Preparation

3.3.1 Data Annotation for Binary Classification

Accurate data labeling was a key aspect of this project, ensuring the dataset was well prepared for binary and multi-class classification tasks. While the Bosch Pharma dataset provided high-quality clinical images, its limited size prompted its integration with the Kaggle DFU dataset to create a more diverse and representative collection. This combined dataset was carefully annotated into four categories: severe, mild, moderate.

The annotation process was performed by expert physicians from Bosch Pharma, who used their expertise to accurately classify each image. They carefully assigned a label to each image using established clinical criteria such as the extent of tissue damage, the depth of the ulcer, the severity of the infection, etc. This systematic approach ensured that the annotations met medical standards and made the dataset a reliable foundation for training deep learning models. By combining datasets and using coherent marking for both sources, the project has reached a balance between diversity and accuracy. The contribution of the Bosch Pharma Medical team overcomes the gap between clinical knowledge and artificial intelligence, creating the basis for the reliable detection of ulcer and the gravity of the model.

Figure 4:Sample Images Categorizing Skin Conditions - (a) No Ulcer, (b) Ulcer

3.3.2 Data Annotation for Multi Classification

To ensure model accuracy and clinical relevance, several essential data cleaning and preparation steps were undertaken before training. This process involved manual filtering, precise class labeling, and final dataset consolidation to create a high-quality and task-specific image dataset.

3.3.2.1 Manual Filtering of Images

The initial dataset provided by OctDaily contained approximately 18,000 raw images of diabetic wounds from various body parts. Since the focus of our study is specifically on diabetic foot ulcers, a rigorous manual filtering process was carried out. With the assistance of medical professionals, particularly diabetic foot specialists, we carefully reviewed each image to isolate only those depicting the foot. This step was critical to eliminate irrelevant samples and reduce noise, thereby enhancing the accuracy and performance of the model.

3.3.2.2 Class Labeling

After filtering, the selected images were categorized into meaningful classes to facilitate both binary and multi-class classification tasks.

- **Binary Classification**: Images were labeled into two broad categories:
  - Diabetic Ulcer
  - No Ulcer

- **Multi-class Classification**: For more detailed classification, we assigned labels based on wound severity and treatment urgency, resulting in the following four classes:
    - **Immediately_Treatable** – 1,827 images
    - **No_Ulcer** – 924 images
    - **Complex_Wounds** – 1,556 images
    - **Treatable_Within_4_Weeks** – 1,940 images

These labels were annotated in consultation with medical professionals to ensure clinical accuracy and consistency across the dataset.

3.3.2.3 Final Dataset Composition

Following manual filtering and class labeling, the final dataset was composed of highquality, foot-specific images aligned with our research scope. The total number of selected and labeled images summed up to **6,247**, spanning four distinct classes for multi-class classification. This curated dataset served as the foundation for all subsequent model training, validation, and testing processes. It not only improved the relevance of the data but also played a key role in enhancing the model's ability to generalize and perform accurately in real-world clinical scenarios.

3.4 Data Preprocessing

Ensuring the quality and consistency of input data is critical for the success of machine learning models, particularly in medical image analysis where subtle visual details carry important diagnostic information. To prepare the dataset for training, validation, and testing, several preprocessing steps were undertaken to enhance image quality, normalize data, and address class imbalances.

3.4.1 Noise Removal

Noise in medical images can obscure vital features, thereby hindering the model's ability to detect and classify ulcers accurately. Common sources of noise in the dataset included background artifacts, scanning irregularities, and irrelevant objects. To mitigate these issues, noise reduction techniques such as Gaussian filtering and median filtering were applied. These filters effectively suppressed unwanted pixel-level noise while preserving important structural details, including skin texture and ulcer edges.

This step significantly improved image clarity and allowed the model to focus on critical features, enhancing pattern recognition during training.

3.4.2 Image Scaling

To maintain uniformity across the dataset, all images were resized to a fixed resolution of 456 × 456 pixels, which corresponds to the expected input size for the EfficientNetB5 architecture. Standardizing image dimensions is essential because convolutional neural networks (CNNs) require consistent input sizes to operate efficiently and effectively. The 456×456 resolution was specifically selected as it strikes a practical balance between computational efficiency and the preservation of critical visual features, particularly important in medical imaging tasks involving ulcer detection. This resizing ensured that key characteristics of the ulcers were retained with minimal distortion, enabling the model to learn effectively. Moreover, having a consistent input size allowed for streamlined batch processing and stable performance across all training, validation, and testing phases.

Figure 5:Image Resizing

3.4.3 Normalization

When training images on EfficientNetB5, normalization is an essential preprocessing step to ensure the input data aligns with the expectations of the model, which was pretrained on the ImageNet dataset. The standard approach involves resizing each image to 456×456 pixels (the input size required by EfficientNetB5) and converting the image data to a float32 format. The pixel values, which originally range from 0 to

255, are then passed through the preprocess_input function from Keras' EfficientNet module. This function scales the pixel values to a range of approximately **[-1, 1]** by applying the transformation: x = (x / 127.5) - 1.
This normalization helps the model converge faster during training and ensures that the input distribution is similar to what the network saw during its original pretraining on ImageNet. Without this step, the model's performance would typically degrade due to mismatched input distributions.

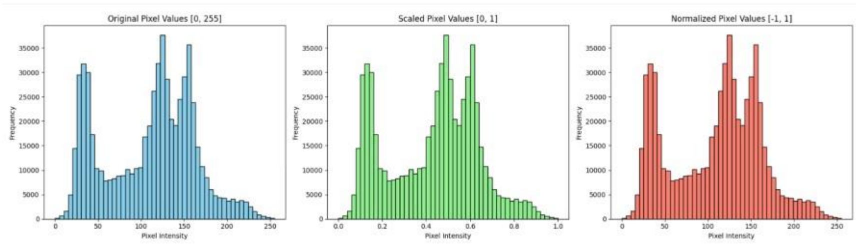

Figure 6:Pixel intensity distributions before and after preprocessing

3.4.4 Data Augmentation

Class imbalance posed a significant challenge, particularly due to the underrepresentation of complex cases such as severe ulcers. To address this issue, targeted data augmentation techniques were applied exclusively to the minority class to artificially expand its size and improve class diversity. These techniques included:

- Rotation: Simulated different viewing angles by rotating the images.
- Flipping: Applied both horizontal and vertical flips to enhance visual variability.
- Scaling: Introduced spatial variation by resizing images at different scales.
- Brightness Adjustment: Modified lighting conditions to reflect real-world scenarios.
- Random Resizing: Created size variations while preserving key features of the ulcers.

Through these augmentation strategies, the number of samples in the complex (severe) class increased from 255 to 1,556, resulting in a more balanced dataset. This improvement not only mitigated the class imbalance but also enhanced the model's robustness and generalization performance when exposed to unseen data.

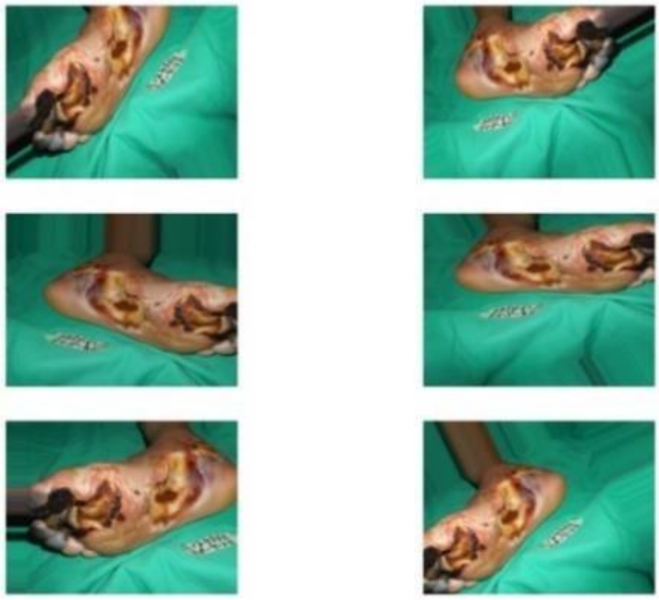

Figure 7:data augmentation techniques applied to input images

3.4.5 Gaussian Blur Application

To enhance the quality of the images and reduce high-frequency noise that could negatively impact model training, Gaussian blur was applied as part of the preprocessing pipeline. This technique smooths the images by averaging pixel intensities with their neighbors using a Gaussian kernel, which helps to reduce finegrained noise and subtle artifacts commonly present in medical images. The controlled blurring preserved important structural features of the ulcers while minimizing distractions caused by noise, thereby improving the robustness of the model. Applying Gaussian blur contributed to more stable feature extraction and helped the network focus on the salient ulcer characteristics during training.

Figure : Gaussian blur applied to an image

3.5 Model Development

3.5.1 Architecture

The chosen model architecture for this study is EfficientNetB5, a convolutional neural network designed to deliver high accuracy while maintaining computational efficiency. EfficientNet models utilize a novel compound scaling method, which simultaneously scales the network's depth (number of layers), width (number of channels per layer), and input image resolution in a balanced manner. This contrasts with traditional CNN scaling approaches that scale these dimensions arbitrarily, often leading to inefficient use of resources.

