COMPARATIVE ANALYSIS OF DEEP LEARNING MODELS FOR AUTOMATED BONE FRACTURE DETECTION

A PROJECT REPORT

Submitted to

ASSAM DON BOSCO UNIVERSITY

by

ANSHUMAN SARMA

DC2022BCA0002

OWEN WARLARPIH

DC2022BCA0009

NYUBENLO SEB

DC2022BCA0010

in partial fulfilment for the award of the degree of

BACHELOR OF COMPUTER APPLICATIONS

DEPARTMENT OF COMPUTER APPLICATIONS

SCHOOL OF TECHNOLOGY, ASSAM DON BOSCO UNIVERSITY

AZARA, GUWAHATI 781 017,
ASSAM, INDIA.

BATCH (2022-2025)

CERTIFICATE

This is to certify that the Project Report entitled, "**COMPARATIVE ANALYSIS OF DEEP LEARNING MODELS FOR AUTOMATED BONE FRACTURE DETECTION**", submitted by **ANSHUMAN SARMA (DC2022BCA0002)**, **OWEN WARLARPIH (DC2022BCA0009)** and **NYUBENLO SEB (DC2022BCA0010)**, to the Assam Don Bosco University, Guwahati, Assam, in partial fulfilment of the requirement for the award of Degree of Bachelor of Computer Applications is a bonafide record of the mini project carried out by them during the semester July 2024 to December 2024.

Internal Guide:
Dr. Gypsy Nandi
Head of the Department
Department of Computer Applications
School of Technology
Assam Don Bosco University

CERTIFICATE

This is to certify that the Project Report entitled, "**COMPARATIVE ANALYSIS OF DEEP LEARNING MODELS FOR AUTOMATED BONE FRACTURE DETECTION**", submitted by **ANSHUMAN SARMA (DC2022BCA0002)**, **OWEN WARLARPIH (DC2022BCA0009)** and **NYUBENLO SEB (DC2022BCA0010)**, to the Assam Don Bosco University, Guwahati, Assam, in partial fulfilment of the requirement for the award of Degree of Bachelor of Computer Applications is a bonafide record of the mini project carried out by them during the semester July 2024 to December 2024.

Dr. Gypsy Nandi
Head of the Department
Department of Computer Applications
Date:

EXAMINATION CERTIFICATE

This is to certify that **ANSHUMAN SARMA, OWEN WARLARPIH** and **NYUBENLO SEB** bearing Roll Numbers **DC2022BCA0002, DC2022BCA0009** and **DC2022BCA0010** respectively of the **Department of Computer Applications** has carried out the mini project work in a manner satisfactory to warrant its acceptance and also defended it successfully.

We wish them all the success in their future endeavours.

Examiners:

01. Internal Examiner:

02. Internal Examiner:

DECLARATION

We hereby declare that the mini project work entitled "**COMPARATIVE ANALYSIS OF DEEP LEARNING MODELS FOR AUTOMATED BONE FRACTURE DETECTION**" submitted to the Assam Don Bosco University, Guwahati, Assam, in partial fulfilment of the requirement for the award of Degree of Bachelor of Computer Applications is an original work done by us under the guidance of **Dr. GYPSY NANDI** and has not been submitted for the award of any degree.

ANSHUMAN SARMA

DC2022BCA0002

Department of Computer Applications

OWEN WARLARPIH

DC2022BCA0009

Department of Computer Applications

NYUBENLO SEB

DC2022BCA0010

Department of Computer Applications

ACKNOWLEDGEMENT

This work, "**COMPARATIVE ANALYSIS OF DEEP LEARNING MODELS FOR AUTOMATED BONE FRACTURE DETECTION**", is dedicated to all the people who contributed to the successful completion of this project.

This project is dedicated to our guide, **Dr. GYPSY NANDI**. We would like to extend our heartfelt thanks to her for the essential guidance, support, and valuable feedback received throughout the project. Her immense expertise and dedication have allowed us to achieve our objectives under his precise guidance.

We take the utmost opportunity to express our gratitude to **Dr. ABHIJIT BORA**, coordinator of the project, for his constant support, guidance, and prompt suggestions at various stages. We would also like to thank, **Mr. LAL OMEGA BORO,** for significant inputs, counsel, and support during critical phases of our project.

We wish to acknowledge the contributions of various authors and researchers whose studies we have referenced in our literature review. Their study furnished a strong foundation for our study and inspired us to look deeper into the subject.

We give thanks to our families and friends, who have been a great source of motivation and patience in our execution of this project. Emotional support was crucial in keeping our momentum going. We thank one another, members of the project team, for the effort, commitment, and synergy that has led to the completion of this project.

ABSTRACT

Deep learning-based automated fracture detection in bones has acquired much attention in improving diagnostics and efficiency in medical imagery. This paper compares and contrasts four CNN models, such as DenseNet121, ResNet50, MobileNet, and VGG16, applied to images of X-rays. The models will be evaluated based on precision, F1 scores, as well as the accuracy of the models in order to detect fractures from the dataset of 8,683 images. DenseNet121 was the highest at 0.55 with dense connectivity, ResNet50 had a good balanced performance in terms of focusing on complex patterns, while the lightweight architecture of MobileNet offered moderate accuracy for the low-resource environment of 0.52. The lowest accuracy was by VGG16 at 0.46, which struggled with detailed fracture information. In the clinical application, DenseNet121 is the most promising model, and ResNet50 and MobileNet promise much for specific applications. This study highlights the potential for further optimization and architectural improvements for deep learning-based models to make them more clinically feasible for fracture detection.

KEYWORDS: *Bone Fracture Detection, Deep Learning, DenseNet121, ResNet50, MobileNet, VGG16, Medical Imaging*

LIST OF TABLES

LIST OF FIGURES

Table of Contents

Chapter 1: Introduction

1.1 Project Title

Comparative Analysis of Deep Learning Techniques for Automated Bone Fracture Detection

1.2 Introduction

- The project will focus on Automated Bone Fracture Detection using deep learning models applied to medical X-rays.
- The research will explore various algorithms, including Convolutional Neural Networks (CNNs), for detecting fractures.
- The study will evaluate these models based on their accuracy and F-1 scores for identifying fractures.
- The effectiveness of the models will be assessed in terms of reducing false positives and false negatives.
- The goal is to provide insights into the strengths and limitations of different deep learning models.
- The findings aim to advance AI-driven diagnostics in medical imaging for fracture detection.

1.3 Objective

- Perform a detailed comparative study of existing deep learning models for bone fracture detection.
- Assess the accuracy and efficiency of different models using medical X-ray datasets.
- Determine which model performs best in terms of fracture detection accuracy, minimizing false positives and negatives.
- Offer recommendations and insights on which models are most suitable for real-world fracture detection applications.

1.4 Problem Statement

Manual identification of bone fractures in X-ray images by radiologists is prone to errors, leading to potential misdiagnosis or delays, especially in regions with a shortage of specialists. Automated systems using deep learning models can assist by improving accuracy and speed. However, the effectiveness of various deep learning techniques, such as CNN-based models, for detecting fractures is not fully understood. This study aims to conduct a comparative analysis of these models to evaluate their accuracy, robustness, and efficiency in fracture detection

1.5 Literature Review

The review of literature summarizes previous research or advancements connected to the project. It recognizes previous advancements in the field and shows how the current project either expands on or deviates from prior research.

In [1], the author talks about how the use of machine learning to analyze medical imaging has recently gained attention. Since depending only on experts has resulted in intolerable errors, the idea of an automated diagnostic has long been desired. The author also talks about the number of fracture related patients reported annually and the effects of misdiagnosed fractures are severe. The authors proposed the bone fracture detection system to consist of four main modules, which are preprocessing, edge detection, feature extraction and classification.

The authors in [2] compare various bone fracture detection systems and the various body parts where fractures are most likely to occur or most prevalent. The author states that the most common machine learning approach to medical imaging analysis is deep learning, which uses deep neural networks network structures inspired by the human brain to interpret complex datasets. In particular, convolutional neural networks (CNNs) exhibit strong performance with image-based tasks by using convolutional filters to automatically learn and extract features for image understanding. The author also stated that ankle fractures are among the most common injuries treated by orthopedic surgeons, accounting for 9% of all bone fractures, however stating that 23% of ankle fractures are missed on the initial radiographic imaging due to factors such anatomical variance, superposition of structures on radiographs, lack of experience, and high physician workload. As stated by the author, the other common sites of bone fractures are wrist fractures, hip fractures and rib fractures.

In [3], the author talks about the use of ResNet-50, which is a part of ResNet (Residual Network) family, for bone fracture detection. The author states that ResNet50 is an effective neural network type for image recognition applications, such as recognizing objects or patterns in images. It is part of the convolutional neural network (CNN) family of networks, which draws inspiration from the way the brain interprets visual data. ResNet50 is unique because of its depth, or the number of layers that enable it to recognize complex patterns in photos. ResNet50 performs extremely well in object recognition in photos, even in complicated settings, thanks to this depth.

The authors in [4] proposed a bone fracture detection and classification system using deep learning approach. The author states that in a deep learning approach, the size of the data set is an important component and if the dataset is small, then the possibilities of overfitting may arise. Therefore, to overcome this problem, the author proposed the use of data augmentation technique to increase the size of the data set, as in the past several researchers have used data augmentation technique to reduce the error rate of the machine learning model. The author states the images are transformed with the open-source programming language python and keras library. The initial size of the dataset was not shared, but after augmentation, the data set size was increased to 4000, out of which 2000 were cancerous bones and 2000 were healthy bones.

In [5], the author compares various deep learning models used for fracture detection and classification. In the paper, the author extensively talks about two models: A Convolutional Neural Network (CNN) and a Multilayer Perceptron (MLP). In short term, a CNN is a neural network optimized for image and spatial data, using convolutional layers to detect features like edges and shapes, making it ideal for computer vision tasks whereas an MLP is a general-purpose neural network with fully connected layers, best suited for structured, non-spatial data, but less effective for image processing due to its lack of spatial awareness. Each model was tested with the same dataset by the author. After the comparison and testing, it was found that CNN performed noticeably better in terms of accuracy and predictive power than the MLP. The author states that the CNN performed better in fracture identification tasks because of its capacity to extract spatial information and hierarchical characteristics from images.

The author in [6], does research on an Artificial Intelligence diagnostic model for multi-site fracture, which is based on a deep convolutional neural network. Their aim was to employ these models to detect fractures across multiple regions, enhance the precision of single fracture identification, and address the challenge of accurately pinpointing multiple fractures. The inclusion criteria for this study were as follows: (I) only images of the ulnar radius, wrist joint, hand, tibia fibula, ankle joint, and foot were included; (II) the imaging diagnosis of fracture was consistent with the clinical diagnosis; (III) the images were compliant with Digital Imaging and Communications in Medicine (DICOM) 3.0 and the exclusion criteria were as follows: (I) duplicate image data; (II) patients under 18 years of age; (III) images with postoperative internal and external fixations; (IV) poor image quality; (V) the images did not pass the labelling-review sample check process of the gold standard for fracture area labelling. In this study, the faster R-CNN algorithm was used to detect single and multiple fracture lesions simultaneously. A single fracture is one image containing one fracture, and multiple fractures are multiple fractures in the same image.

The author of [7] does a systematic review and diagnostic test accuracy meta-analysis of various artificial intelligence for scaphoid fracture detection. The scaphoid is a small, boat-shaped bone located in the wrist. It is one of the eight carpal bones that make up the wrist joint and sits on the thumb side of the wrist, near the base of the radius (one of the forearm bones). The study states that Scaphoid fractures are the most common type of carpal bone fracture, with an incidence of 82–89%. The relatively limited number of studies evaluating the function of AI in the detection of scaphoid fracture on wrist radiographs and their recent publication (2020 or later) reflect the fact that this field is new, active, and evolving. The preliminary results showed great promise: 10/19 and 8/19 CNNs showed excellent and good AUC performance, respectively. Meaning, these models were able to accurately detect scaphoid fractures in plain radiography after training on considerably small datasets.

In [8], the author proposed AI model for vertebral fracture detection using YOLOv4 and ResUNet. The author stated that vertebral fractures are a common problem and the most prevalent of thoracolumbar compression and burst fractures. However, vertebral fractures are difficult to diagnose: an experienced orthopaedist or radiologist is required to detect and determine the type of vertebral fracture. The author proposed a method that is capable of distinguishing between burst or compression fractures in the thoracolumbar region using X-ray images. The author also compared their model accuracy to existing models such as support vector machine, extreme gradient boosting (XGBoost), random forest, multilayer perceptron, and k-nearest-neighbor models.

In [9], the author presents a comprehensive review of how deep supervised learning techniques are transforming the detection of bone fractures in radiological images. The authors highlight that traditional fracture detection methods often result in missed diagnoses, leading to delayed treatment. This review emphasizes how deep learning, a subset of artificial intelligence, can assist radiologists by improving the accuracy and efficiency of bone fracture detection. The study also discusses the current challenges faced in implementing these techniques and offers insights into future developments in the field.

In [10], the author discusses on the feasibility and accuracy of automatic detection and classification of rib fractures on thoracic CT using convolutional neural network. The author developed a CNN model to test if the efficacy was comparable to that of a trained radiologist. After the test, the AI-assisted diagnosis attained a precision of 91.1% and sensitivity of 86.3%, measurably surpassing the unaided work of experienced radiologists and requiring significantly less time. In summary, the model suggests the feasibility of AI-assisted diagnosis of rib fractures, which could improve diagnostic efficiency, and reduce diagnosis time and radiologists' workload.

The table below (Table 1.1) shows a comparative analysis of the research paper that were studied above mentioning the various techniques used in each of the research papers.

Table 1.1 Literature Review

Sl. No	Paper Title	Year	Techniques used / Models used
1.	Detection of bone fracture based on machine learning techniques [1]	2023	Pre-processing: RGB to Grayscale conversion, Gaussian Filter, Canny edge detection, Gray Level Co-occurrence Matrix (GLCM) approach. Classification: Support Vector Machine (SVM), and Random Forest (RF).
2.	The Role of Artificial Intelligence in the Identification and Evaluation of Bone Fractures [2]	2024	Comparison of performance of various techniques used for fracture detection
3.	CNN BASED BONE FRACTURE DETECTION FOR MEDICAL IMAGING USING RESNET-50[3]	2024	ResNet50(CNN Architecture)
4.	Bone Fracture Detection and Classification using Deep Learning Approach [4]	2020	Data Augmentation and Deep CNN Model
5.	Comparative Analysis of Deep Learning Models for Fracture Detection and Classification in X-ray Images [5]	2024	Comparison of performance of various techniques used for fracture detection
6.	Artificial intelligence diagnostic model for multi-site fracture X-ray images of extremities based on deep convolutional neural networks [6]	2024	Unet-based bone segmentation algorithm and R-CNN detection algorithm
7.	Artificial Intelligence for X-ray scaphoid fracture detection: a systematic review and diagnostic test accuracy meta-analysis [7]	2023	Convolutional Neural Network (CNN)
8.	Automated detection of vertebral fractures from X-ray images: A novel machine learning model and survey of the field [8]	2024	YOLOv4, ResUnet, Random Forest Approach

9.	Bone Fracture Detection Using Deep Supervised Learning from Radiological Images: A Paradigm Shift [9]	2022	Deep Supervised Learning Techniques
10.	Automatic Detection and Classification of Rib Fractures on Thoracic CT Using Convolutional Neural Network: Accuracy and Feasibility[10]	2020	Faster R-CNN and YOLOv3 (You Only Look Once, version 3)

1.6 Proposed Plan

This project will include the following important stages:

1) Data Collection:

- Gather a comprehensive dataset of X-ray images, ensuring a representative sample of both fractured and healthy bone images.