EfficientNetB5, being one of the larger variants in the EfficientNet family, takes advantage of this scaling by increasing model capacity and input resolution (456×456 pixels), allowing it to capture more detailed spatial features relevant to complex image classification tasks such as diabetic foot ulcer severity grading. Its architecture is based on mobile inverted bottleneck convolution blocks (MBConv), which help reduce the number of parameters and computational cost without sacrificing representational power.

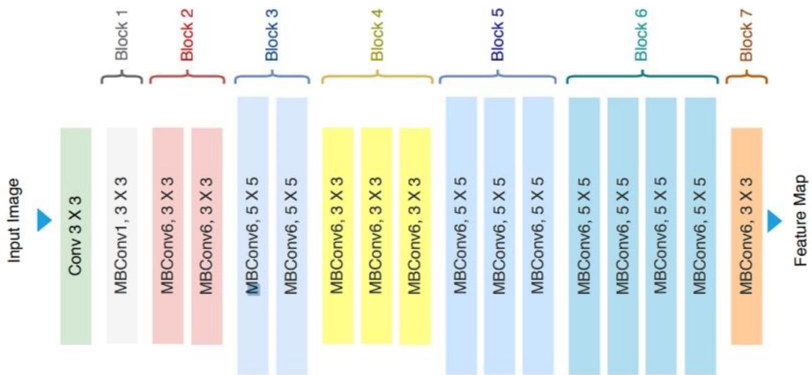

Figure 8:EfficientNetB5 architecture

3.5.2 Justification of Model Choice

Research has shown that EfficientNet architectures consistently outperform other CNNs such as ResNet, DenseNet, and Inception on various benchmark datasets, including medical imaging domains. For example, studies applying EfficientNet to ulcer classification and wound assessment have reported higher accuracy and better generalization, attributed to the model's ability to efficiently learn multi-scale features due to its compound scaling strategy.

Moreover, EfficientNetB5's pretraining on the large ImageNet dataset provides a strong transfer learning foundation, enabling the model to leverage learned low- and mid-level features that are transferable to medical images, despite limited annotated data. This reduces the need for extensive training data and improves convergence speed. Overall, EfficientNetB5's balance of accuracy, efficiency, and scalability makes it an ideal choice for this ulcer classification task.

3.5.3 Model Training

Model training is a crucial phase in developing an accurate and reliable classification system. This section describes the data splitting strategy, hyperparameter choices, and the computational environment used to train the EfficientNetB5 model on the diabetic foot ulcer dataset.

3.5.3.1 Training-Validation-Test Split

The dataset was divided into three mutually exclusive subsets: training, validation, and testing, with proportions of 80%, 10%, and 10%, respectively. This division maximizes the amount of data available for learning while preserving enough samples to evaluate the model's performance reliably.

- **Training set (80%)**: Used to optimize the model parameters by iteratively learning from the labeled images.
- **Validation set (10%)**: Utilized during training to monitor the model's performance on unseen data and to guide hyperparameter tuning, helping prevent overfitting.
- **Test set (10%)**: Held out entirely during training, this subset provides an unbiased evaluation of the final model's ability to generalize to new, unseen data.

Figure 9: Splitting of Dataset

The splitting was done using stratified sampling to maintain the original class distribution across all subsets, ensuring that each set represents the full diversity of the dataset.

3.5.3.2 Hyperparameters and Configuration

The model was configured and trained using the following hyperparameters:

- **Architecture:** EfficientNetB5, pre-trained on ImageNet, and fine-tuned on the diabetic foot ulcer dataset.
- **Input size:** 456 × 456 pixels, conforming to EfficientNetB5's requirements.
- **Batch size:** 32 images per training batch, balancing memory constraints and training efficiency.
- **Epochs:** 50, providing sufficient iterations for the model to learn patterns without overfitting.
- **Optimizer:** Adam optimizer with an initial learning rate of 0.0001, selected for its adaptive gradient capabilities.
- **Loss function:** Categorical cross-entropy, suitable for multi-class classification.
- **Early stopping:** Enabled with a patience of 5 epochs to halt training if validation loss does not improve, preventing overfitting.

- **Learning rate scheduler:** Learning rate reduced by a factor of 0.1 when validation loss plateaued for 3 consecutive epochs, allowing fine-tuning of weights.
- **Data shuffling:** Performed each epoch to ensure randomized batch composition and reduce bias.

These parameters were selected based on previous studies and empirical testing to optimize model performance on the diabetic foot ulcer classification task.

3.5.3.3 Training Environment

Training was performed using the Kaggle cloud platform, leveraging its GPUaccelerated computing resources for efficient model development:

- **Hardware:** NVIDIA Tesla P100 GPU with 16 GB VRAM, providing the computational power necessary for training deep neural networks on highresolution images.
- **Software:** Python 3.8 environment with TensorFlow 2.x and Keras deep learning frameworks.
- **Operating system:** Linux-based system managed by Kaggle.
- **GPU acceleration:** CUDA and cuDNN libraries were pre-installed and optimized for GPU use on the Kaggle platform.
- **Training time:** Approximately 6 hours for 50 epochs, depending on batch size and dataset size.

This environment enabled rapid prototyping and experimentation without requiring local hardware, streamlining the research workflow.

3.5.4 Model Evaluation

3.5.4.1 Performance Metrics (Accuracy, Precision, Recall, F1-score)

To evaluate the EfficientNetB5 model's classification performance, we utilized key metrics including accuracy, precision, recall, and F1-score. These metrics help assess how well the model distinguishes between the four medical categories. The evaluation was performed on the test set using the classification_report from scikit-learn. The results provide both a macro and class-wise view of model effectiveness.

```
Class: Complex_Wounds
  TP: 111, FP: 21, FN: 45, TN: 449

Class: Immediately_Treatable
  TP: 131, FP: 57, FN: 52, TN: 386

Class: Treatable_Within_4_Weeks
  TP: 133, FP: 80, FN: 61, TN: 352

Class: no_ulcer
  TP: 93, FP: 0, FN: 0, TN: 533

Classification Report:

                          precision    recall  f1-score   support

          Complex_Wounds       0.84      0.71      0.77       156
   Immediately_Treatable       0.70      0.72      0.71       183
Treatable_Within_4_Weeks       0.62      0.69      0.65       194
                no_ulcer       1.00      1.00      1.00        93

                accuracy                           0.75       626
               macro avg       0.79      0.78      0.78       626
            weighted avg       0.76      0.75      0.75       626
```

Figure 10: Performance Metrices

3.5.4.2 Confusion Matrix

A confusion matrix was generated to visualize the performance of the classifier in terms of true and false predictions across all four classes.

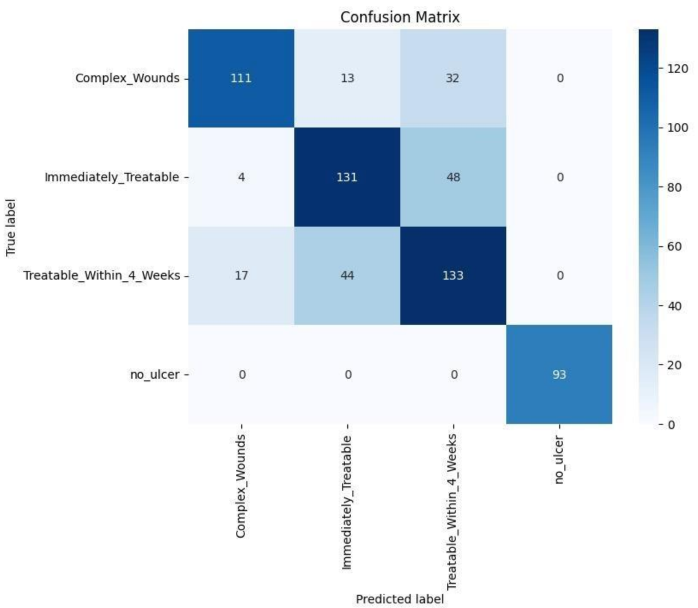

Figure 11: Confusion Matrix

3.5.4.3 Class-wise Evaluation

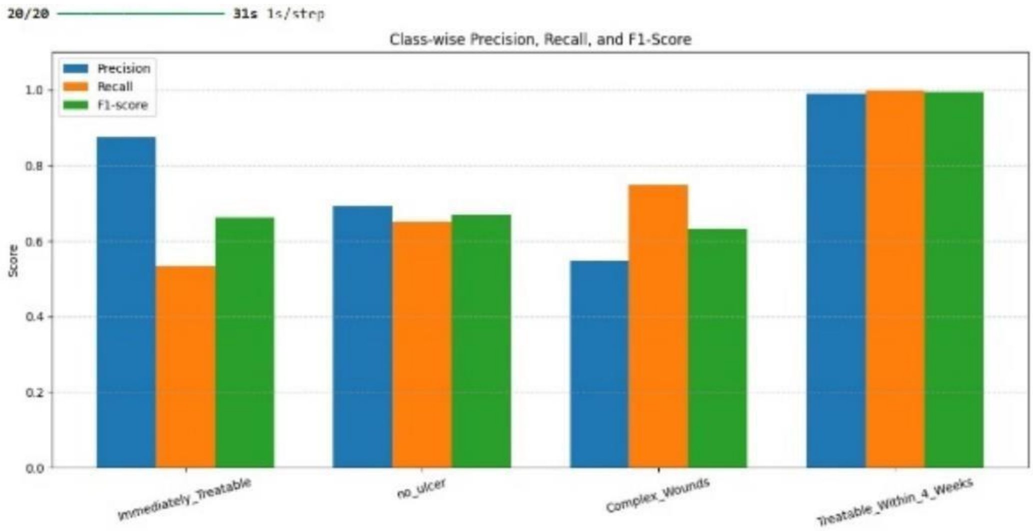

Figure 12: Class Wise Evaluation

3.5.4.4 Example Test Predictions

To supplement the quantitative analysis, we include sample test images with their true and predicted labels.

Figure 13: Test Predictions for Model Evaluation

Figure 14: Prediction Results

3.6 Fine Tuning of Model

This report documents the fine-tuning process of a pre-trained EfficientNetB5 model, initially achieving 74% accuracy, on the Diabetic Foot Ulcer (DFU) dataset. The objective was to enhance the model's performance for classifying foot ulcer images into four categories: Immediately Treatable, No Ulcer, Complex Wounds, and Treatable Within 4 Weeks. The fine-tuning process involved adjusting the model's architecture, optimizing hyperparameters, and applying data augmentation techniques to improve generalization.

3.6.1 Model Configuration

The pre-trained EfficientNetB5 base model was made partially trainable by unfreezing the last 30 layers, while earlier layers remained frozen to retain learned features. The model was recompiled with the Adam optimizer at a learning rate of 1e-5, using categorical cross-entropy as the loss function and accuracy as the evaluation metric.

3.6.2 Class Weighting

To address class imbalance, class weights were computed using the compute_class_weight function from scikit-learn. The resulting weights were:

- Immediately Treatable: 0.68
- No Ulcer: 1.34
- Complex Wounds: 0.80
- Treatable Within 4 Weeks: 0.64

These weights were applied during training to penalize misclassifications of underrepresented classes.

3.6.3 Training Setup

The model was fine-tuned for 30 epochs with a batch size of 16. Mixed precision training was enabled to optimize computational efficiency. The following callbacks were used:

- **ModelCheckpoint**: Saved the best model weights based on validation accuracy to /kaggle/working/best_model_finetuned.h5.
- **EarlyStopping**: Stopped training if validation accuracy did not improve for 10 epochs, restoring the best weights.
- **ReduceLROnPlateau**: Reduced the learning rate by a factor of 0.5 if validation accuracy plateaued for 5 epochs, with a minimum learning rate of 1e-6.

3.6.4 Results

The fine-tuned model was evaluated on the test set, achieving a final accuracy of 82.2%

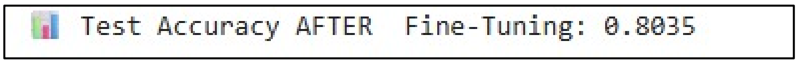

Figure 15: Accuracy After Fine Tuning

Confusion matrix shows:

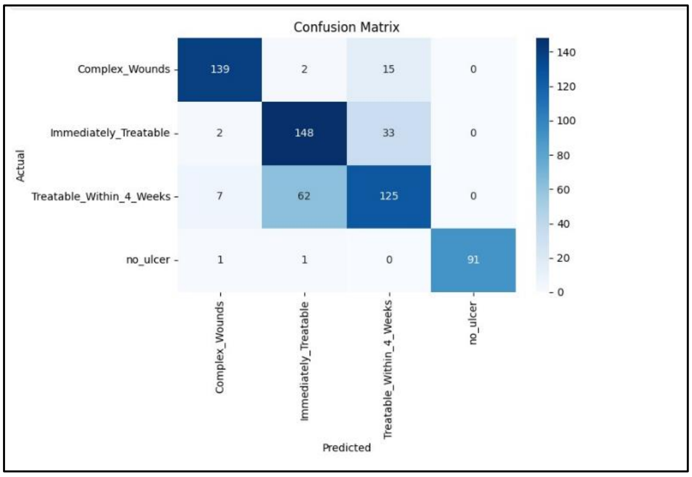

Figure 16:Training and validation accuracy (left) and loss (right) over epochs.

The fine-tuning process improved the model's performance by leveraging transfer learning and targeted unfreezing of layers. Data augmentation and class weighting mitigated overfitting and class imbalance, respectively. The use of callbacks ensured optimal model selection and prevented overtraining.

The EfficientNetB5 model was successfully fine-tuned on the DFU dataset, achieving a test accuracy of 80.35%. The fine-tuned model was saved to for future deployment.

Chapter 4

Mobile & Website Development

4.1 Website Development Overview

The website for the Diabetic Foot Ulcer Detection Project was developed using the MERN stack, which includes MongoDB, Express.js, React.js, and Node.js. This combination was chosen for its ability to support scalable, fast, and reliable web applications, making it a solid foundation for a healthcare-focused platform that may need to grow and adapt over time. It also reflects a shift away from more traditional web development methods, offering greater flexibility and performance.

4.1.1 UI/UX Design on Figma

The user interface and experience for the Diabetic Foot Ulcer Detection Project were designed using Figma, with a strong focus on simplicity, clarity, and accessibility. The goal was to build an interface that would feel approachable for all users, particularly patients who may not be familiar with digital platforms as well as efficient for healthcare professionals.Once finalized, the design was shared as an interactive prototype, which played a key role in guiding the front-end development team during the implementation phase. It helped ensure that the visual and functional intentions behind the design were carried through into the final product.

4.1.1.1 Registration Page

This is a simple and clean sign-up page for a platform that helps detect ulcers. User can create an account by entering your details or just sign up quickly with your Google account.

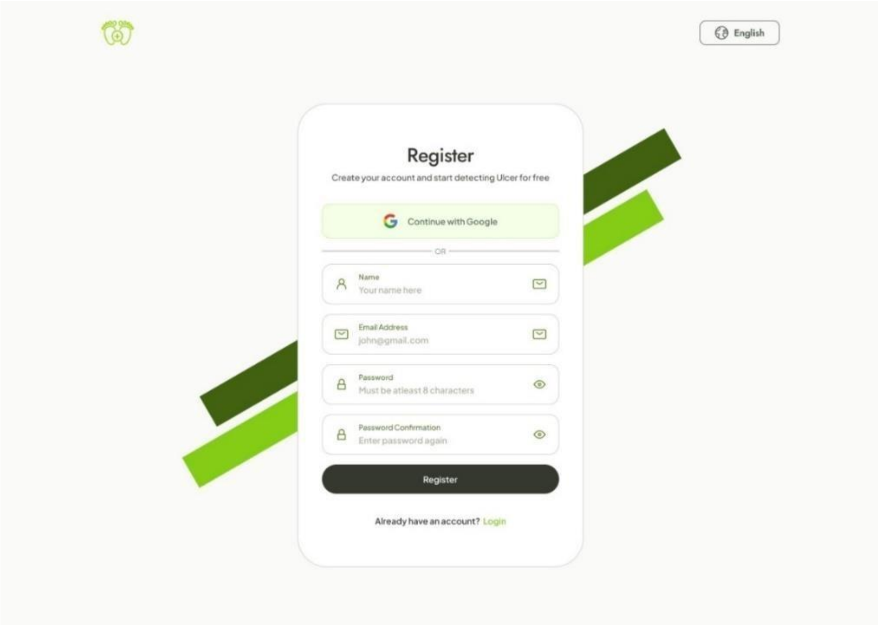

Figure 17: Registration Page Ui

4.1.1.1 Login Page:

This login page provides users with a straightforward way to sign in using either their Google account or by entering their email and password.

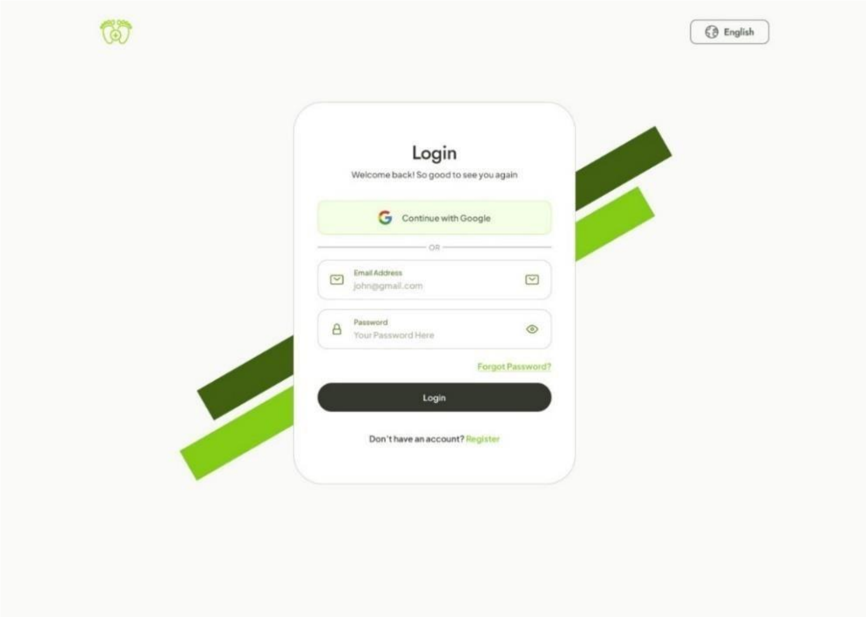

Figure 18: Login Page

4.1.1.3 View Detection History

This page displays recent ulcer detection results, including the date, condition (such as Mild, Severe, or No Ulcer), and the submitted image.