2) Model Selection and Development:

- Select deep learning models for comparison, including DenseNet121, ResNet50, EfficientNetB0, and VGG16.
- Train these models on the dataset under standardized conditions to ensure comparability.

3) Performance Evaluation:

- Evaluate each model using accuracy, precision, recall, and F1 score.
- Assess each model’s speed, robustness, and its effectiveness in minimizing false positives and false negatives.

4) Benchmarking and Comparative Analysis:

- Create a benchmark report that details the strengths and weaknesses of each model.
- Analyze results to identify the most effective model for real-world bone fracture detection.

5) Recommendations and Future Improvements:

- Based on the comparative analysis, provide recommendations for model improvements and applications in clinical diagnostics.

Chapter 2: Feasibility Study

A feasibility analysis evaluates the viability of a suggested project or system. It assesses different factors like technical, operational, economic, and schedule feasibility to guarantee the project's success.

2.1 Technical Feasibility:

Technical feasibility assesses the available technical resources for the project to ascertain if they are adequate for successful project completion.

2.1.1 Hardware Requirements:

The table below mentions the minimum hardware requirement specifications required by the developers and users.

Table 2.1 Hardware Requirements

Category	For Developers
Hardware Requirements	- High-performance GPU for training deep learning models
	- Server or cloud infrastructure for hosting the DL model and data
	- High-capacity storage devices for large datasets

2.1.2 Software Requirements:

The table below mentions the minimum software requirement specifications required by the developers and users.

Table 2.2 Software Requirements

Category	For Developers
Software Requirements	Language: Python 3.12.7
	Deep Learning Frameworks: TensorFlow, Keras for model development
	IDE: Jupyter Notebook, Google Colab
	- Cloud Storage solutions for storing medical images

2.2 Operational Feasibility:

Operational Feasibility examines the practicality of implementing AI models for fracture detection in clinical settings, focusing on scalability, integration, and accessibility.

- **Scalability**: DenseNet and ResNet, with high accuracy and deeper architectures, are suited for large hospitals with robust computational resources. MobileNet's lightweight design is ideal for low-resource environments like rural clinics, making it versatile for diverse healthcare settings.
- **Integration:** These models can be integrated into existing radiology workflows via cloud-based platforms or PACS systems. Their ability to provide real-time diagnostic support can enhance radiologist efficiency by acting as a second-opinion tool, reducing the risk of human error.

2.3 Economic Feasibility:

Economic feasibility assesses the financial elements of the project, such as cost-benefit analysis, to establish if the project is financially feasible.

2.3.1 COCOMO Model:

The Basic COCOMO model is a static, single-valued model that computes software development effort (and cost) as a function of a program size expressed in estimated lines of code (LOC).

COCOMO Model Constructive Cost Model (COCOMO)

The basic COCOMO equations take the form:

Effort Applied (E) = ab (KLOC)bb, [person-months]

Development Time (D) - cb (Effort Applied)b * d [months]

People Required (P) = Effort Applied/Development Time [Count].

Where KLOC is the estimated number of delivered lines (expressed in thousands) of code. The coefficients ab, bb, cb and db are given in the following table:

Table 2.3 COCOMO MODEL

Software Project	ab	bb	cb	db
Organic	2.4	1.05	2.5	0.38
Semi – detached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.32

Our project type is Organic project, Estimate LOC = 3500

Now the basic COCOMO equation of our project is

Effort Applied (E) = ab (KLOC) bb [person-months]

= 2.4(3.5K)1.05 [person-months] = 9.31 [person – months]

Development Time (D) = cb (Effort Applied) db [months]

= 2.5 (9.31)0.38 [months] = 5.84[months]

People Required (P) = Effort Applied/Development Time [count]

=9.31/5.84[count] = 1.59 [count]

The development time for this project is 5.84 months which will require approximately 2 persons. As we have a limited time of approximately 3.5 months to complete this project. Therefore, we will require more people to develop this project. Since we have 3 members in our group the project development time is justified.

2.4 Schedule Feasibility:

Schedule feasibility involves examining the project timeline to confirm it can be finished within the specified time limit.

2.4.1 Work Breakdown Structure:

The Work Breakdown Structure (WBS) breaks down the project into smaller segments, outlining the individual tasks and goals to be accomplished.

The total project development time (in hours), for our web application is ***420 hours***. To further explain the calculation of ours,

Start date of the project = 02/08/2024.

Total number of weeks = 15 weeks

Thus,

Total number of days = 105 days

One day = 4 hours of work

So now,

Total hours = Total number of days * No of work hours per day

= 420 hours

Thus, the total number of hours applicable for this project according to the schedule allotted is 420. The WBS Diagram (Fig 2.1) shown below, further describes the work hour load allotted for each task.

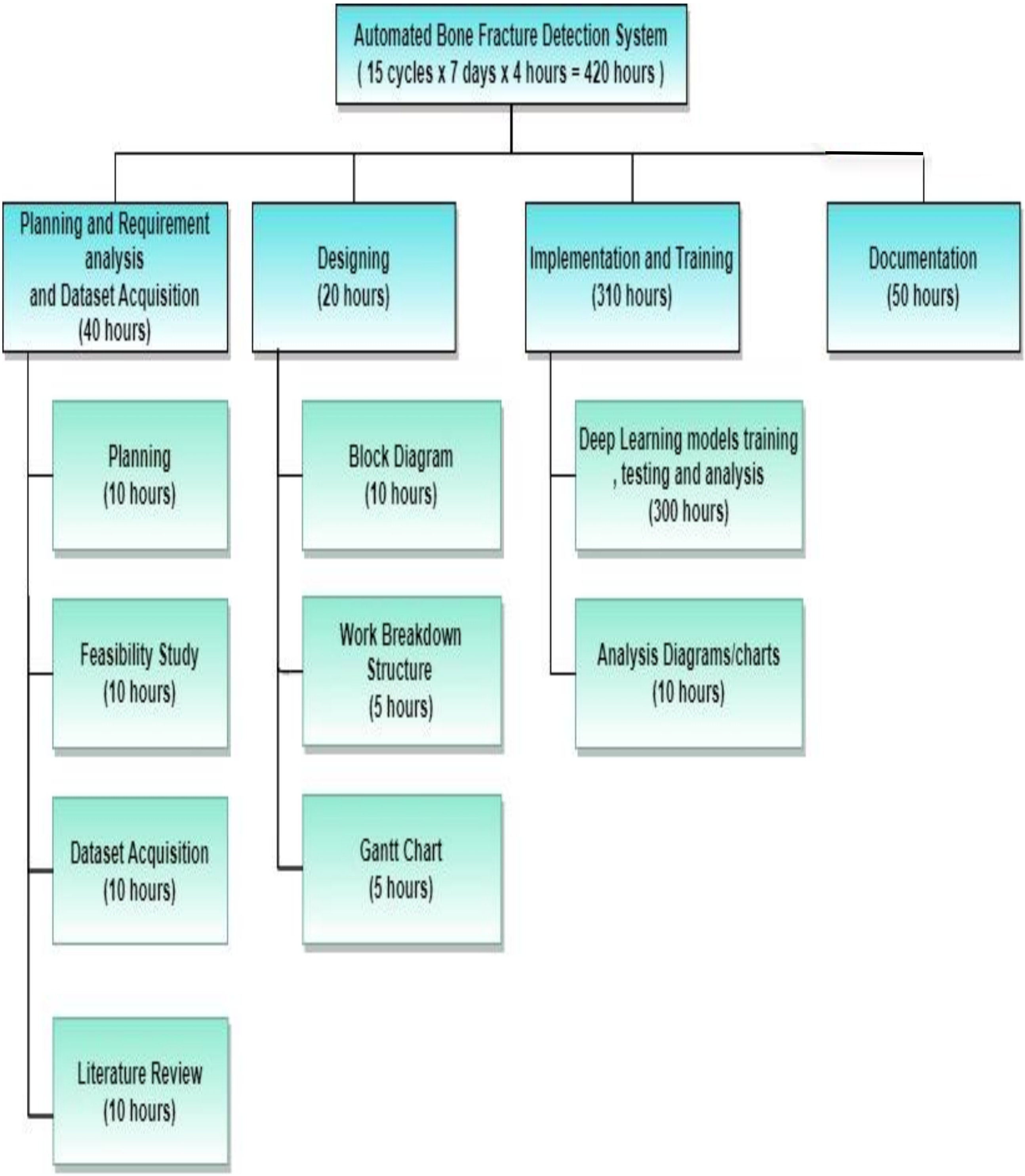


Fig. 2.1 Work Breakdown Structure

2.4.2 Gantt Chart:

A Gantt chart portrays the project timeline by displaying the start and end dates of each task along with the project's overall schedule. The Gannt Chart (Fig 2.2) illustrated below shows the estimated timeline of completion of individual project task.

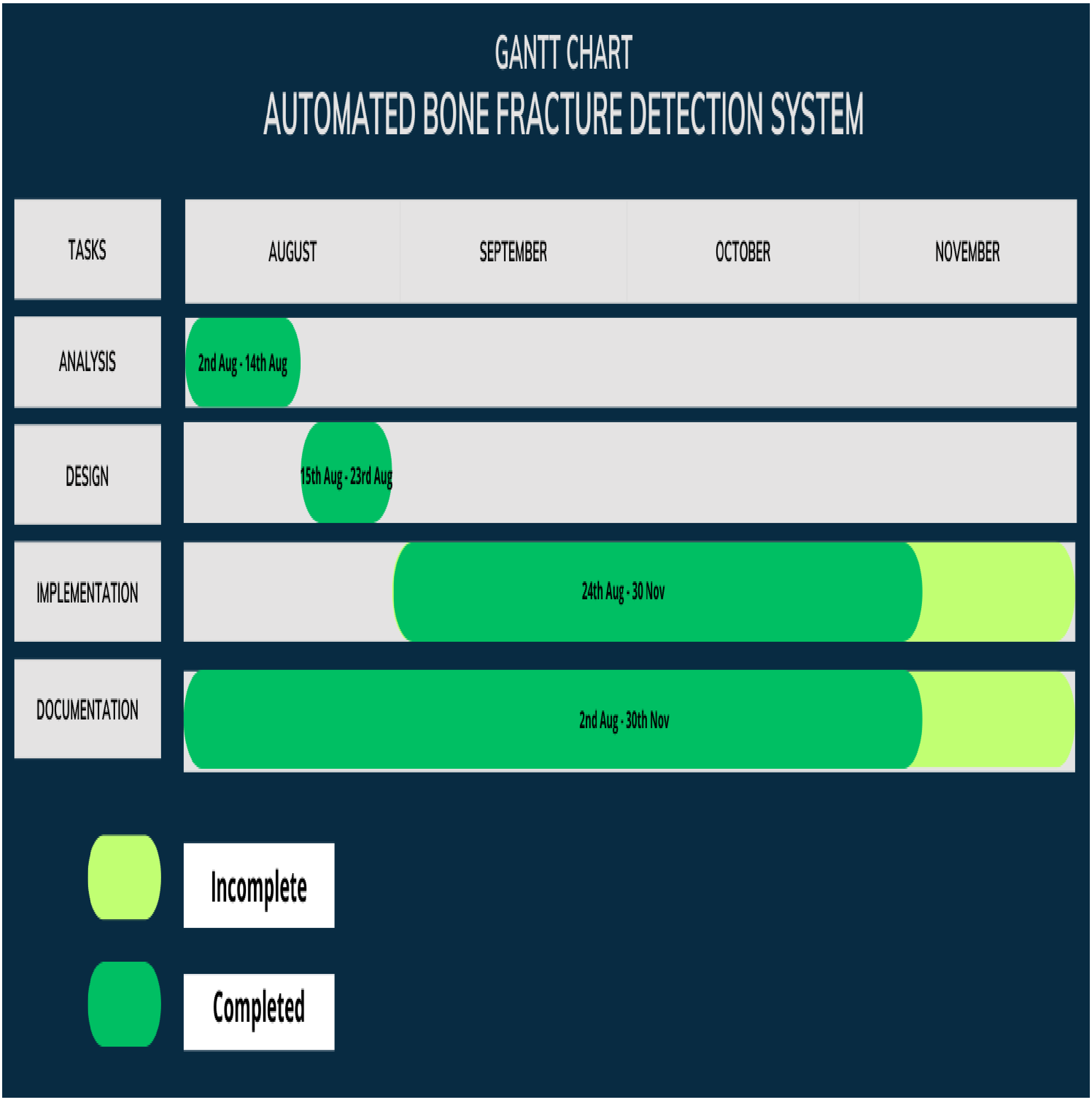


Fig. 2.2 Gantt Chart

Chapter 3: Design Diagram

Design diagrams are essential for illustrating the system's structure and functionality. They offer a visual depiction of different parts, how they are connected, and how information or actions move throughout the system.

A block diagram gives a broad view of the system's structure, showing the primary parts and how they work together. It acts as a roadmap for grasping the organization of the system and the transfer of data among various modules

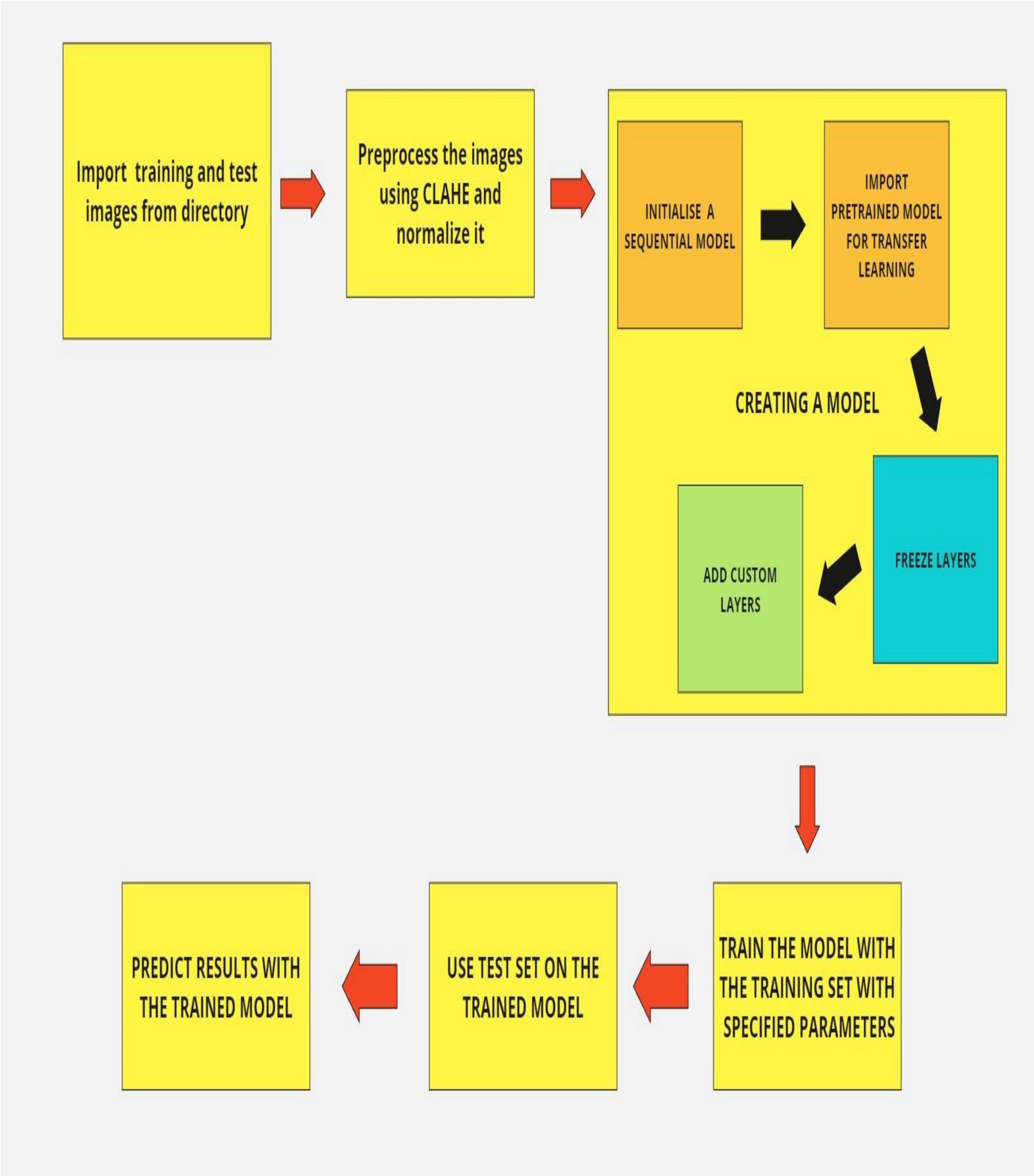


Fig 3.1: Block Diagram of the models (Generalized)

Chapter 4: Methodology and Implementation

This section provides an overview of the methodology employed to conduct a comparative analysis of four convolutional neural network (CNN) models — DenseNet, MobileNet, VGG16, and ResNet50 — for automated bone fracture detection from medical X-ray images.