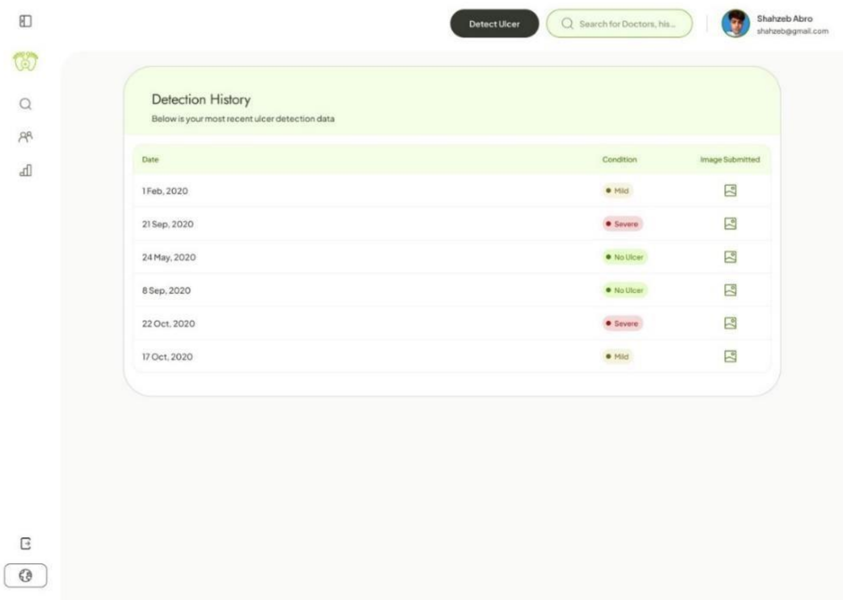

Figure 19: Detection History

4.1.1.4 Telemedicine Support

This page lists top doctors in Pakistan who specialize in ulcer treatment. Each entry includes the doctor's name, photo, and email for easy contact.

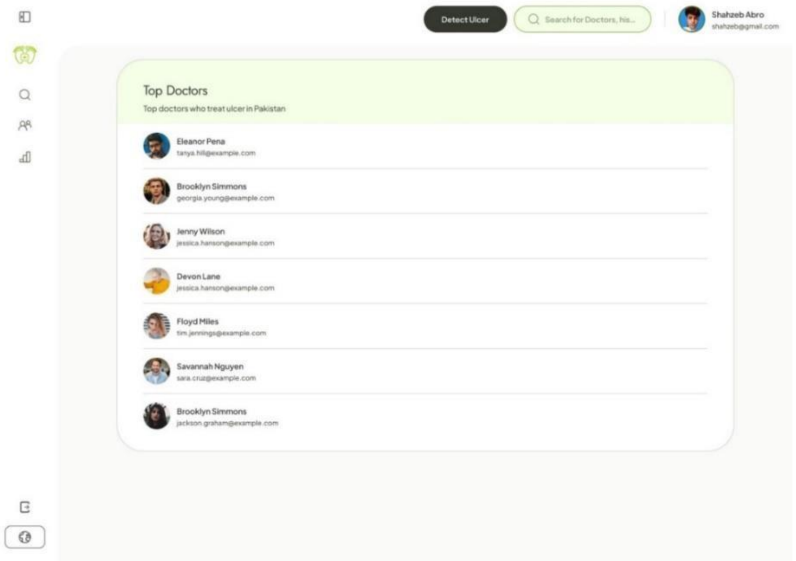

Figure 20: View Top Doctors List

4.1.1.5 Detection Page

The detection feature uses an AI algorithm to analyze submitted images and identify the presence and severity of ulcers. It provides results instantly, labeling conditions as "Mild," "Severe," or "No Ulcer."

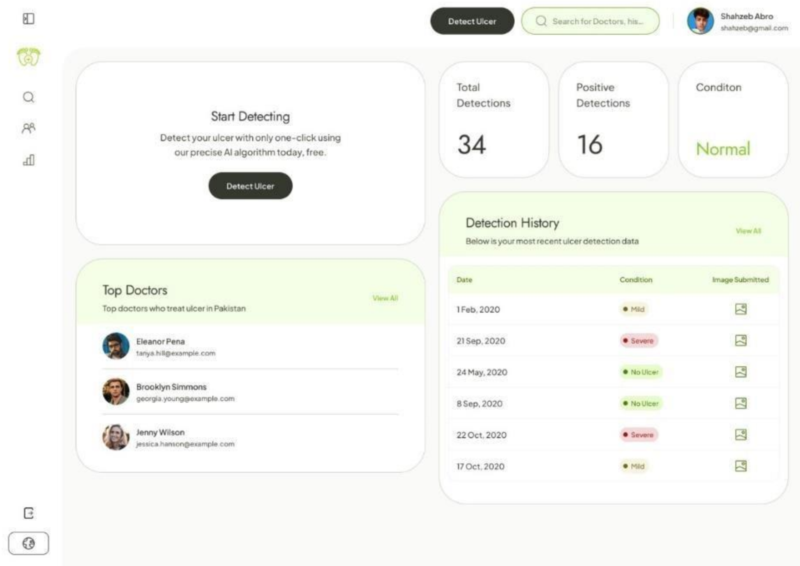

Figure 21: Detection Page

4.1.2 Implementation

The website integrates an AI-powered backend with a responsive frontend for seamless ulcer detection and result display.

4.1.2.1 Backend Implementation

The backend for the authentication system was developed using Node.js and Express.js, with a focus on both security and usability.

- **User Data Management with MongoDB:** A detailed schema was designed to store user information, including the username, email, encrypted password, and an OAuth token for users opting to log in with Google. MongoDB's flexibility made it ideal for managing this structured data securely and efficiently.

- **Password Encryption**: To protect user credentials, passwords are hashed using the bcrypt library. This means that even if someone were to gain unauthorized access to the database, the passwords would still be protected and unreadable.

- **Google OAuth 2.0 Integration**: The system also allows users to log in with their Google accounts, providing a fast and familiar alternative to traditional login. This feature is powered by passport.js with Google Strategy, simplifying authentication through OAuth 2.0 while maintaining strong security standards.

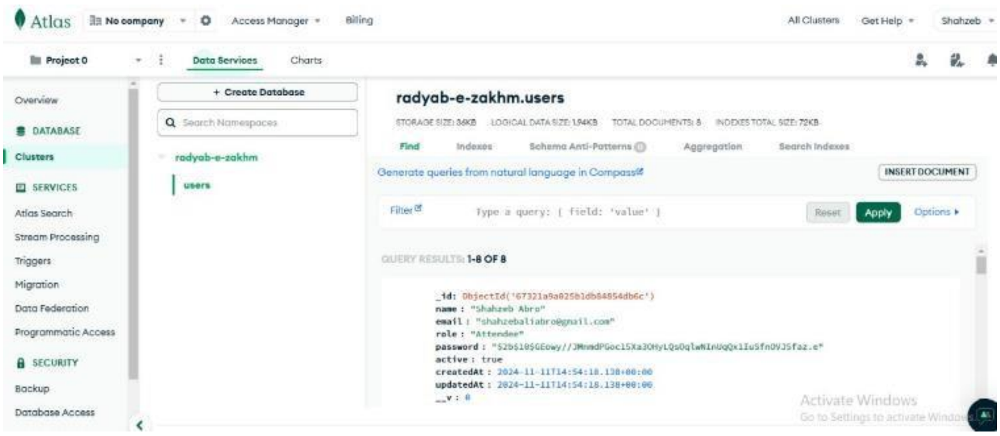

Figure 22:Database in MongoDB

4.1.2.2 Frontend Implementation

The front end of the website was developed using React.js, with a focus on modular design, responsiveness, and smooth user experience. The interface was designed to be intuitive for healthcare professionals, allowing easy access to all key features. The system brings together a variety of technical and practical features to offer users a smooth and effective experience for early foot ulcer detection and medical consultation. Here's an overview of the key features that have been built into platform:

• Authentication System

The authentication system is a key part of the website, developed to protect sensitive medical information and control access to the platform. It handles user registration and login, ensuring that only authorized users—such as healthcare professionals—can interact with the system. To make the login process smoother, especially for users who prefer convenience, the system also includes Google Sign-In as an option. This allows users to log in using their Google accounts without needing to create a separate username and password. The following section explains how the authentication system works on the backend.

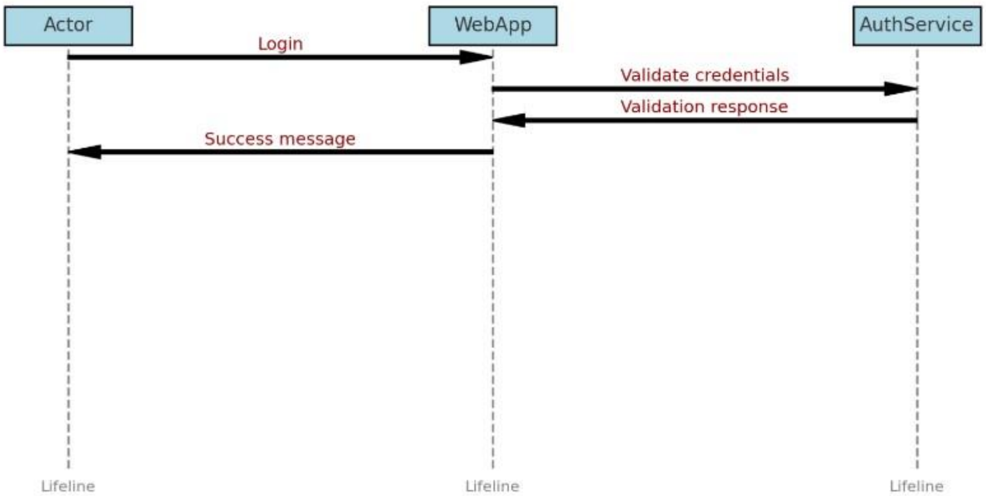

Figure 23:Sequence Diagram of Authentication

- **Google Sign-In Integration:** To improve accessibility and streamline the login process, Google OAuth was integrated using the react-google-login package, allowing users to log in with just one click.

- **Ulcer Detection Interface:** A built-in feature makes it easy for users to upload images of their feet directly through the platform. Once uploaded, the backend returns detection results from the AI model, which are displayed directly on the page.

- **Detection History View:** Users can review their past ulcer scans, displayed in an easy-to-read format using tables or cards. Each entry includes the date, time, and result for clear tracking.

- **Consultation Module:** A modular component was built to let users browse available doctors and request consultations. This feature is reusable and cleanly integrated within the broader interface.

• **Bilingual Urdu-English Feature**

To make the platform accessible to a wider audience, especially patients and healthcare workers in Pakistan, a bilingual feature was added to the website. Users can seamlessly switch between English and Urdu throughout the platform, ensuring ease of use for both language preferences. All key modules including authentication, ulcer detection, consultation, and history are fully localized. This feature enhances user experience, promotes inclusivity, and improves understanding for non-English speakers.

Figure 24:Bilingual Feature

4.1.2.3 Model Integration

The trained model for detecting foot ulcers was made accessible online by deploying it through Flask, allowing it to handle real-time predictions and interact with the website. It communicates with the frontend through RESTful API calls, enabling real-time prediction results whenever an image is uploaded.

Uploaded images are sent from the frontend to the Flask server, where they are processed by the model. The server then returns a structured response with prediction details and confidence levels.

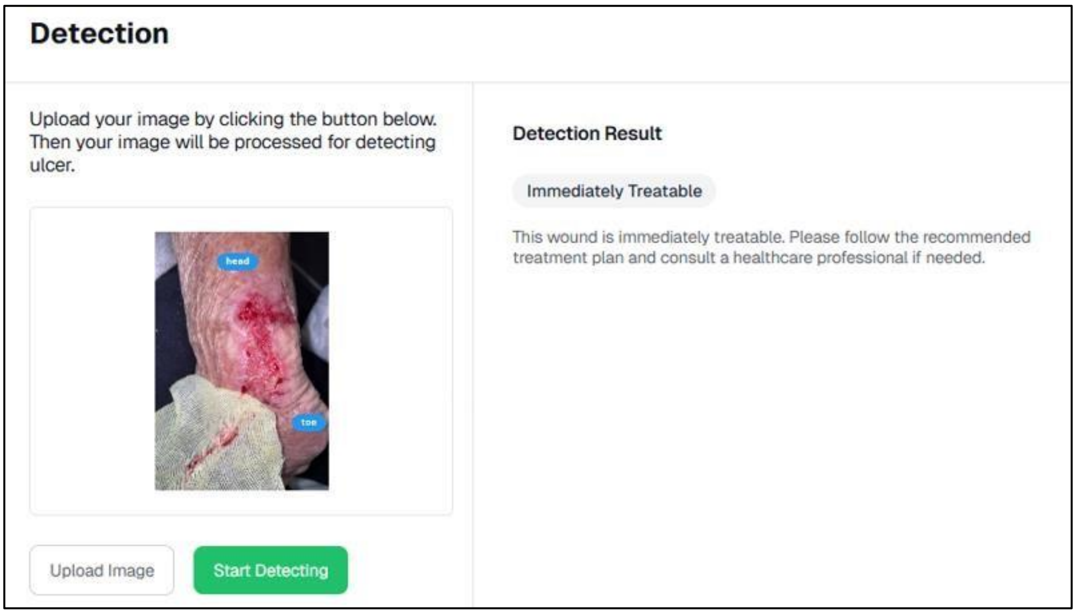

Figure 25: Model Results

4.1.2.4 Testing and Debugging

- **Unit Testing:** Individual frontend components, especially the login and registration forms, were tested using Jest and the React Testing Library to ensure they behaved as expected.
- **API Testing:** Postman was used to test various API endpoints, particularly those related to user registration and login, to confirm they returned correct responses and handled errors gracefully.
- **Cross-Browser Testing:** The site was tested across major web browsers like Chrome, Firefox, and Edge to ensure consistency in layout and functionality.

4.1.2.5 Deployment:

• **Hosted on Hostinger VPS**

The entire application was deployed on a Hostinger Virtual Private Server, giving complete control over both backend and frontend environments for better reliability and scalability.

• **Nginx and Gunicorn Configuration**

Nginx was set up as a reverse proxy to serve the React frontend and route backend requests to the Flask server, which is managed by Gunicorn for efficient processing.

• **Secured with HTTPS**

SSL certificates from Let’s Encrypt were installed, ensuring that all data transferred during user login, registration, and image uploads is encrypted and secure.

• **Custom Domain & Service Management**

A personalized domain name was set up by adjusting the DNS settings through Hostinger, giving the website a professional and easily recognizable web address. Process management was handled through Supervisor, which ensures that the backend services automatically restart if any issues arise.

4.2 Mobile Application Development

The Diabetic Foot Ulcer Detection mobile app was designed with accessibility and ease of use in mind, making it convenient for both patients and healthcare professionals. Built using React Native for the frontend, and powered by Node.js and Express.js on the backend, the app delivers a smooth and responsive experience across both Android and iOS devices. MongoDB, integrated with Mongoose, handles data storage efficiently, supporting the app's scalability as user demand grows.

The core features allow users to upload a photo of their foot, which is then processed by an AI model to detect signs of a diabetic foot ulcer. The process is simple and userfriendly, requiring minimal input from the user while delivering quick and accurate results. Beyond detection, the app offers additional support tools, such as personalized doctor recommendations based on the scan results. It also includes a dedicated history section where users can view past scans, results, and track their health progress over time. This section explores how the mobile app evolved throughout development, with a focus on its main features and the design choices that prioritize a smooth and supportive user experience.

4.2.1 UI/UX Design on Figma

The design prioritizes user ease with intuitive, touch-optimized, and visually consistent elements.

- **Home Dashboard:** The dashboard is designed to give users quick access to key features such as ulcer detection, health updates, and doctor suggestions right from the home screen. This reduces unnecessary navigation and helps users stay focused on what matters most.
- **Optimized for Touch Devices:** Every interactive element buttons, forms, and menus has been designed with mobile usage in mind. Larger tap areas and responsive feedback ensure the app feels natural and smooth to use on any smartphone.
- **Visual Consistency:** The layout uses consistent typography, color palettes, and icons, making it easy for users to understand what they're seeing briefly. Important health-related details are highlighted for quick recognition.

- **Simplified Input:** Options like Google Sign-In and image-based ulcer detection streamline interaction and reduce user effort.