4.1 Introduction to Deep Learning Models for Fracture Detection

Deep learning, particularly CNNs, has demonstrated substantial success in image recognition tasks, making it a suitable choice for medical imaging applications like fracture detection. This study focuses on four CNN-based architectures, each with unique structural and operational characteristics, enabling detailed evaluation across multiple performance metrics.

- **DenseNet**: DenseNet connects each layer to every other layer in a feed-forward fashion, enhancing feature reuse. By improving gradient flow and reducing redundant learning, DenseNet achieves efficient learning, which is valuable in detecting subtle fractures from complex X-ray images.
- **MobileNet**: Designed for efficiency on mobile and embedded devices, MobileNet utilizes depthwise separable convolutions, reducing computational requirements without sacrificing accuracy. This model is promising for real-time fracture detection in resource-constrained environments.
- **VGG16**: Known for its simplicity and depth, VGG16 comprises 16 layers with small convolutional filters (3x3) and uniform max-pooling, which helps capture detailed image features critical for fracture identification.
- **ResNet50**: ResNet50 introduces residual connections to address the vanishing gradient problem, enabling the network to learn complex patterns in deep layers. Its capability to capture high-level features makes it suitable for identifying complex fractures that might be challenging to detect.

4.2 Dataset Overview

The dataset[11] used for training and validating the models includes a total of 8,683 X-ray images with two primary categories: fractured and non-fractured bones. The images were preprocessed to a resolution of 224x224 pixels to ensure compatibility with the CNN architectures used. Training and validation data were split as follows:

- **Training Set**: 4,384 fractured and 4,299 non-fractured images.
- **Validation Set**: 360 fractured and 240 non-fractured images, totaling 600.
- **Batch Size**: 16
- **Epochs**: 20

Each model was trained with this setup to identify fractures with optimal accuracy while balancing computational efficiency.

Fig 4.1: X-ray images of fractured and non-fractured bones (dataset)

4.3 Model Training and Development

Each model was trained using a standardized dataset of labeled X-ray images, containing both fractured and non-fractured cases. The steps involved in training these models are as follows:

a) **Data Preprocessing**:
   - Images were resized and normalized to meet each model's input requirements.
   - Image enhancement techniques such as CLAHE(Contrast Limited Adaptive Histogram Equalization) was implement to improve the contrast of the images for better readability.

b) **Model Training**:
   - The dataset was split into training and validation sets with the models trained on the training set.
   - Hyperparameters, such as learning rate, batch size, and dropout rates, were tuned to improve validation data performance.
   - Cross-entropy loss and Adam optimizer were used to minimize errors, ensuring model stability during training.

c) Evaluation and Fine-tuning:

- Each model's performance was evaluated based on accuracy, precision, recall, F1-score, and inference time.
- Models were fine-tuned by adjusting layer configurations, learning rates, and dropout rates to improve robustness and reduce errors

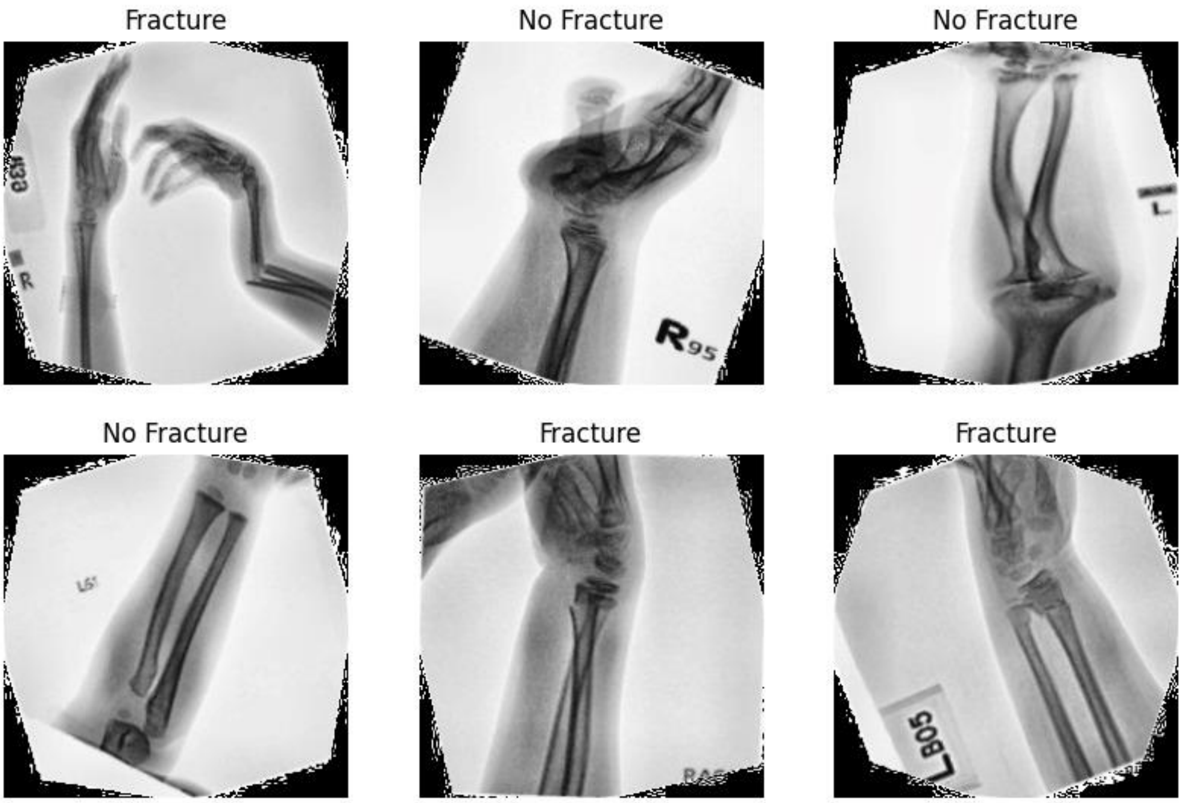


Fig 4.2 Dataset after applying CLAHE function

4.3.1 DenseNet Model Analysis: (Source Code:[12])

DenseNet, short for Dense Convolutional Network, is designed to address some of the limitations of traditional CNNs, particularly the issues related to vanishing gradients, and redundant feature learning. The core concept is dense connectivity, where each layer in the network receives input from all preceding layers.

Architecture:

- Dense Block:
  - A dense block consists of multiple convolutional layers where the output of each layer is concatenated with the outputs of all previous layers within the same block.
  - Each layer performs batch normalization, followed by ReLU activation, a 1x1 convolution (to reduce feature map size), and a 3x3 convolution.

    - This architecture encourages feature reuse, leading to more efficient learning.
- Transition Layer:
    - Between dense blocks, DenseNet uses transition layers that consist of a 1x1 convolution followed by 2x2 average pooling.
    - This helps reduce the number of feature maps, controlling the model size.
- Growth Rate:
    - DenseNet introduces the concept of growth rate, which determines how many filters each layer contributes to the next. This parameter helps control the model's complexity.
    - For DenseNet121, the growth rate is typically set to 32, meaning each layer adds 32 feature maps.
- Final Layers:
    - The model concludes with a global average pooling layer, followed by a fully connected layer with softmax or sigmoid activation, depending on the number of classes.

How it Works:

- DenseNet's core idea is that each layer gets access to the raw outputs of all previous layers, which helps mitigate vanishing gradients and promotes feature reuse.
- Unlike traditional CNNs, where each layer sees only its immediate previous layer, DenseNet layers are densely connected, enabling the network to learn richer, more diversified features without redundant information.

Training Results:

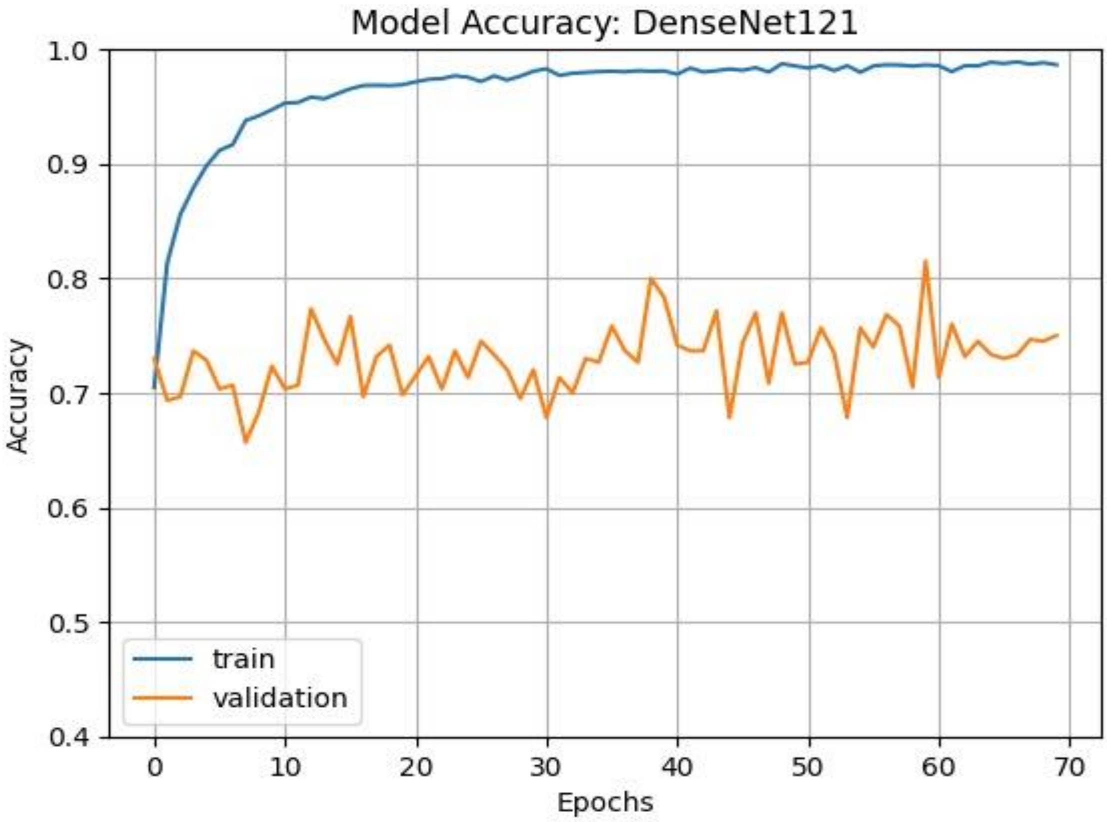


Fig 4.3: Training vs Validation Accuracy: Densenet121

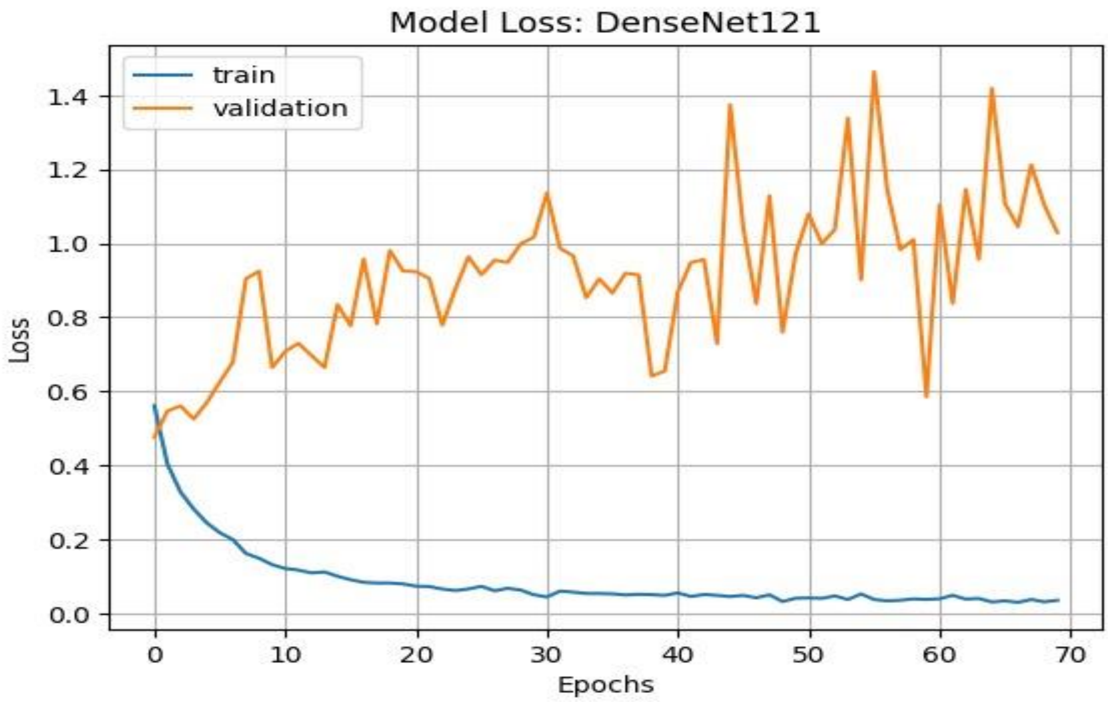


Fig 4.4: Training vs Validation Loss: Densenet121

```
38/38 ━━━━━━━━━━━━━━━━━━━━ 12s 228ms/step
Accuracy: 0.55
Classification Report:
                precision    recall  f1-score   support

 Not Fractured       0.63      0.61      0.62       360
     Fractured       0.44      0.46      0.45       240

      accuracy                           0.55       600
     macro avg       0.53      0.53      0.53       600
  weighted avg       0.55      0.55      0.55       600

Confusion Matrix:
 [[220 140]
 [130 110]]
```

Fig 4.5: Test Accuracy, Classification Report: Densenet121

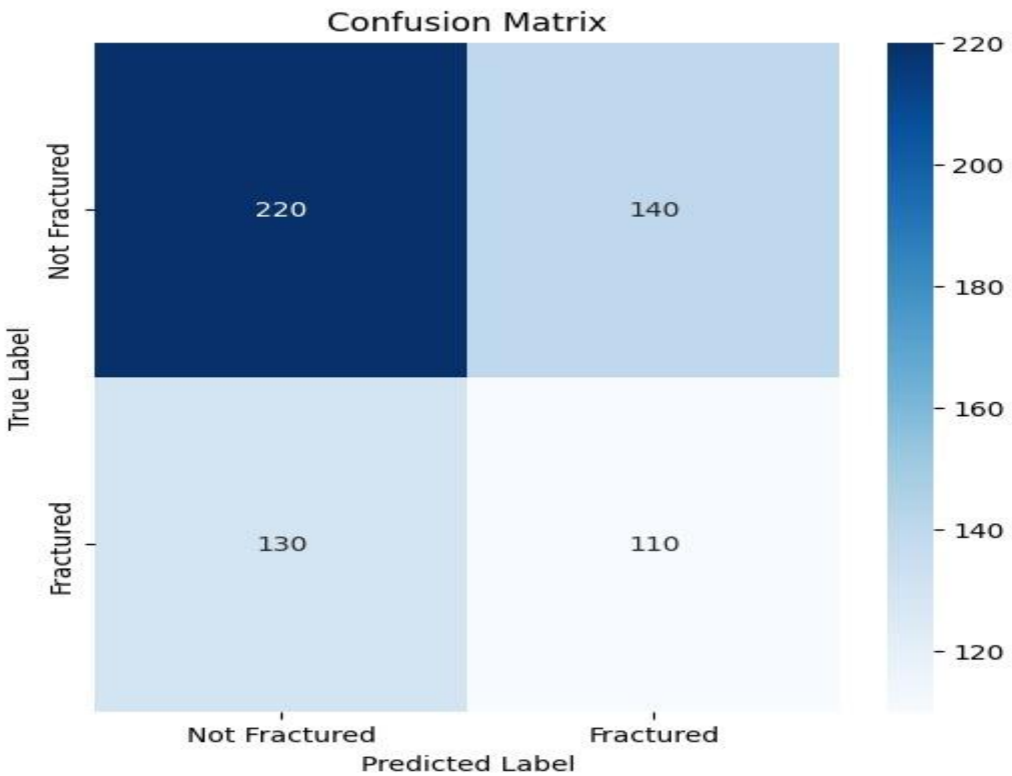


Fig 4.6: Heat Map of Confusion Matrix: Densenet121

DenseNet's densely connected structure ensures that each layer has access to gradients from all preceding layers, which enhances feature learning and reduces redundant computations.

This setup is beneficial for fracture detection, where it's essential to capture subtle changes across various bone textures and structures.

- **Results Interpretation**: With an accuracy of 0.55, DenseNet performed the best among the models, likely due to its effective feature reuse, which helps in learning complex patterns across the dataset's 4,384 fractured and 4,299 non-fractured training images. Its moderate accuracy indicates it captured critical fracture features, but additional training or data augmentation might be necessary to handle edge cases in the dataset, such as ambiguous fractures. Given the image resolution and batch processing setup, DenseNet shows potential as a strong candidate for tasks demanding detail-sensitive feature extraction in X-ray images.

4.3.2 MobileNet Model: (Source Code:[13])

MobileNet model designed for efficient computation, particularly for mobile and edge devices. It introduces the concept of inverted residuals and linear bottlenecks to achieve high performance with fewer parameters.

Architecture:

- Depthwise Separable Convolution:
  - Instead of using standard convolutions, MobileNet utilizes depthwise separable convolutions, which split the convolution operation into two parts:
    1. Depthwise Convolution: Applies a single filter per input channel, focusing on spatial filtering.
    2. Pointwise Convolution (1x1 Convolution): Combines the output of depthwise convolutions across channels, performing feature aggregation.
  - This reduces computational cost significantly.
- Inverted Residual Block:
  - Traditional residual connections are flipped in MobileNet; instead of the residual connection linking two wider layers, it connects thinner layers, hence the term "inverted".
  - Each block starts with a 1x1 convolution (expansion phase), followed by a 3x3 depthwise convolution, and concludes with another 1x1 convolution (compression phase).
  - The activation function used in the expansion phase is ReLU6, chosen for its robustness in low-precision arithmetic (important for mobile devices).

- Linear Bottleneck:
  - The final 1x1 convolution in each block does not use an activation function (linear), avoiding non-linearities that might destroy useful information in low-dimensional feature spaces.

How it Works:

- MobileNet builds efficient models by focusing on reducing the number of multiplications and parameters.
- By employing depthwise separable convolutions and inverted residuals, it achieves a balance between computational efficiency and high model performance.

Training Results:

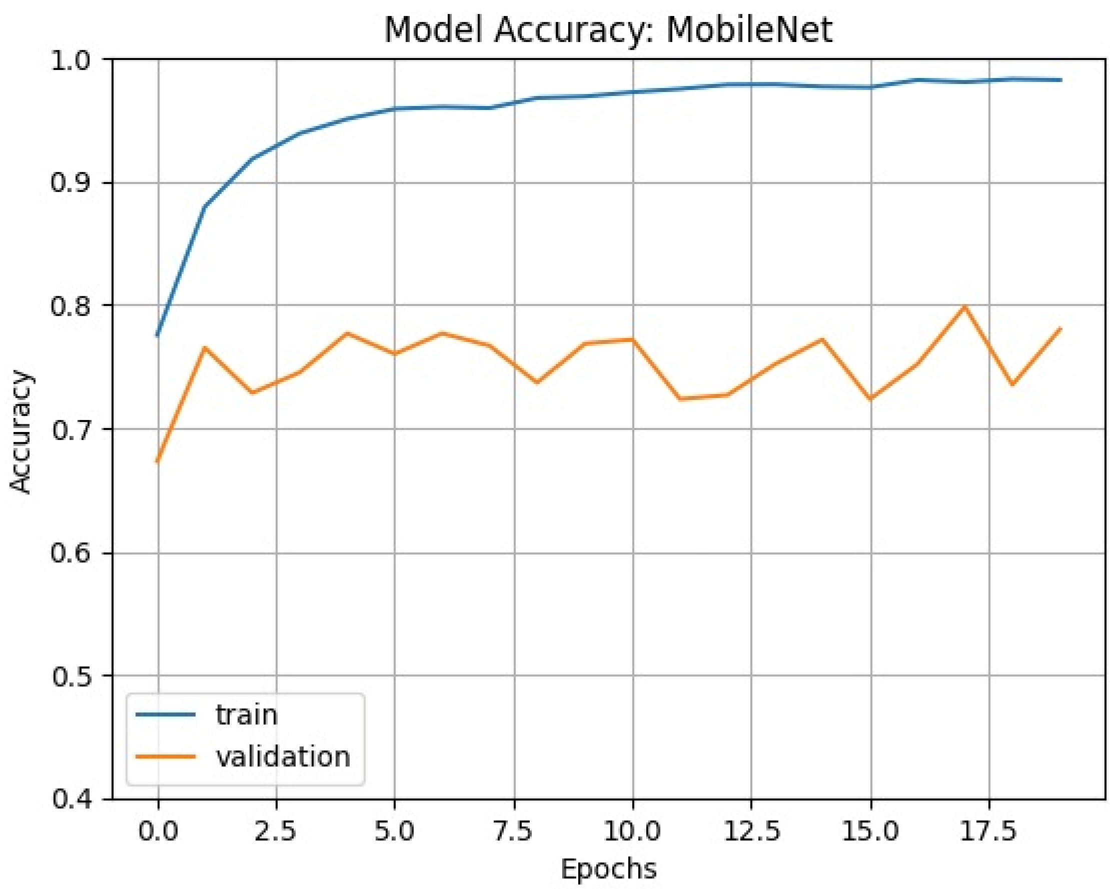


Fig 4.7: Training vs Validation Accuracy: MobileNet

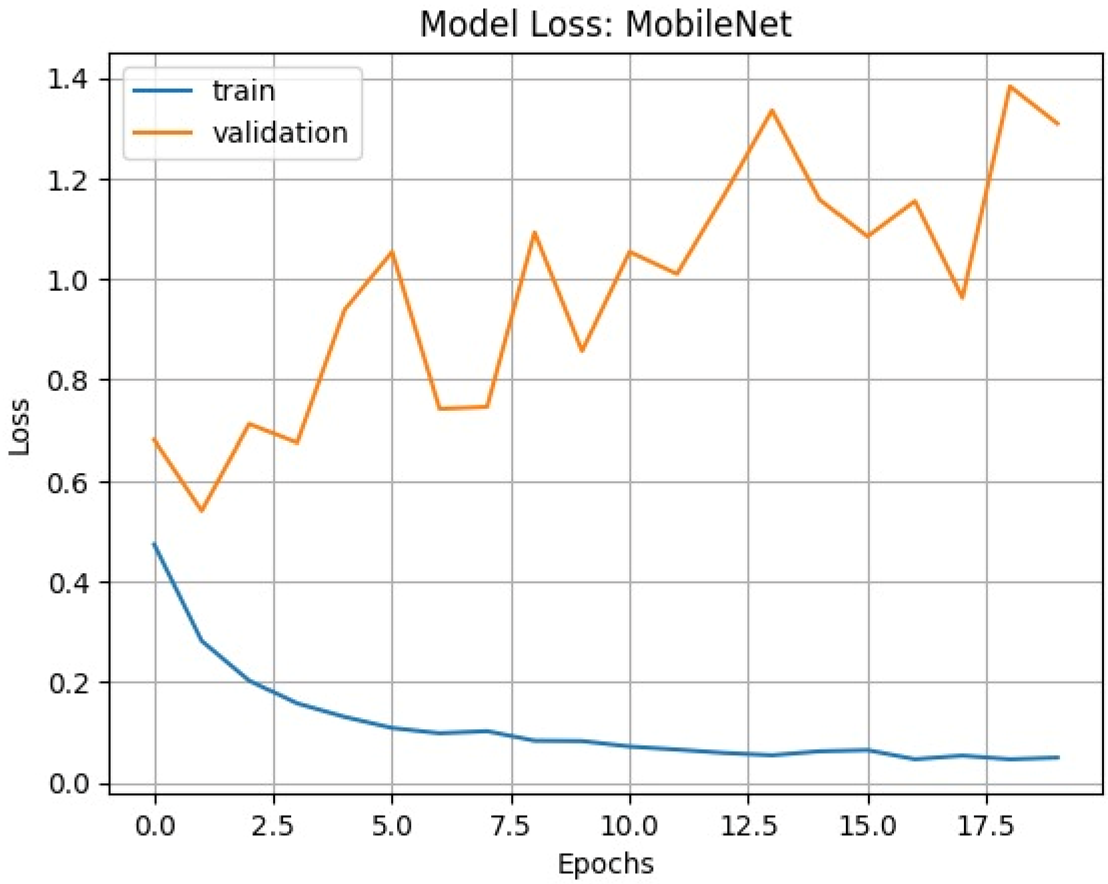


Fig 4.8: Training vs Validation Loss: MobileNet

```
Accuracy: 0.52
Classification Report:
               precision    recall  f1-score   support

 Not Fractured       0.60      0.62      0.61       360
     Fractured       0.40      0.38      0.39       240

      accuracy                           0.52       600
     macro avg       0.50      0.50      0.50       600
  weighted avg       0.52      0.52      0.52       600

Confusion Matrix:
 [[224 136]
 [150  90]]
```

Fig 4.9: Test Accuracy, Classification Report: MobileNet

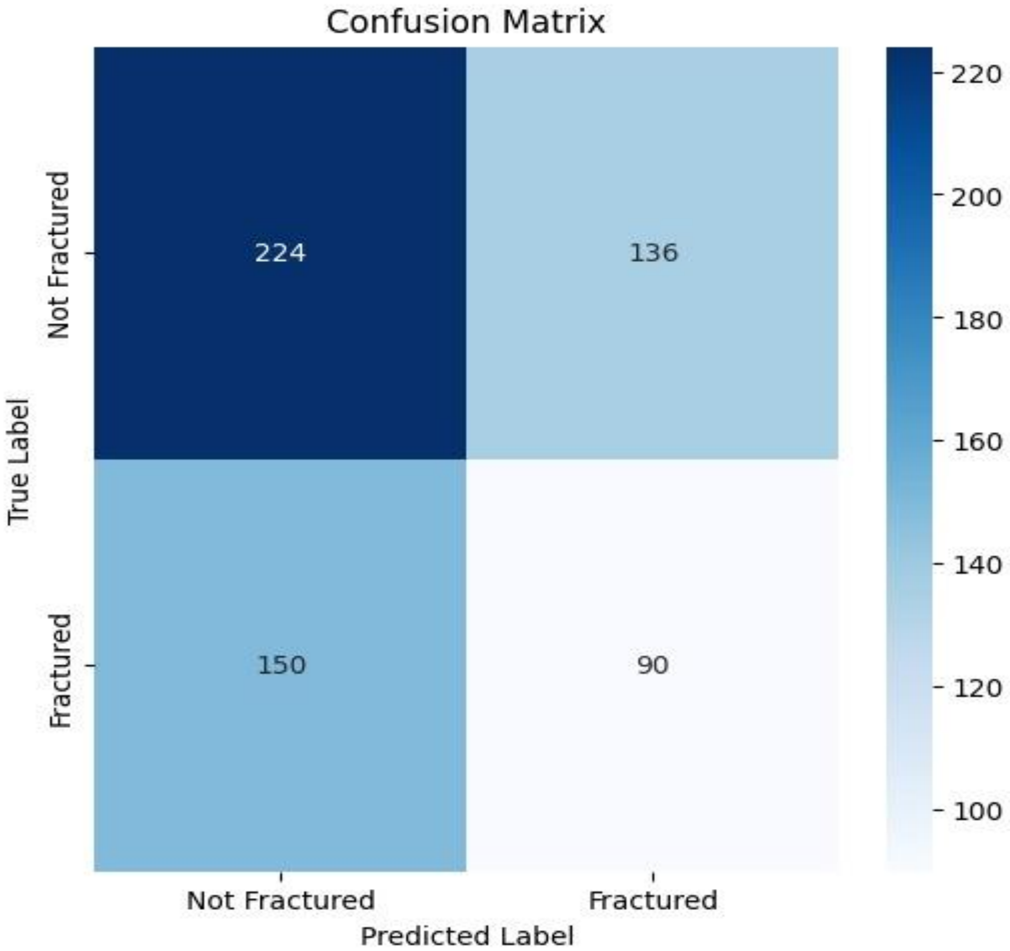


Fig 4.10: Heat Map of Confusion Matrix: MobileNet

MobileNet's lightweight architecture, which uses depthwise separable convolutions, enables efficient learning without significant computational overhead. This efficiency is beneficial given the large dataset size and high image resolution, making MobileNet a viable option for mobile or real-time diagnostic applications.

- **Results Interpretation**: MobileNet achieved an accuracy of 0.52, balancing between speed and fracture detection accuracy. Its lower accuracy compared to DenseNet suggests some limitations in capturing finer fracture details. However, with 20 epochs and a batch size of 16, MobileNet showed strong potential in real-time scenarios where fast inference is required. Despite slightly lower accuracy, MobileNet's architecture makes it an ideal candidate for mobile diagnostics or resource-constrained environments.

4.3.3 VGG16 Model: (Source Code:[14])

VGG16, introduced by the Visual Geometry Group at Oxford, was one of the first deep CNN architectures with a uniform design. It demonstrated that increasing depth with smaller filter sizes (3x3) could yield substantial improvements in image classification.

Architecture:

- Convolutional Layers:
  - VGG16 comprises 16 layers, including 13 convolutional layers and 3 fully connected layers.
  - It exclusively uses 3x3 convolutions with padding and stride of 1, maintaining spatial dimensions across layers.
  - It also uses max-pooling layers (2x2 with stride 2) to downsample feature maps.
- Fully Connected Layers:
  - After the convolutional layers, the network flattens the feature maps and connects them to three fully connected layers.
  - The last layer uses a softmax activation for multi-class classification or sigmoid activation for binary tasks.