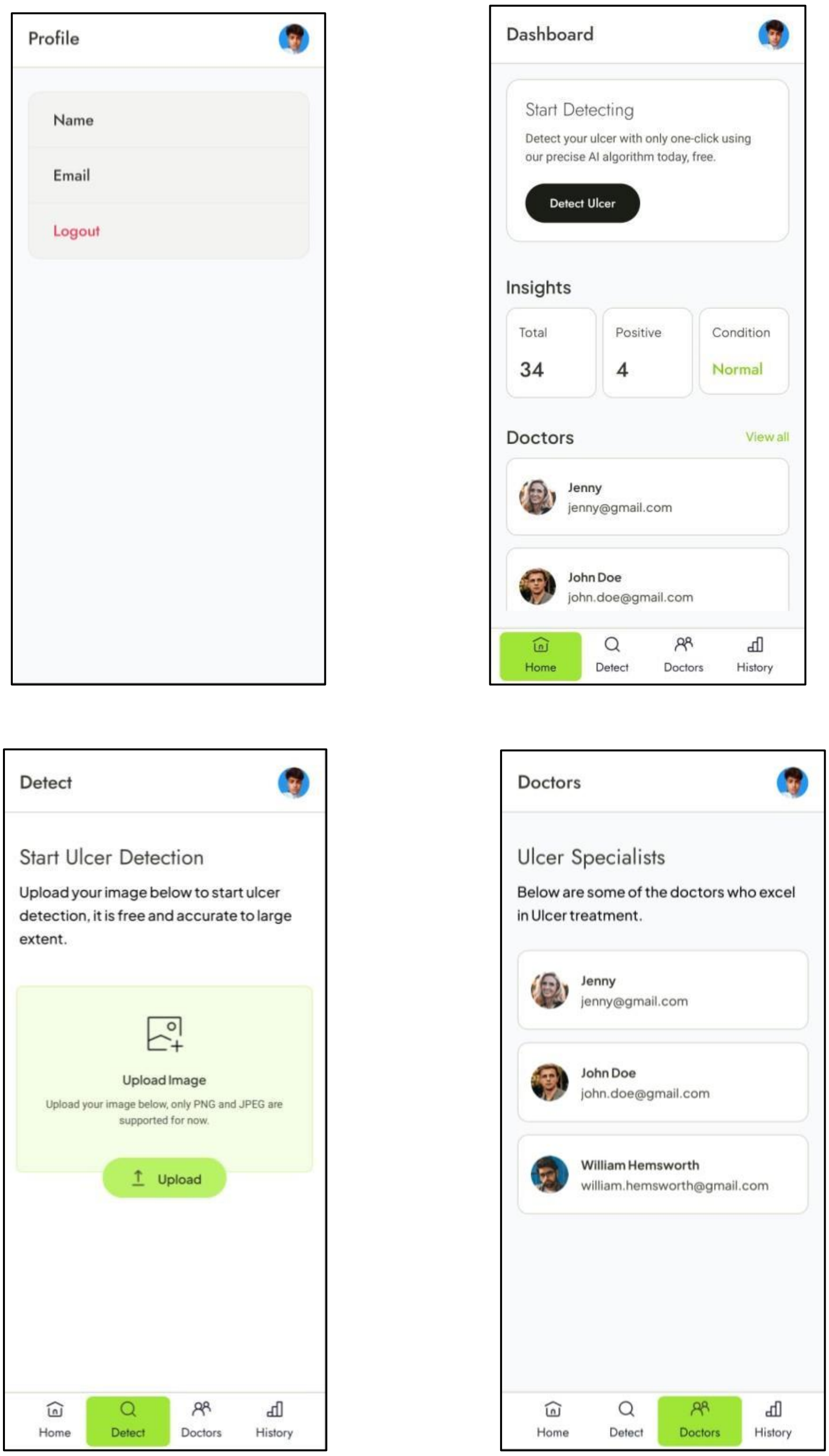

Figure 26: Mobile App UI UX

4.2.2 Mobile Application Workflow

The mobile application follows a structured workflow starting from user registration to ulcer detection and result tracking. Each stage ensures a seamless user experience through dedicated pages and intuitive navigation.

- **User Registration/Login**
  The user begins by signing up or logging into the application using their credentials.
- **Redirect to Main Dashboard**
  Once the login or registration is successful, the user is redirected to the man dashboard (home screen). This screen includes a section labeled "Start Detecting" for initiating ulcer detection and presents insights into their previous records.
- **Dashboard Features**
  The dashboard offers an intuitive interface with several key components: a "Detect Ulcer" button for starting the detection process, insight cards showing total detections, number of positive results, and current condition, a short list of doctors with their names and emails, and a history preview summarizing the user's detection activity over time.
- **Ulcer Detection Process:** When the user taps on the "Detect Ulcer" button, they are redirected to a separate detection page. Here, they can upload an image for analysis. The AI model processes the image and detects whether an ulcer is present. If the condition is found to be moderate or severe, the user is offered an option to connect with available doctors. In cases where the condition is normal or mild, the result is saved directly to the history for future reference.
- **Doctors Page**
  If the user clicks on a doctor's name or the "View All" option from the dashboard, they are taken to the dedicated Doctors page. This page displays a list of all available doctors along with their contact emails for easy communication and follow-up in case of serious diagnoses.
- **History Page**
  By clicking on the insight cards or selecting the "History" tab from the bottom navigation, the user is taken to the History page. This section contains a detailed

record of all previous ulcer detections, including the date, results, and severity levels for reference and ongoing tracking.

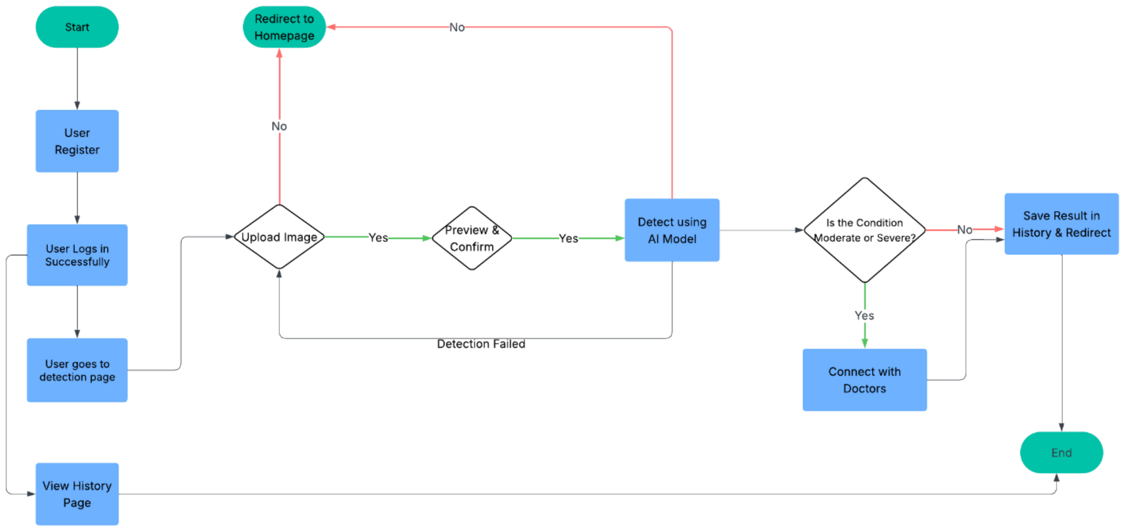

Figure 27: Workflow Diagram

4.2.3 Implementation

The mobile application is developed with a focus on seamless integration between the frontend and backend, ensuring a smooth user experience and accurate ulcer detection using AI.

4.2.3.1 Backend Implementation

The backend of the mobile application was developed using Node.js and Express.js to ensure a reliable, secure, and scalable server environment. MongoDB, integrated with Mongoose, powers the database layer, offering flexibility and speed in managing user data and detection results. JWT (JSON Web Tokens) is used to handle user authentication, ensuring data security and protecting sensitive operations.

- **Built with Node.js and Express.js:** The backend was designed to be fast and efficient, making sure everything runs smoothly behind the scenes and connects well with the user-facing side of the app.

- **MongoDB & Mongoose:** Offers a flexible NoSQL solution for storing user records and scanning histories, with Mongoose managing schemas and database operations.
- **JWT Authentication:** Enhances security by issuing tokens upon login, safeguarding user sessions, and restricting unauthorized access.
- **RESTful APIs:** Endpoints are structured to handle user registration, login, image uploads, fetching detection results, and retrieving past scans efficiently.

4.2.3.2 Frontend Implementation:

The frontend was developed using React Native, allowing the app to run smoothly across various mobile devices while offering a native-like user experience. The design is modular, responsive, and built with reusability in mind, making the app both scalable and user-friendly. The following section outlines the key features that shape the functionality and user experience of the mobile application. These include both technical implementations and user-focused functionalities that make the app effective, responsive, and easy to use.

- **Easy to Use:** Whether someone's using a phone, tablet, or laptop, the app adjusts itself to fit just right keeping things simple, clear, and easy to navigate no matter what the device.
- **Navigation System:** A smooth and intuitive navigation system, built using React Navigation, helps users effortlessly move between key sections like Home, Detection, Doctors, and History.
- **Image Upload Feature**: Allows users to take or select foot images, compresses them efficiently, and sends them securely to the server.
- **Reusable Components:** Encourages maintainable code by breaking the interface into modular, reusable parts.

- **Login/Registration:** Simple and secure access to user profiles, backed by JWTbased sessions.

- **Image Upload:** Quick image capture/upload to initiate ulcer detection with minimal steps.

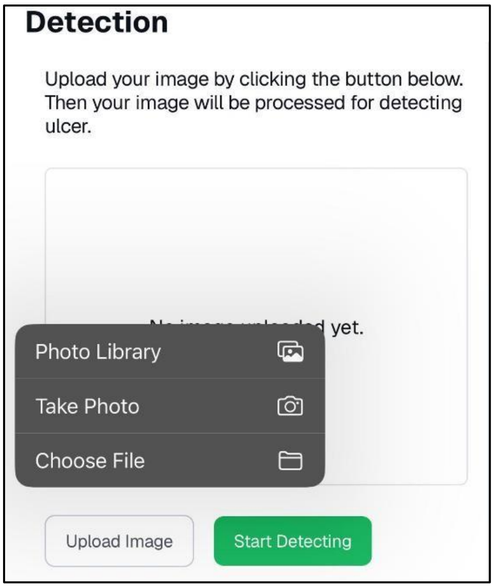

Figure 28: Image Upload on App

4.2.4 Model Integration

To enable intelligent detection capabilities, the app communicates with a Flask server that hosts the AI model. This integration is built to be seamless, ensuring that users receive real-time feedback after uploading images.

- **Flask API Integration:** The AI model is wrapped in a Flask backend that receives image input, performs analysis, and returns results.
- **Real-Time Feedback:** Once a prediction is made, the results are parsed and displayed instantly within the app interface, providing immediate value to users.