How it Works:

- VGG16 processes the input image through a series of convolutional and max-pooling layers, extracting increasingly complex features.
- The small filters (3x3) allow the network to capture intricate details in the image.
- The fully connected layers combine these extracted features for final classification.

Training Results:

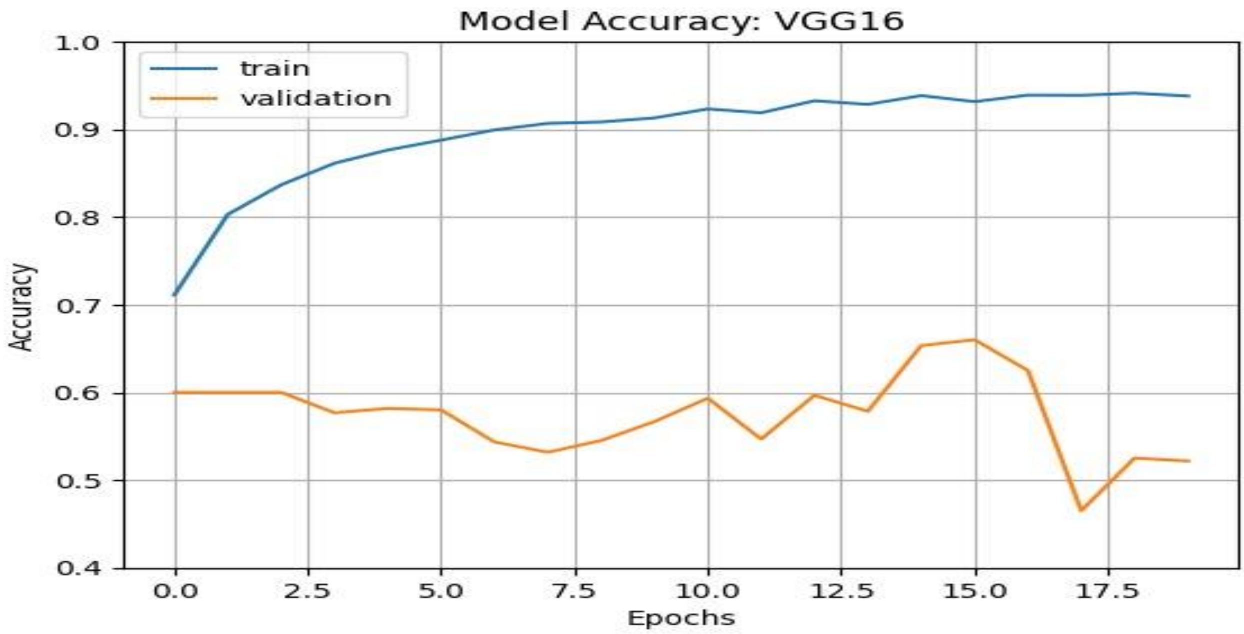


Fig 4.11: Training vs Validation Accuracy: VGG16

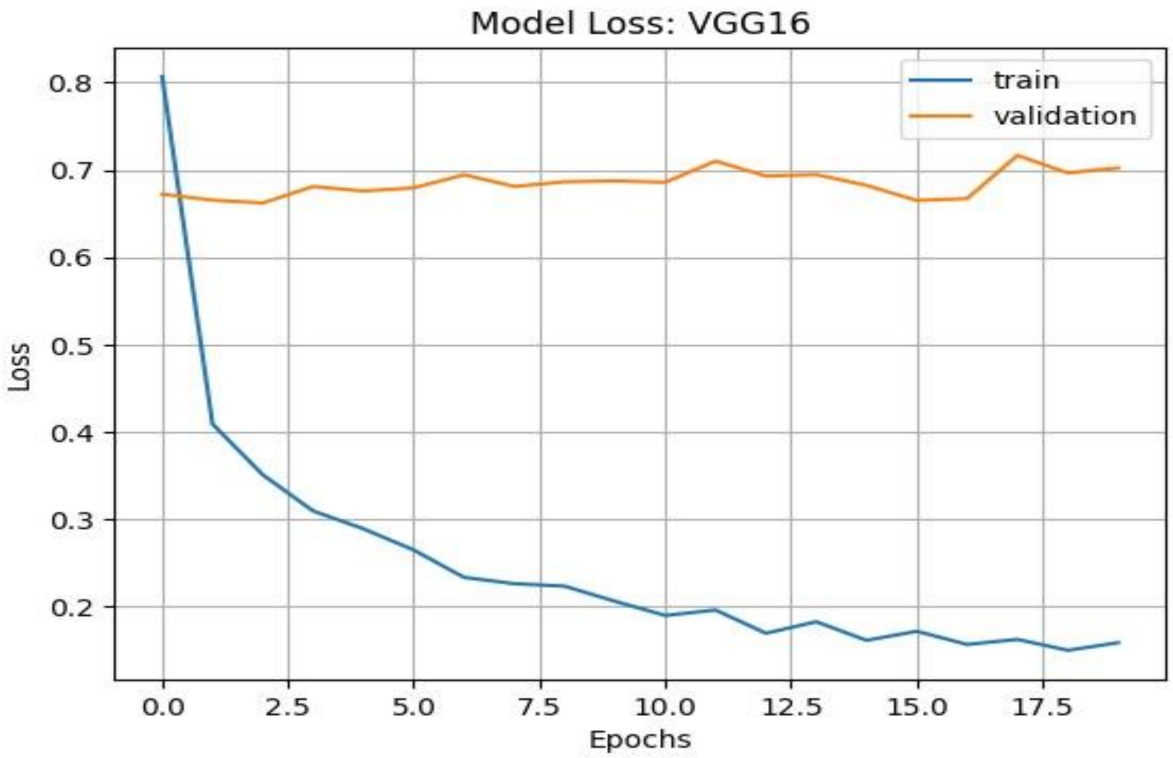


Fig 4.12: Training vs Validation Loss: VGG16

```
38/38 ━━━━━━━━━━━━━━━━━━━━ 19s 481ms/step
Accuracy: 0.46
Classification Report:
                precision    recall  f1-score   support

Not Fractured       0.58      0.33      0.42       360
    Fractured       0.39      0.65      0.49       240

     accuracy                           0.46       600
    macro avg       0.49      0.49      0.45       600
 weighted avg       0.51      0.46      0.45       600

Confusion Matrix:
 [[117 243]
 [ 84 156]]
```

Fig 4.13: Test Accuracy, Classification Report: VGG16

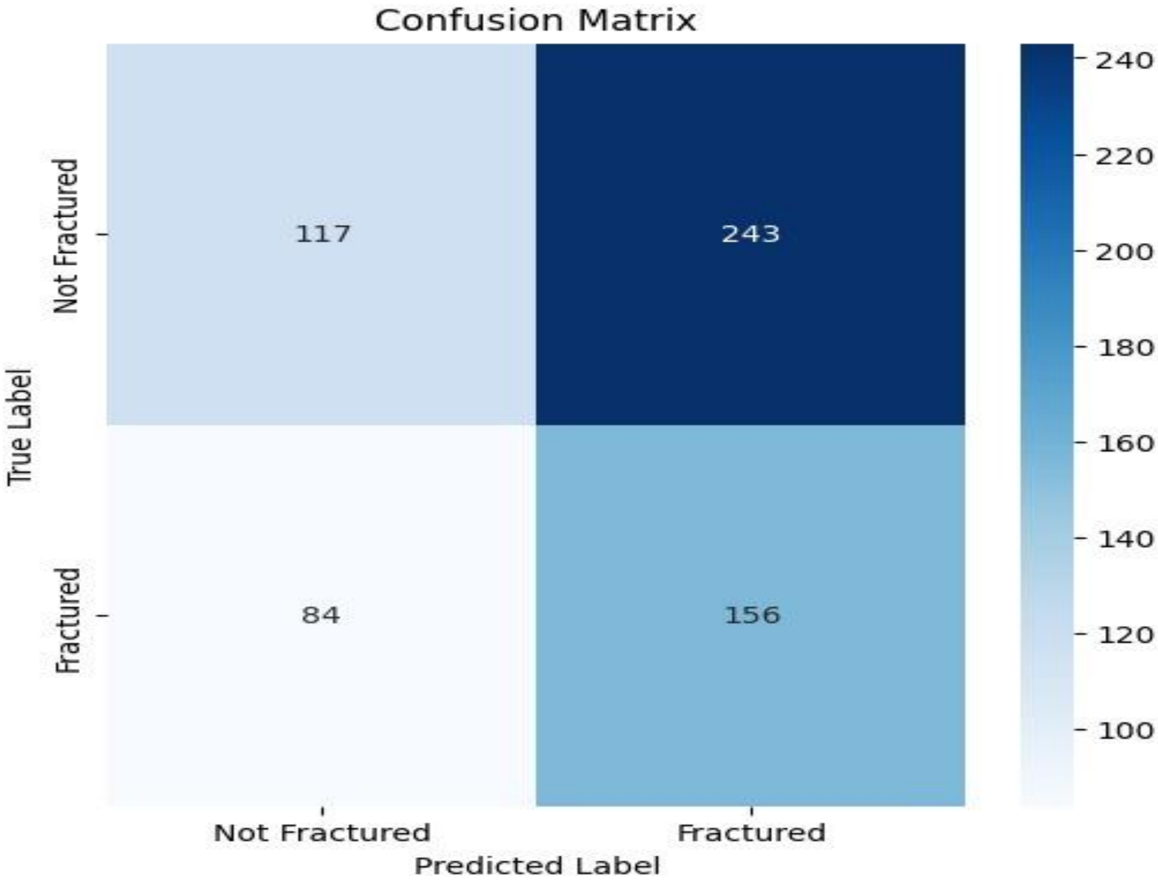


Fig 4.14: Heat Map of Confusion Matrix: VGG16

VGG16, with its simple and uniform layer architecture, uses small (3x3) filters across its 16 layers, which is beneficial for extracting clear, well-defined features. Its consistent structure aids in identifying simple patterns but may lack the complexity needed for detecting subtle fracture characteristics.

- **Results Interpretation**: VGG16 recorded an accuracy of 0.46, the lowest among the models. This performance suggests that while VGG16 is capable of identifying clear fracture patterns, it struggled with complex or ambiguous fractures within the dataset. Given the 4,384 fractured and 4,299 non-fractured training images, VGG16's accuracy indicates it might benefit from additional layers or a hybrid model approach. However, for straightforward cases where fractures are distinctly visible, VGG16 remains a feasible option.

4.3.4 ResNet Model: (Source Code:[15])

ResNet, or Residual Network, introduced the concept of residual learning to tackle the degradation problem observed in very deep networks, where increasing the number of layers leads to higher training error.

Architecture:

- Residual Block:
  - The basic building block of ResNet is the residual block, which contains two or three convolutional layers followed by a skip (identity) connection that adds the input of the block to its output.
  - A standard block consists of:
    - A 1x1 convolution to reduce dimensions (bottleneck).
    - A 3x3 convolution for spatial feature extraction.
    - A 1x1 convolution to restore dimensions.
  - The skip connection is added to the output, enabling the network to learn residual mappings rather than direct mappings.
- Bottleneck Design:
  - ResNet50 uses the bottleneck architecture, reducing the number of parameters while maintaining depth. It has 50 layers consisting of multiple residual blocks.

How it Works:

- Residual connections help the model learn identity functions if necessary, effectively allowing the model to skip certain layers.
- This solves the degradation problem, enabling very deep networks (hundreds of layers) to be trained effectively.

Training Results:

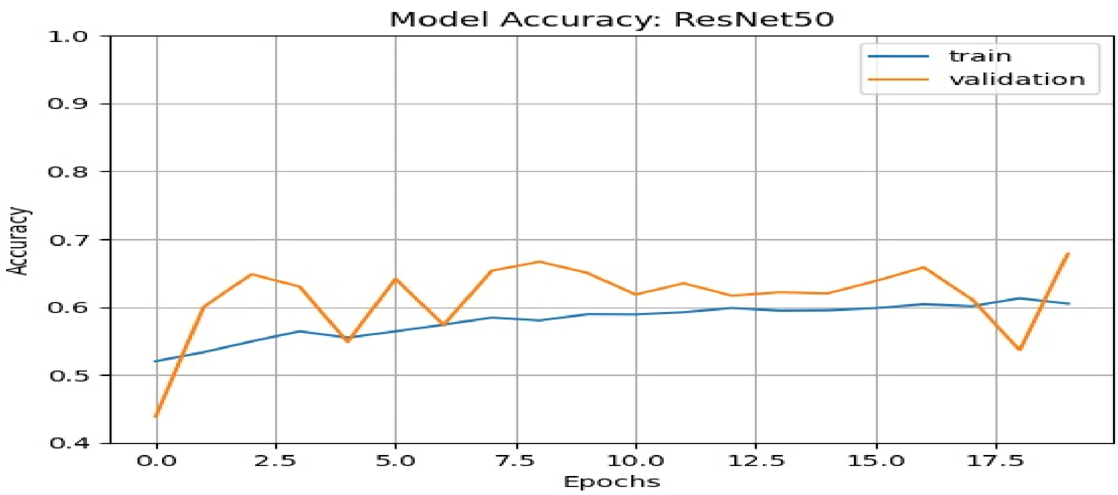


Fig 4.15: Training vs Validation Accuracy: ResNet50

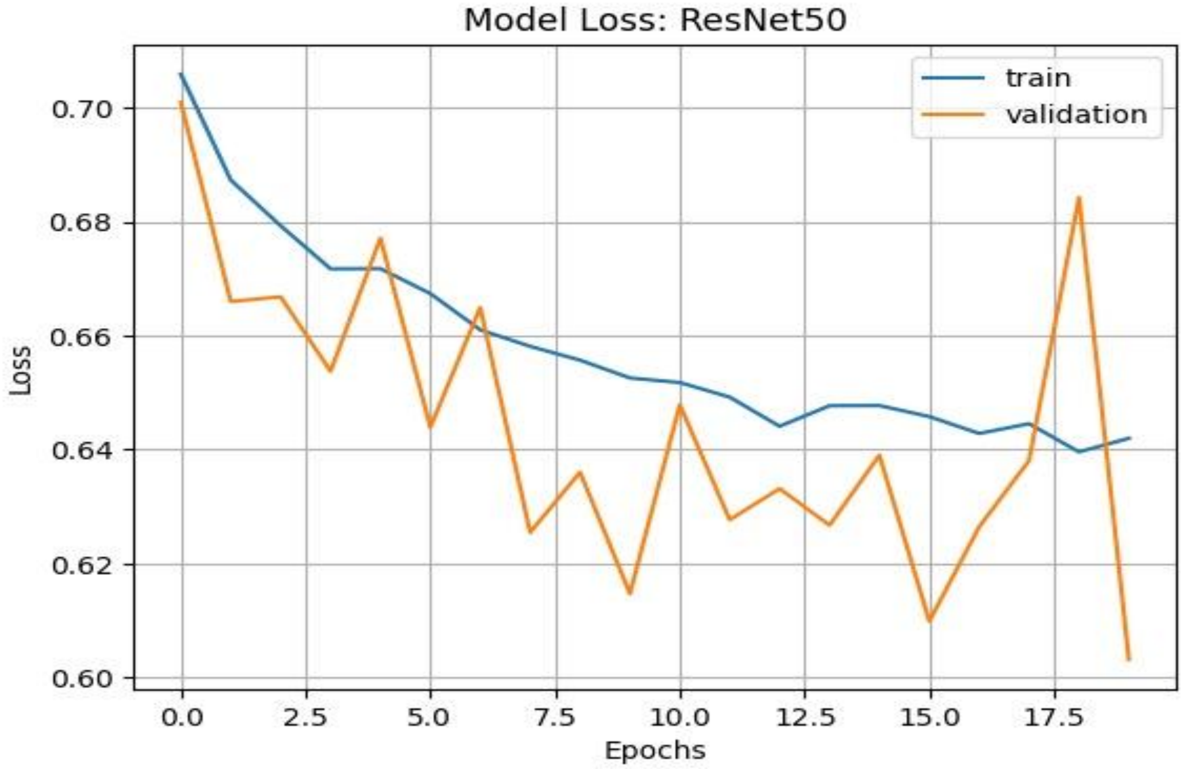


Fig 4.16: Training vs Validation Loss: ResNet50

Accuracy: 0.49

Classification Report:

	precision	recall	f1-score	support
Not Fractured	0.60	0.47	0.52	360
Fractured	0.40	0.53	0.45	240
accuracy			0.49	600
macro avg	0.50	0.50	0.49	600
weighted avg	0.52	0.49	0.50	600