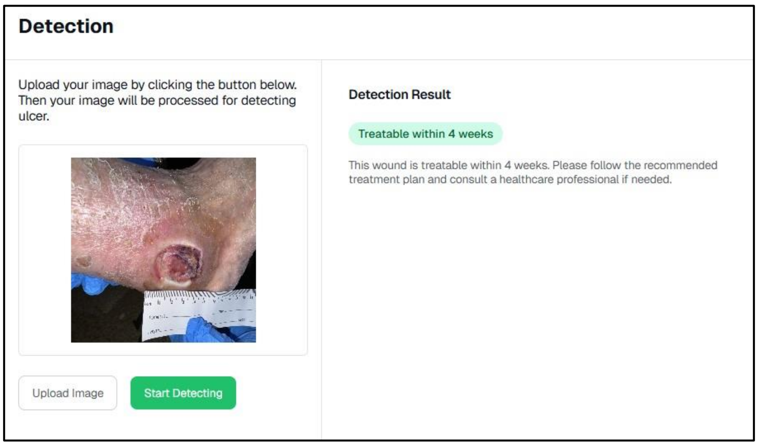

Figure 29:Detection Results on Mobile App

4.2.5 Testing & Debugging

To ensure the app worked smoothly and met user expectations, we conducted thorough testing under different real-world conditions.

- **Unit & Integration Testing:** Tested individual components and their interactions using Jest and React Native Testing Library.
- **Device Testing:** Conducted testing on both emulators and real devices to catch any UI or performance issues**.**

4.2.6 Deployment

The application was prepared for public use through careful packaging, testing, and publishing steps.

- **APK Generation:** Used React Native CLI to produce a signed APK optimized for release.
- **Code Optimization:** Applied Proguard rules and other performance enhancements to secure and streamline the final build.
- **Device Testing:** Deployed the APK to Android devices for final checks.
- **Play Store Publishing:** Set up a Google Play Developer account, added required information (screenshots, description, privacy policy), and published the app through the Play Console.

4.3 Challenges Faced

Throughout the development of the mobile application, the team encountered several practical challenges that required thoughtful solutions and iterative improvements.

- **Cross-Platform Compatibility:** Designing an app that looks and feels consistent across different Android devices wasn't always straightforward. Differences in screen sizes and OS versions sometimes caused layout issues, so we had to spend extra time fine-tuning the UI and testing responsiveness on various screen types.
- **API Communication and Error Handling**: Establishing smooth communication between the mobile app, the Node.js backend, and the Flask server wasn’t as simple as expected. During testing, there were moments when image uploads would fail or detection results wouldn’t return properly. This pushed us to build stronger error handling and make the app more resilient to network hiccups.
- **Authentication Integration:** We wanted to offer users both traditional email/password login and the convenience of Google sign-in. But combining the two securely meant we had to carefully manage session tokens and ensure they worked seamlessly across different devices, which took considerable planning and testing.
- **Model Accuracy and Image Quality:** The accuracy of the detection model heavily depended on how users captured foot images. Poor lighting, wrong angles, or unclear photos sometimes led to incorrect results. To tackle this, we added prompts and tips to help users take better pictures, improving both usability and detection accuracy.
- **Deployment and APK Signing:** When it came time to release the app, building a production-ready APK brought its own set of hurdles. Setting up the keystore, signing the building, and making sure everything was optimized and secure without breaking any functionality required a careful step-by-step process.

4.4 Future Work

As we look ahead, there are several meaningful enhancements we aim to explore for both the mobile and web platforms to make the system more impactful, accessible, and scalable:

- **Offline Detection Capability**
  Internet access is limited or unreliable in many rural and remote regions. To overcome this barrier, the system will include an offline detection feature within the mobile application. This will allow patients to perform ulcer screening directly on their devices without needing an active internet connection.

- **Real-Time Telemedicine Features**
  The system will incorporate real-time communication options such as chat and video consultations. These features will connect patients with healthcare professionals quickly and conveniently. Users can seek immediate guidance, discuss symptoms, and receive advice without leaving their homes.

- **Integration with Wearable Health Devices**
  Wearable devices such as smart insoles that monitor foot pressure and temperature changes will be connected with the system. These devices provide continuous health monitoring and generate early alerts for possible ulcer development before symptoms become visible.

- **Improved Accessibility and Language Support**
  Making the system accessible to all users is a priority. Future updates will include support for multiple regional languages widely spoken in Pakistan, including Punjabi, Sindhi, Pashto, and Balochi. This will ensure users from different cultural and linguistic backgrounds can navigate and use the system comfortably.

- **Expanding Disease Detection to Other Conditions**

  The system's scope will be expanded beyond diabetic foot ulcers to detect other diseases common in vulnerable populations. These include conditions such as leishmaniasis, conjunctivitis, and acute respiratory infections. Special focus will be placed on regions prone to disease outbreaks following floods or other natural disasters. Early detection in these areas can support timely medical response, reduce disease spread, and improve overall community health.

Chapter 5

Conclusions

5.1 Summary

This chapter encapsulates the aims, objectives, findings, and achievements of the diabetic foot ulcer (DFU) detection project, summarizing the critical aspects discussed in earlier chapters. The project has aimed to develop an efficient, accurate, and scalable solution for early detection and classification of DFUs using advanced image processing and machine learning techniques. By leveraging clinical data from Bosch
Pharma and publicly available datasets, this project has addressed significant gaps in DFU diagnosis while striving to improve patient outcomes.The research has successfully incorporated innovative preprocessing strategies, such as noise removal, image scaling, and data augmentation, to enhance image clarity and tackle data imbalances. These measures have improved the reliability and generalization capabilities of the proposed models. Furthermore, the adoption of a binary-to-multiclass classification approach has enabled a nuanced understanding of ulcer severity, contributing to a more precise diagnostic framework. Employing state-of-the-art machine learning models has also ensured alignment with the project's overarching objectives of accessibility, scalability, and efficiency.The findings demonstrate that integrating high-quality clinical data with advanced computational models can significantly enhance the accuracy and reliability of DFU detection systems. However, the project has identified certain constraints, such as the limited availability of diverse and balanced datasets and challenges in achieving real-time deployment efficiency. These limitations have influenced the development of recommendations for future work.

5.2 Recommendations for Future Work

1. **Expand Dataset Diversity and Quality**
To improve model accuracy, especially across different demographics, future work should focus on expanding the dataset with diverse foot images from various ethnic groups, age ranges, and ulcer severity levels. Collaborating with more healthcare facilities could increase the volume and diversity of data, making the model more robust in real-world settings.
2. **Enhance Model Performance through Advanced Techniques**
Employ more advanced deep learning techniques, such as ensemble learning, or explore transformer-based models that may better capture ulcer characteristics. Additionally, further fine-tuning of the CNN architecture can enhance detection accuracy while maintaining real-time performance on mobile devices.
3. **Introduce Multi-Stage Classification**
Future iterations could incorporate a multi-stage classification system to assess the severity of detected ulcers, categorizing them as mild, moderate, or severe. This could provide patients with more actionable insights and assist healthcare providers in prioritizing cases based on urgency.
4. **Implement Offline Functionality**
To ensure usability in areas with limited internet connectivity, future development could focus on enhancing offline capabilities, such as using edge computing and lightweight models optimized for mobile devices. This would allow the application to analyze images and detect ulcers without relying on cloud processing.
5. **Incorporate Additional Health Parameters**
Expanding the app to include the monitoring of other diabetes-related conditions, such as skin integrity or circulation issues, could increase its utility for diabetic patients. Integrating these metrics would provide a more holistic tool for managing diabetic foot health.
6. **Establish Data Privacy and Compliance Standards**
As the app collects sensitive medical data, future work should establish compliance with health data regulations (such as HIPAA or GDPR) to ensure data privacy and security.

References

[1] R. Sarmun, M. E. H. Chowdhury, and M. F. A. Rahman, "Diabetic Foot Ulcer Detection: Combining Deep Learning and Image Processing," 2024. [2] A. Shata and N. Abdulaziz, "Early Detection of Diabetic Foot Ulcer Using CNNs," 2021.

[3] M. Muthuraja, S. Natarajan, and N. Aravindhraj, "EfficientNet-Based DFU Detection from Kaggle Dataset," 2024.

[4] C. Giridhar, B. Akhila, and S. Pranay, "Detection of Multi-Stage Diabetes Foot Ulcers Using CNNs," 2024.

[5] T. Nguyen, P. Tran, and M. Dinh, "Ensemble Approaches for DFU Classification," IEEE Trans. Med. Imaging, vol. 42, no. 3, pp. 678–685, 2023. [6] R. Gupta and S. Sharma, "Hybrid Models for Accurate DFU Classification," Journal of Medical Imaging Research, vol. 10, pp. 112–120, 2022.

[7] H. Zhao, X. Lin, and K. Chen, "Socio-economic Impacts of Untreated DFUs: A Machine Learning Perspective," 2024.

[8] F. Ali, A. M. Khan, and N. Saleem, "Contrast Enhancement Techniques for Improving DFU Detection," 2023.

[9] Z. Hassan, Y. Javed, and K. M. Siddiqui, "Artifact Removal in DFU Preprocessing," 2023.

[10] A. Khan, M. Ahmed, and S. Rehman, "Attention Mechanisms in DFU Detection Using CNNs," 2024.

[11] L. Wang, T. Zhang, and J. Yang, "Efficient Image Segmentation for Diabetic Foot Ulcers," 2022.

[12] D. Patel and S. Gupta, "DFU Detection in Resource-Constrained Environments," 2023.

[13] K. Brown and M. Taylor, "Evaluation Metrics for Medical Image Analysis Models," 2021.

[14] N. Kumar and P. Singh, "Transfer Learning for DFU Detection: A Comprehensive Study," 2024.

[15] Y. Tanaka and S. Hirose, "Edge Detection Techniques in Medical Imaging," 2022

Glossary

Convolutional Neural Network (CNN)
A deep learning algorithm which can take in an input image, assign importance to various aspects/objects in the image, and differentiate them from each other.

Diabetic Foot Ulcer (DFU)
An open sore or wound that occurs in individuals with diabetes, typically on the bottom of the foot, often caused by diabetic neuropathy.

DFUC (Diabetic Foot Ulcer Challenge)
A competition or dataset specifically created to improve research and development of algorithms for detecting diabetic foot ulcers.

MERN
A technology stack that includes MongoDB, Express.js, React.js, and Node.js, often used in developing full-stack applications.

TensorFlow
An open-source library developed by Google for deep learning and machine learning applications.

Edge Computing
A distributed computing paradigm that brings computation and data storage closer to the data source, which improves response times and saves bandwidth.

Bilingual
Supporting two languages, in this context English and Urdu, for accessibility purposes in the application.

Google OAuth
An open standard for access delegation commonly used as a way to grant websites or applications limited access to user information without exposing passwords.

Amazon Web Services (AWS)
A comprehensive cloud computing platform offered by Amazon, providing a variety of services such as storage, databases, and computing power.

Data Augmentation
A technique used in machine learning to increase the diversity of training data without collecting new data, often through transformations like rotation, flipping, or scaling images.

Data Preprocessing
Steps taken to clean and transform raw data into a suitable format for modeling, often including normalization, resizing, and noise reduction in images.

Deep Learning
A subset of machine learning involving neural networks with multiple layers, used for complex tasks like image recognition and natural language processing.

Image Scaling
Resizing images to a standard resolution, often to meet model input requirements and ensure consistency across datasets.

JSON Web Token (JWT)
A compact, URL-safe token format used to securely transfer information between two parties, commonly used for authentication in web applications.

Machine Learning Models
Algorithms that learn patterns from data to make predictions or decisions, such as convolutional neural networks (CNNs) used in this report for detecting foot ulcers.

Noise Removal
A preprocessing technique to remove unwanted artifacts or noise from images, improving clarity and helping models detect important features.

Real-Time Monitoring
Continuous observation and analysis of data in real-time, allowing the application to provide instant feedback or alerts to users.

ROC-AUC (Receiver Operating Characteristic - Area Under Curve) *A performance metric for classification models, with a higher AUC indicating better model performance in distinguishing between classes.*

Transfer Learning
A machine learning method where a model developed for one task is reused as the starting point for a model on a different task, improving efficiency and performance.

Report

ORIGINALITY REPORT

15%	10%	6%	10%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Higher Education Commission Pakistan Student Paper	5%
2	www.coursehero.com Internet Source	1%
3	www.mdpi.com Internet Source	<1%
4	www.ssgmce.ac.in Internet Source	<1%
5	open.metu.edu.tr Internet Source	<1%
6	C Kishor Kumar Reddy, P R Anisha, Samiya Khan, Marlia Mohd Hanafiah, Lavanya Pamulaparty, R Madana Mohana. "Sustainability in Industry 5.0 - Theory and Applications", CRC Press, 2024 Publication	<1%

7	Submitted to Consorcio CIXUG Student Paper	<1%
8	mis.itmuniversity.ac.in Internet Source	<1%
9	www.aal-persona.org Internet Source	<1%
10	www.pce.ac.in Internet Source	<1%
11	ami.lnu.edu.ua Internet Source	<1%
12	Gonzales Vera, Ricardo Alonso. "Robust Deep Learning Methods for Accountable Contrast-Agent-Free CMR Imaging in Clinical Applications", University of Oxford (United Kingdom) Publication	<1%
13	Submitted to Oxford House College Student Paper	<1%
14	Sayyafzadeh, Shahrzad. "Thumbnail Image Forensic Acquisition and Analysis in the Age of Artificial Intelligence", Florida Agricultural and Mechanical University, 2024 Publication	<1%

15	Submitted to U niversity of Bedfordshire Student Paper	<1%
16	n3xtcoder.org Internet Source	<1%
17	www.trustabletech.org Internet Source	<1%
18	Joice. C Sheeba, M. Selvi. "Pedagogical Revelations and Emerging Trends", CRC Press, 2025 Publication	<1%
19	Mehdi Ghayoumi. "Generative Adversarial Networks in Practice", CRC Press, 2023 Publication	<1%
20	Submitted to University of Western Australia Student Paper	<1%
21	Submitted to University of Wollongong Student Paper	<1%
22	dspace.cc.tut.fi Internet Source	<1%
23	www.ccsinsight.com Internet Source	<1%

24	Kamran Amjad, Sohaib Asif, Zafran Waheed, Ying Guo. "A Novel Lightweight Deep Learning Framework with Knowledge Distillation for Efficient Diabetic Foot Ulcer Detection", Applied Soft Computing, 2024 Publication	<1%
25	Submitted to University of Abertay Dundee Student Paper	<1%
26	arxiv.org Internet Source	<1%
27	content.iospress.com Internet Source	<1%
28	pmc.ncbi.nlm.nih.gov Internet Source	<1%
29	www.bartleby.com Internet Source	<1%
30	www.tuberktoraks.org Internet Source	<1%
31	Submitted to Grand Canyon University Student Paper	<1%
32	Submitted to Brunel University Student Paper	<1%
33	Submitted to Nims University Rajasthan, Jaipur Student Paper	<1%

34	Submitted to University of Wolverhampton	Student Paper	<1%
35	Submitted to North South University	Student Paper	<1%
36	Submitted to University of Technology, Sydney	Student Paper	<1%
37	Submitted to Technical University of Košice	Student Paper	<1%
38	www.analyticsvidhya.com	Internet Source	<1%
39	www.rapidinnovation.io	Internet Source	<1%
40	Submitted to City University	Student Paper	<1%
41	nuacem.com	Internet Source	<1%
42	repository.canterbury.ac.uk	Internet Source	<1%
43	www.etasr.com	Internet Source	<1%
44	Submitted to CSU, San Jose State University	Student Paper	<1%

45	ijarasem.com Internet Source	<1%
46	sdgresources.relx.com Internet Source	<1%
47	core.ac.uk Internet Source	<1%
48	myd.neduet.edu.pk Internet Source	<1%
49	studentsrepo.um.edu.my Internet Source	<1%
50	www.icaset.in Internet Source	<1%
51	irbackend.kiu.ac.ug Internet Source	<1%
52	researchcommons.waikato.ac.nz Internet Source	<1%
53	Ajay Kumar, Deepak Dembla, Seema Tinker, Surbhi Bhatia Khan. "Handbook of Deep Learning Models for Healthcare Data Processing - Disease Prediction, Analysis, and Applications", CRC Press, 2025 Publication	<1%

54 Duan, Zhihao. "Learning-Based Image Compression for Unified Human and Machine Vision", Purdue University <1%

Publication

55 bmcmedinformdecismak.biomedcentral.com Internet Source <1%

56 webnersolutions.com Internet Source <1%

57 www.irjes.com Internet Source <1%

58 Alex Khang, Vugar Abdullayev, Vladimir Hahanov, Vrushank Shah. "Advanced IoT Technologies and Applications in the Industry 4.0 Digital Economy", CRC Press, 2024 <1%

Publication

59 B. R. Tapas Bapu, Kalaimagal Sivamuni,Dhandapani Samiappan,R. R. Sathiya."Advanced Detection of Diabetic Foot UlcersUsing Hamiltonian Quantum GANs and CellAttention Networks, with Genghis Khan Shark Optimization", Biomedical Materials & Devices, 2025 <1%

Publication

60 Deepa Jose, Preethi Nanjundan, Sanchita Paul, Sachi Nandan Mohanty. "AI-Driven IoT Systems for Industry 4.0", CRC Press, 2024 Publication <1%

61	Shalli Rani. "Emerging Technologies and the Application of WSN and IoT - Smart Surveillance, Public Security, and Safety Challenges", CRC Press, 2024 Publication	<1%
62	Zahra Jafari, Ebrahim Karami. "Breast Cancer Detection in Mammography Images: A CNNBased Approach with Feature Selection", Information, 2023 Publication	<1%
63	cutshort.io Internet Source	<1%
64	dspace-test.interleaf.ie Internet Source	<1%
65	ijain.org Internet Source	<1%
66	khazna.ku.ac.ae Internet Source	<1%
67	mdpi-res.com Internet Source	<1%
68	thesai.org Internet Source	<1%
69	thesis.univ-biskra.dz Internet Source	<1%

70	www.springerprofessional.de Internet Source	<1%