Confusion Matrix:

```
[[168 192]
 [113 127]]
```

Fig 4.17: Test Accuracy, Classification Report: ResNet50

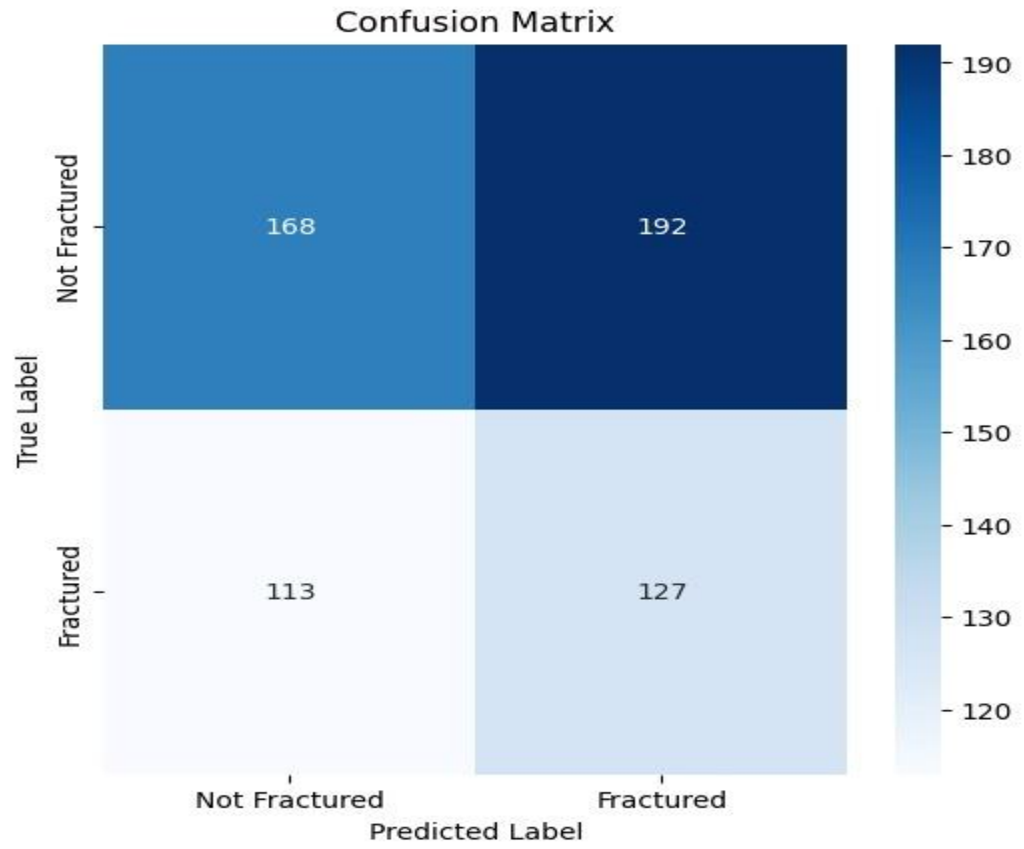


Fig 4.18: Heat Map of Confusion Matrix: ResNet

ResNet's residual connections help address vanishing gradients by allowing layers to skip over one another, a design that aids in capturing high-level features from complex images. This characteristic is advantageous for medical imaging tasks that require deep networks to detect subtle and intricate details.

- **Results Interpretation**: With an accuracy of 0.49, ResNet's performance was slightly below DenseNet and MobileNet. The residual structure helped it maintain stable training across the 20 epochs, but its accuracy suggests room for improvement, potentially through additional data augmentation or fine-tuning specific to fracture patterns. ResNet's residual layers proved beneficial in handling the high image resolution (224x224 pixels), though higher accuracy would be needed for consistent clinical applications.

4.4 Simple GUI Implementation:

The implementation of a Graphical User Interface (GUI) for this project is aimed at simplifying the user interaction with the trained model and making the fracture detection process more accessible. This section outlines the key functionalities of the GUI and their integration.

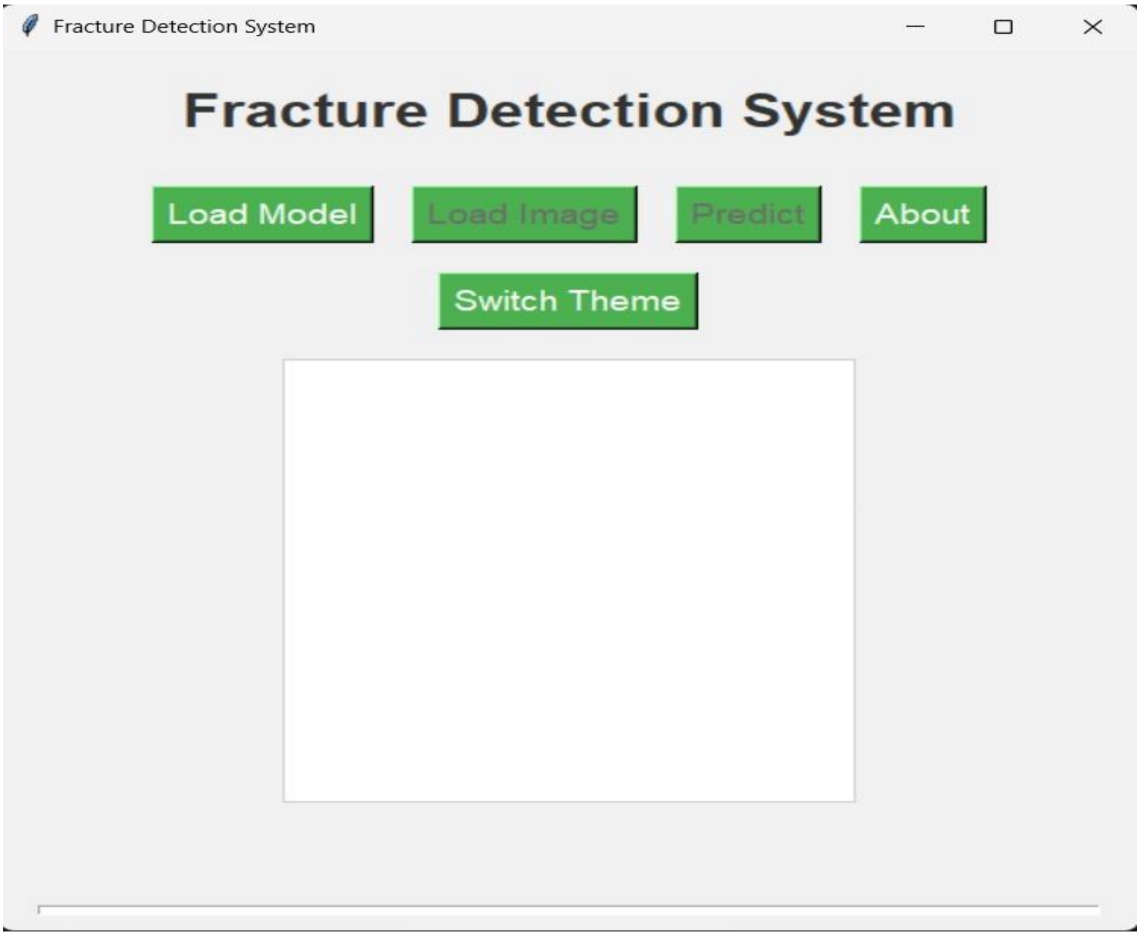


Fig 4.19: Simple GUI

4.4.1 Load Model

This module enables users to select and load the desired pre-trained deep learning model from the system. The GUI provides a dropdown menu or a file selection option, where models like DenseNet121, MobileNet, VGG16, and ResNet50 can be loaded. Once selected, the application verifies the model's compatibility and initializes it for predictions.

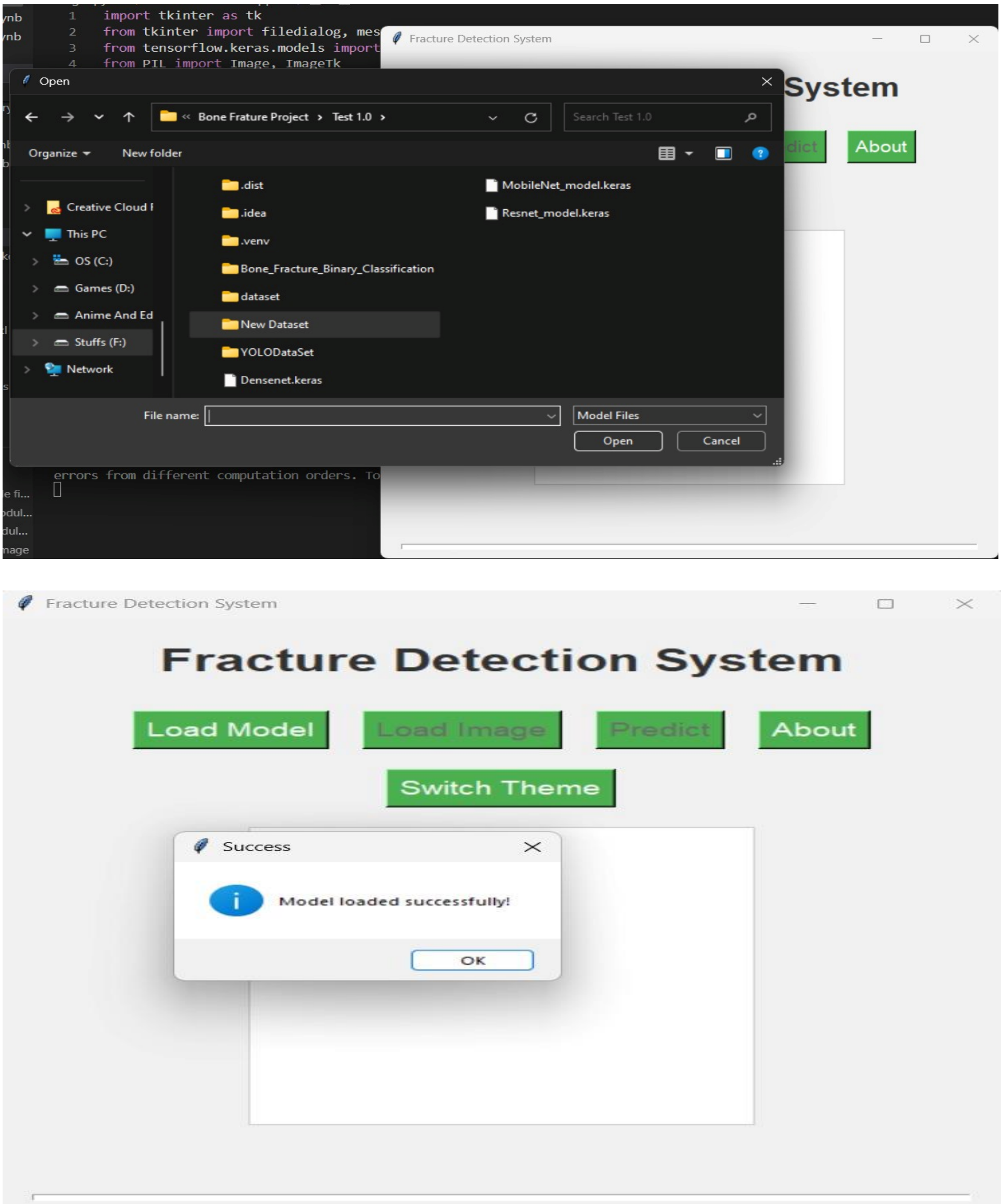
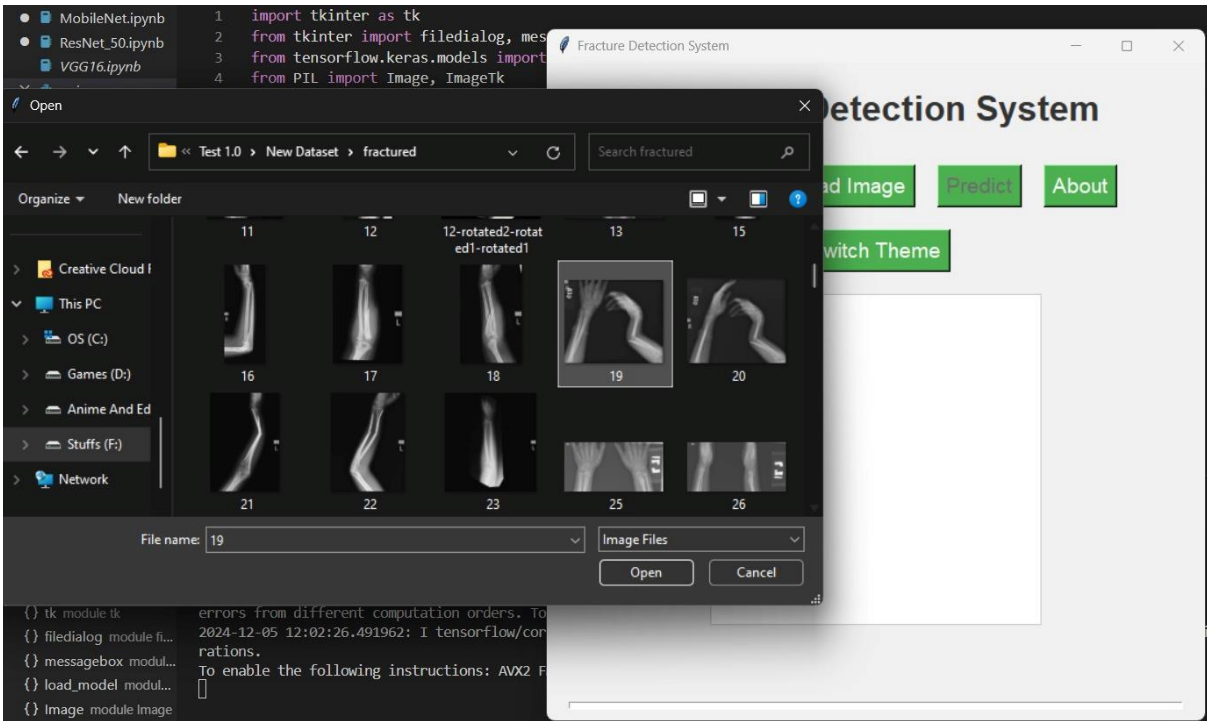


Fig 4.20: Load Model

4.4.2 Load Image

In this section, the GUI allows users to upload medical X-ray images in standard formats such as JPEG or PNG. The interface ensures user-friendly navigation with an upload button and a preview pane that displays the selected image. This step is crucial for ensuring the correct input to the model for accurate fracture detection.



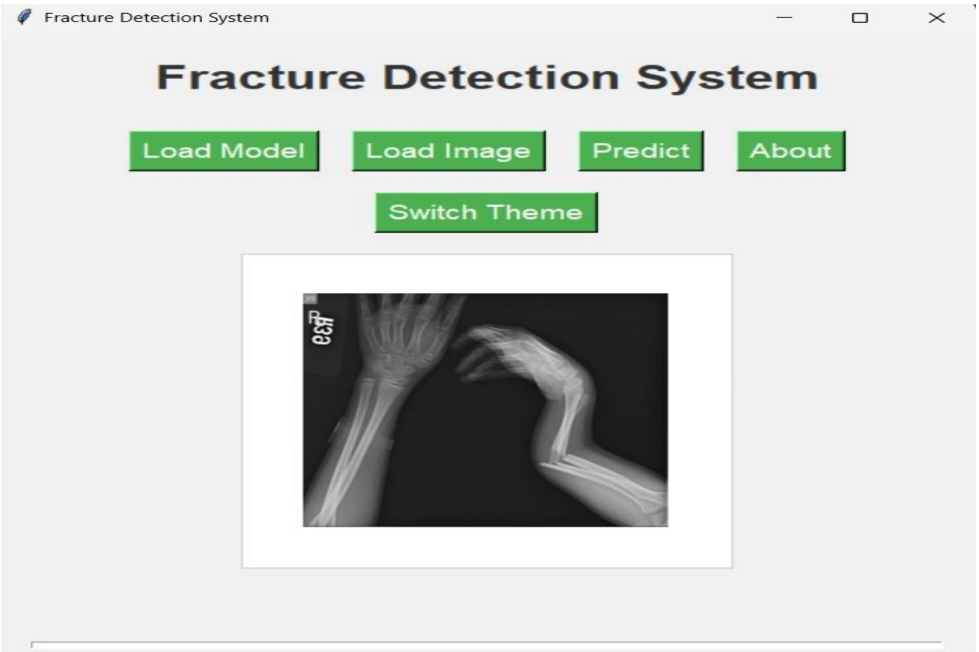


Fig 4.21: Load Image

4.4.3 Predict

Upon loading the model and image, the user can initiate the prediction process using a clearly labeled "Predict" button. The GUI integrates the model's inference capabilities and displays the results, such as fracture detection status (fractured/non-fractured) and corresponding probabilities. Additional visual aids, like heatmaps or localized fracture indicators, may be provided to enhance interpretability.

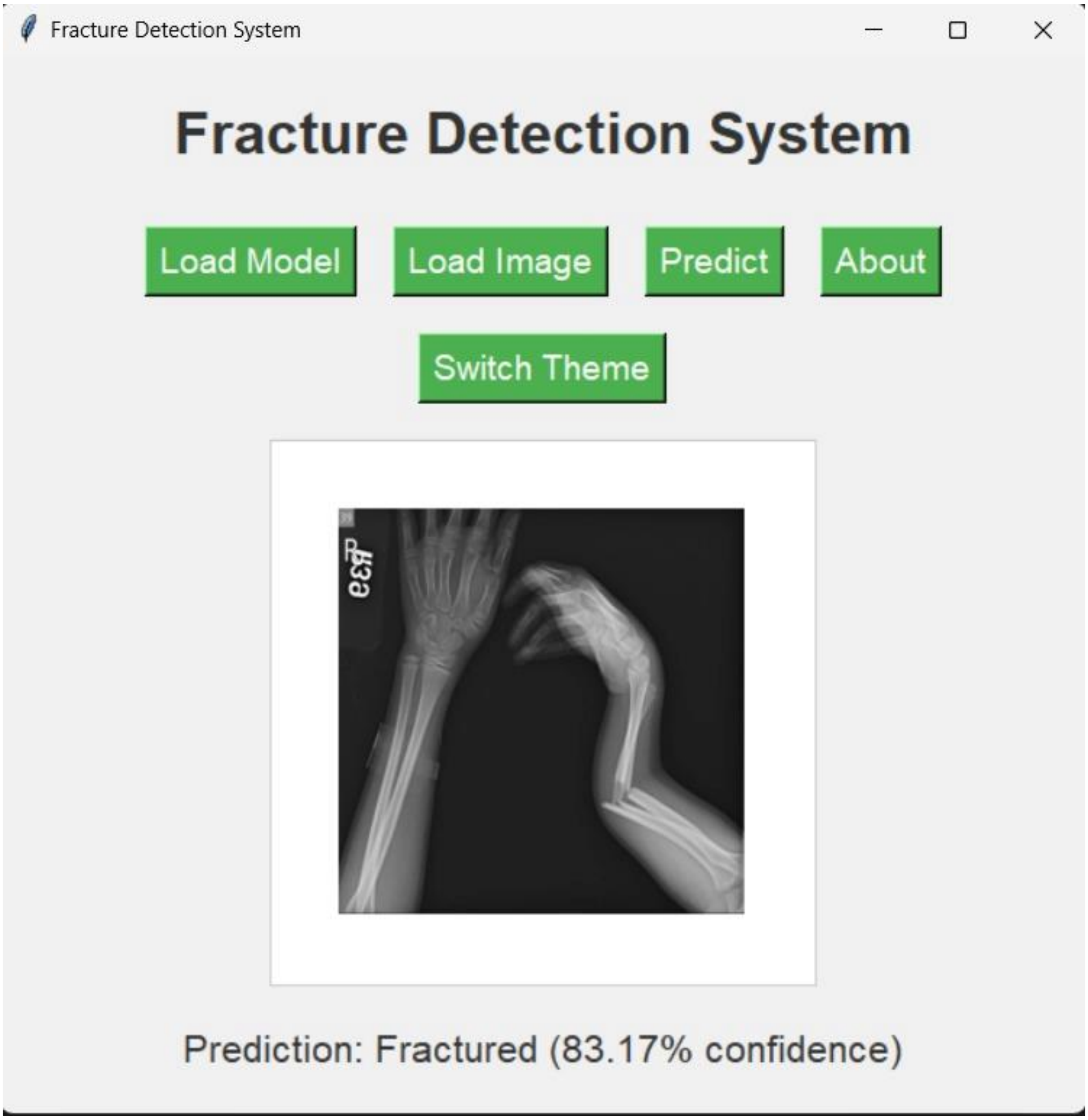


Fig 4.22: Predict

Chapter 5: Comparative Analysis

This report provides an in-depth comparison and interpretation of four CNN models—ResNet, DenseNet, MobileNet, and VGG16—for detecting bone fractures in medical X-ray images. The primary metrics analyzed include test accuracy, F1 score, and classification report, each revealing unique insights into the models' performance and reliability.

5.1 Comparison of Test Accuracy

Test accuracy is a crucial metric as it indicates each model's performance on unseen test data, reflecting its reliability in real-world applications where the models encounter new cases.

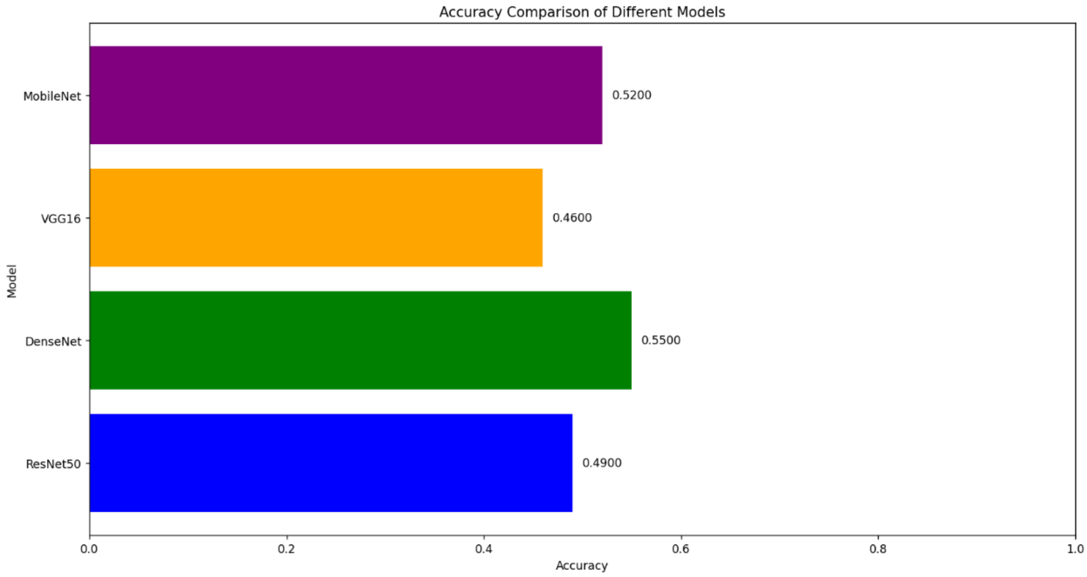


Fig 5.1: Accuracy comparison of different models

- **ResNet**: ResNet achieved a test accuracy of 0.49, which is moderately reliable but not the highest among the models. ResNet's architecture, featuring residual connections, enables it to learn deeper features without degradation, mitigating issues like vanishing gradients. Although it does not outperform DenseNet in test accuracy, ResNet's robust learning of complex patterns makes it a viable candidate for applications requiring depth in feature extraction, though it may benefit from further optimization to match clinical accuracy demands.
- **DenseNet**: DenseNet demonstrated the highest test accuracy at 0.55. Its densely connected layers promote efficient feature reuse across the network, enhancing its capacity to learn and accumulate detailed patterns across the network. This dense connectivity allows DenseNet to excel at capturing subtle indicators of fractures, making it especially effective for medical imaging tasks that demand fine detail. DenseNet's superior test accuracy suggests it generalizes better than the other models in identifying fractures.

- **MobileNet**: MobileNet achieved a test accuracy of 0.52, a respectable result given its lightweight design. MobileNet's use of depthwise separable convolutions minimizes computational costs, which makes it suitable for environments with limited resources, like mobile applications. However, this efficiency comes at the cost of some accuracy, as MobileNet may miss the intricate details necessary for detecting subtle fractures. MobileNet could still be valuable in scenarios where efficiency and computational speed outweigh the need for peak accuracy.

- **VGG16**: VGG16 had the lowest test accuracy at 0.46, suggesting it may struggle with more challenging fracture detection tasks. VGG16's architecture, featuring uniform 3x3 convolutional layers, does provide basic feature extraction capabilities, but it lacks the advanced connectivity of other models. This limitation can hinder its ability to retain crucial details, impacting its accuracy. For clinical applications, VGG16 may require additional fine-tuning to enhance its ability to detect fractures with reliable accuracy.

  Interpretation: DenseNet and ResNet demonstrated stronger test accuracy, making them better suited for clinical applications that demand accurate predictions on unseen cases. MobileNet, while slightly less accurate, provides a balance between performance and efficiency, making it ideal for low-resource environments. VGG16's lower test accuracy indicates it may need further training or architecture adjustments to be considered reliable for fracture detection.

5.2 Comparison of F1 Score

The F1 score balances precision and recall, essential in clinical settings where both false positives (misclassifying healthy tissue as fractured) and false negatives (missing actual fractures) have significant consequences.

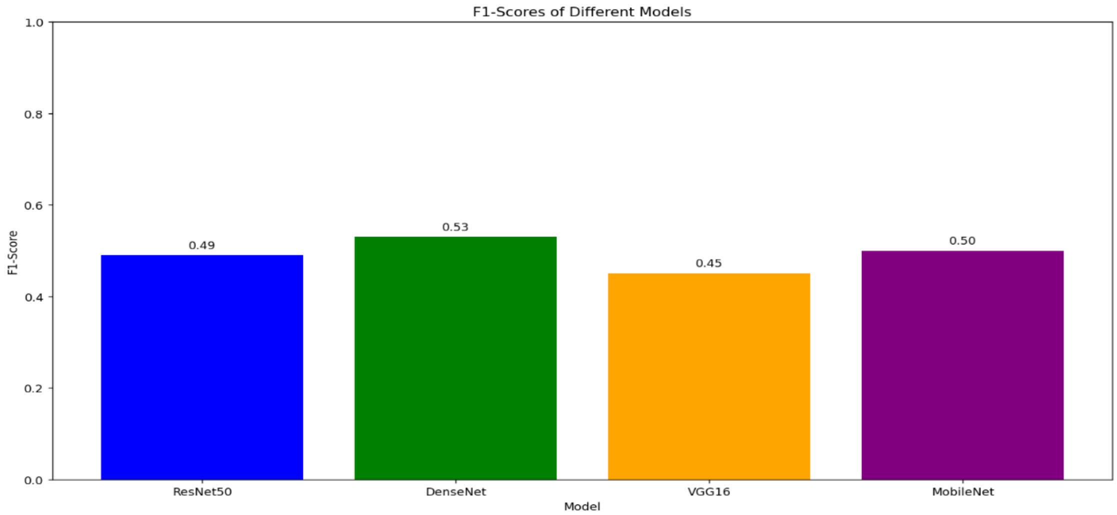


Fig 5.2: F-1 Scores of Different Models

- **ResNet**: ResNet's F1 score of 0.49 indicates a balanced yet moderate performance in precision and recall. The model's residual connections enable it to capture complex patterns, though it may slightly favor precision over recall, potentially missing some fractures. ResNet's balance between precision and recall makes it suitable for applications where avoiding misdiagnosis is critical, though its F1 score suggests room for improvement to reduce false negatives in cases with subtle fracture indicators.

- **DenseNet**: DenseNet achieved the highest F1 score at 0.53, suggesting an optimal balance between precision and recall. Its densely connected architecture enables it to capture and reuse features effectively, supporting high recall and reducing missed fractures. DenseNet's strength in both sensitivity (recall) and specificity (precision) makes it particularly valuable in clinical environments where accuracy is paramount, minimizing the risk of missed fractures.

- **MobileNet**: MobileNet's F1 score of 0.50 is relatively balanced but lower than DenseNet's. MobileNet's efficient architecture, designed for low-resource environments, means it may have a limited ability to capture intricate fracture details, affecting recall. However, its precision is adequate, suggesting it is less likely to produce false positives. MobileNet's balance makes it suitable in cases where avoiding false positives is more important than achieving the highest sensitivity.

- **VGG16**: VGG16 scored the lowest in F1 at 0.45, likely due to limitations in capturing complex features. Its simpler architecture can lead to missed fractures (low recall) and misclassifications (low precision). For applications requiring high accuracy and reliability, VGG16 would benefit from modifications or additional regularization to improve its performance in both sensitivity and specificity.

  Interpretation: DenseNet's higher F1 score suggests it is the most balanced model, achieving both high sensitivity and precision, which is critical in clinical applications. ResNet also performs well, making it valuable for cases with a high need for diagnostic reliability. MobileNet's balanced but slightly lower F1 score is sufficient where computational efficiency is prioritized. VGG16, however, may not be suitable for critical clinical use without further optimization due to its lower F1 score.

5.3 Classification Report Analysis

The classification report provides a detailed breakdown of model performance for each class (fractured vs. non-fractured), capturing precision, recall, and F1 scores.

- **ResNet**:[Fig 4.15]
  - **Precision**: ResNet achieved a precision of 0.60 for the "Not Fractured" class and 0.40 for the "Fractured" class, indicating moderate confidence in distinguishing non-fractured cases.
  - **Recall**: With a recall of 0.47 for "Not Fractured" and 0.53 for "Fractured," ResNet shows a slightly better sensitivity for fractured cases.
  - **F1-score**: The F1-scores of 0.52 and 0.45 for "Not Fractured" and "Fractured" respectively, indicate a reasonable but moderate balance between identifying both classes, though ResNet may occasionally miss fractures.

- **DenseNet**:[Fig 4.3]
  - **Precision**: DenseNet's precision was 0.63 for "Not Fractured" and 0.44 for "Fractured," demonstrating a strong ability to correctly identify non-fractured cases.
  - **Recall**: Recall was 0.61 for "Not Fractured" and 0.46 for "Fractured," reflecting a balanced sensitivity that supports reliable fracture detection.
  - **F1-score**: DenseNet achieved F1-scores of 0.62 for "Not Fractured" and 0.45 for "Fractured," reinforcing its capability to balance sensitivity and precision, making it effective in both classes for clinical applications.
- **MobileNet**:[Fig 4.7]
  - **Precision**: MobileNet showed a precision of 0.60 for "Not Fractured" and 0.40 for "Fractured," indicating a moderate rate of accurate non-fracture detections.
  - **Recall**: Recall was 0.62 for "Not Fractured" and 0.38 for "Fractured," suggesting MobileNet may miss some fractures.
  - **F1-score**: The F1-scores were 0.61 for "Not Fractured" and 0.39 for "Fractured," suggesting a trade-off between precision and recall that could affect its sensitivity for fracture detection.
- **VGG16**:[Fig 4.11]
  - **Precision**: VGG16's precision was 0.58 for "Not Fractured" and 0.39 for "Fractured," indicating some difficulty in reliably distinguishing fractures.
  - **Recall**: Recall was low for "Not Fractured" at 0.33 but higher for "Fractured" at 0.65, reflecting a tendency to miss non-fractured cases.
  - **F1-score**: With F1-scores of 0.42 for "Not Fractured" and 0.49 for "Fractured," VGG16 shows limitations in distinguishing both classes accurately, suggesting it may need further tuning for reliable fracture detection.

Interpretation: DenseNet and ResNet exhibit the best balance across precision, recall, and F1 scores. DenseNet's slightly higher recall makes it particularly valuable in clinical scenarios where missed fractures could have severe consequences. ResNet's balanced scores make it reliable across both classes, suitable for applications where diagnostic reliability is essential. MobileNet's moderate performance and balance make it useful in resource-limited applications. VGG16's lower scores highlight the need for additional training or adjustments to meet clinical standards.

5.4 Validation Accuracy and Loss Analysis

Analyzing validation accuracy and loss offers insights into the models' training performance and potential overfitting:

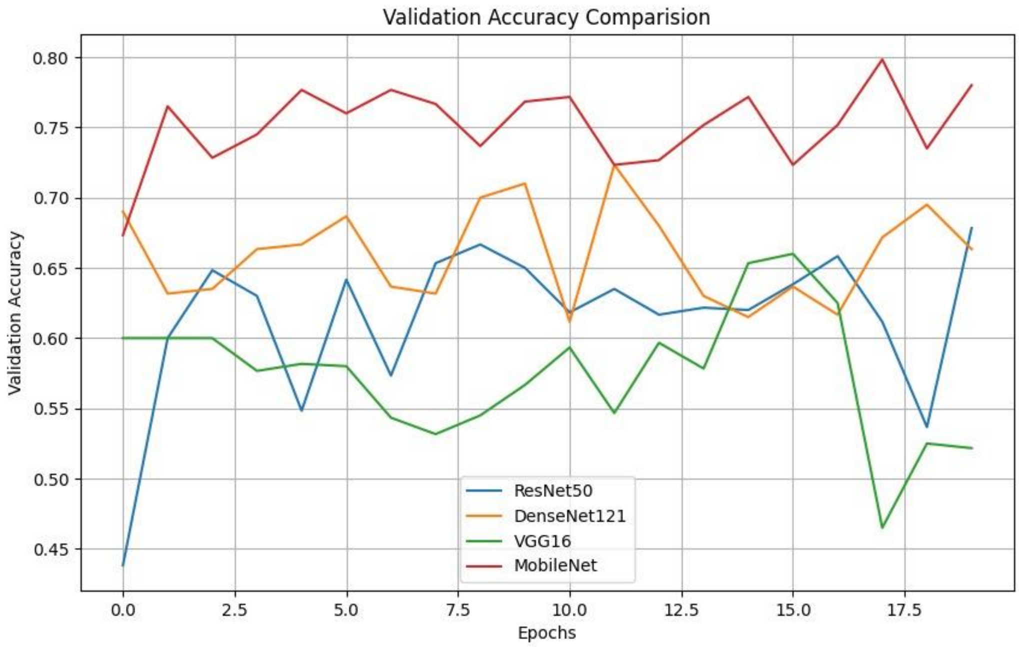


Fig 5.3: Validation Accuracy Comparison

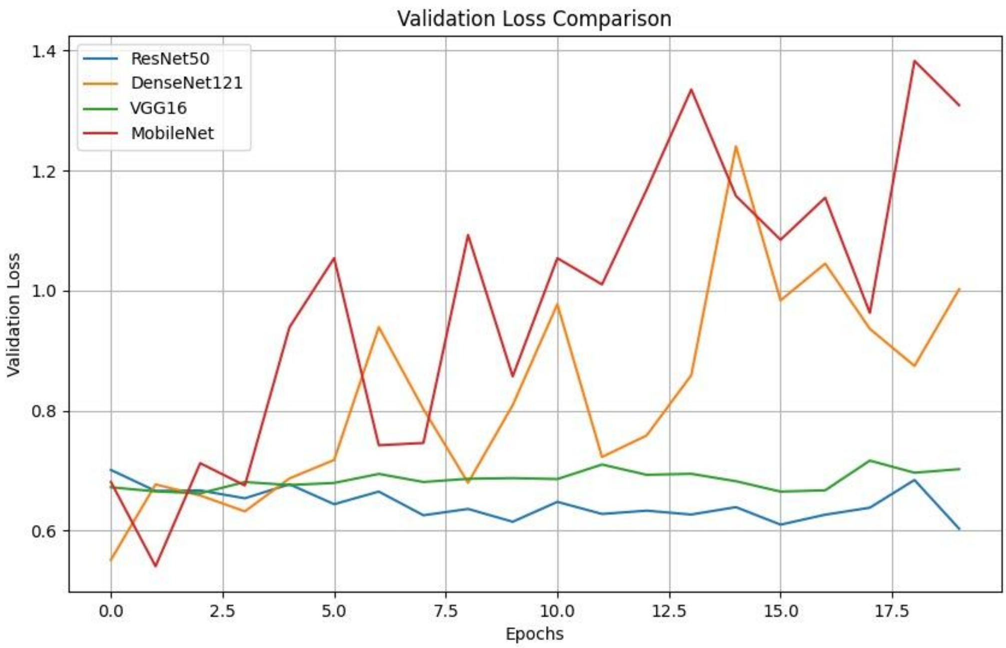


Fig 5.4: Validation Loss Comparison

- **MobileNet** achieved the highest validation accuracy (**0.7800**), but its high validation loss (**1.3088**) indicates significant overfitting, potentially learning noise from the training data.

- **ResNet** showed strong validation accuracy (**0.6783**) with a lower validation loss (**0.6032**), suggesting good generalization without overfitting.
- **DenseNet** maintained high validation accuracy (**0.6633**) but exhibited increased loss (**1.0020**), signaling possible overfitting and sensitivity to training noise.
- **VGG16** presented the lowest validation accuracy (**0.5217**) but a moderate validation loss (**0.7023**), indicating stable but limited learning capability.

  Interpretation: MobileNet's high validation accuracy, paired with elevated loss, suggests it may require regularization techniques like dropout to reduce overfitting. ResNet's balanced performance indicates robust learning and good generalization. DenseNet's performance, while strong, may benefit from further tuning to address potential overfitting. VGG16's lower validation accuracy underscores the need for architectural improvements to enhance its learning capacity.

5.5 Final Insights and Performance Summary

- **DenseNet** emerges as the top performer, ideal for clinical applications demanding high accuracy and balanced precision-recall metrics. Its architecture excels in capturing detailed features, minimizing diagnostic errors.
- **ResNet** provides a reliable alternative with balanced accuracy and stable generalization. It is suited for a variety of diagnostic tasks, offering robust performance with fewer false negatives.
- **MobileNet** is the best choice for efficiency-focused environments, such as mobile or edge devices, where computational resources are limited. Its performance can be enhanced with strategies to reduce overfitting.
- **VGG16** struggles with lower accuracy and high false positives, indicating the need for architectural refinements to enhance its capability in fracture detection tasks.

5.6 Model Recommendations:

- **DenseNet** is recommended for high-accuracy clinical scenarios requiring reliable detection of subtle features.
- **ResNet** is suited for applications needing balanced, consistent performance across diverse datasets.
- **MobileNet** is optimal for real-time, resource-constrained environments but requires additional tuning to mitigate overfitting.
- **VGG16** needs further structural enhancements to improve its precision and generalization, making it less suitable for high-stakes diagnostic use without modifications.

5.7 Addressing Current Limitations

The current test accuracies, with the highest being 0.55 for DenseNet, indicate that these models have not yet reached the level of precision required for clinical deployment. The moderate F1 scores across all models suggest a need for enhanced feature extraction and better handling of class imbalance. Validation losses, especially for MobileNet and DenseNet, point to overfitting issues that need to be addressed to improve generalization.

Key Areas for Improvement:

- **Data Quantity and Quality**: Expanding the dataset and using high-quality, well-labeled X-ray images is vital to enhance the models' ability to generalize effectively.
- **Architecture Refinement**: Exploring more advanced and novel architectures or combining elements of existing models could yield significant improvements in performance.
- **Regularization and Training Techniques**: Implementing more robust regularization strategies and fine-tuning training processes will help mitigate overfitting and improve the models' ability to generalize to new cases.

Chapter 6: Conclusion

This project presents a comprehensive comparative analysis of four deep learning models—DenseNet121, ResNet50, MobileNet, and VGG16—for automated bone fracture detection from X-ray images. The goal was to evaluate these models based on their accuracy, computational efficiency, and overall reliability in identifying fractures, ultimately contributing to AI's growing role in enhancing medical diagnostics.

1. **Model Performance Summary**:
   - **DenseNet121** demonstrated the highest accuracy (0.55) among the models, attributed to its densely connected architecture, which enhances feature reuse and learning efficiency. This makes it a strong candidate for detailed diagnostic tasks requiring the detection of subtle fracture patterns.
   - **ResNet50** provided balanced performance (accuracy of 0.49) due to its residual connections, which mitigated the vanishing gradient problem and supported stable training. Its robust feature extraction capabilities make it suitable for detecting complex and less obvious fractures.
   - **MobileNet**, with an accuracy of 0.52, showed promise as a lightweight and efficient model. Its design is well-suited for real-time diagnostic applications, particularly in settings with limited computational resources, highlighting its potential for deployment in rural clinics or on mobile devices.
   - **VGG16** recorded the lowest accuracy (0.46), indicating limitations in its feature extraction capabilities for this specific task. However, its simpler architecture might be beneficial for straightforward cases or in scenarios where computational efficiency is a higher priority than peak accuracy.

2. **Clinical Implications**:
   - The deployment of AI models like DenseNet and ResNet in clinical settings could revolutionize fracture detection by providing rapid, consistent, and accurate assessments. These models can serve as powerful decision-support tools, aiding radiologists in high-stress environments where diagnostic errors are more likely.
   - MobileNet's efficiency offers a scalable solution for improving diagnostic capabilities in underserved areas, where access to specialized radiologists is limited. Its integration into mobile or cloud-based applications could democratize healthcare, providing timely and accurate diagnoses to patients who might otherwise experience delays.

3. **Challenges and Areas for Improvement**:
    - The study identified moderate accuracy levels across all models, indicating room for improvement before these systems can be fully adopted in clinical practice. Enhancing dataset quality, increasing the variety of fracture types, and employing advanced techniques like data augmentation and transfer learning could help address current limitations.
    - Future research should explore newer architectures, such as transformer-based models or hybrid approaches, to capture complex patterns more effectively. Additionally, refining model interpretability will be essential to gaining radiologists' trust and ensuring seamless integration into diagnostic workflows.

4. **Ethical Considerations and Future Directions**:
    - As AI systems are integrated into medical diagnostics, addressing ethical concerns such as data privacy, bias, and compliance with healthcare regulations will be crucial. Ensuring that AI tools complement rather than replace human expertise will help build trust and acceptance among healthcare providers.
    - Expanding this comparative analysis to include multiclass classification of fracture types and incorporating newer, cutting-edge models will provide deeper insights and improve the system's robustness. The continued development of these technologies promises significant advancements in the field of medical diagnostics, potentially transforming the landscape of patient care.

Final Remarks: The comparative analysis in this project offers valuable insights into the capabilities and limitations of current deep learning models for bone fracture detection. While the findings demonstrate the potential of these models to enhance diagnostic accuracy and efficiency, they also underscore the need for further research and optimization. As AI continues to evolve, its integration into clinical workflows will likely become a standard practice, offering enhanced decision-making tools that support healthcare professionals in delivering timely, accurate, and effective patient care.

References

[1] Detection of bone fracture based on machine learning techniques. https://www.sciencedirect.com/science/article/pii/S2665917423000594

[2] The Role of Artificial Intelligence in the Identification and Evaluation of Bone Fractures. https://www.mdpi.com/2306-5354/11/4/338

[3] CNN BASED BONE FRACTURE DETECTION FOR MEDICAL IMAGING USING RESNET-50. https://ijtrs.com/uploaded_paper/CNN%20BASED%20BONE%20FRACTURE%20DETECTION%20FOR%20MEDICAL%20IMAGING%20USING%20RESNET-50.pdf

[4] Bone Fracture Detection and Classification using Deep Learning Approach. https://www.researchgate.net/publication/341248451_Bone_Fracture_Detection_and_Classification_using_Deep_Learning_Approach

[5] Comparative Analysis of Deep Learning Models for Fracture Detection and Classification in X-ray Images. https://www.naturalspublishing.com/download.asp?ArtcID=28776

[6] Artificial intelligence diagnostic model for multi-site fracture X-ray images of extremities based on deep convolutional neural networks. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10895109/

[7] Artificial Intelligence for X-ray scaphoid fracture detection: a systematic review and diagnostic test accuracy meta-analysis. https://link.springer.com/article/10.1007/s00330-023-10473-x

[8] Automated detection of vertebral fractures from X-ray images: A novel machine learning model and survey of the field lhttps://www.sciencedirect.com/science/article/abs/pii/S092523122301069X

[9] Bone Fracture Detection Using Deep Supervised Learning from Radiological Images: A Paradigm Shift. https://www.mdpi.com/2075-4418/12/10/2420

[10] Automatic Detection and Classification of Rib Fractures on Thoracic CT Using Convolutional Neural Network: Accuracy and Feasibility. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7289688/

[11] The Dataset Drive link: https://drive.google.com/drive/folders/1YBV2qJIY3v-qRNyjuVu3lqir4qbvp4-0?usp=drive_link

[12] The Google Drive link for source code of Densenet121: https://drive.google.com/file/d/1jv05PiHPgxLb9n0Sw7CmNJZGQpGS2oIk/view?usp=sharing

[13] The Google Drive link for source code of MobileNet: https://drive.google.com/file/d/1-WuxoIgofyYf2i3Y6D5pjkAjWyM-Kvge/view?usp=sharing

[14] The Google Drive link for source code of VGG16: https://drive.google.com/file/d/1NobwisYVqYN__XGNimdf72OY4F_jYSpQ/view?usp=sharing

[15] The Google Drive link for source code of ResNet50: https://drive.google.com/file/d/1DNsuKOf-TVhoiYMVjV7xZXQXJNzkNr40/view?usp=sharing