SRI RAMACHANDRA
INSTITUTE OF HIGHER EDUCATION AND RESEARCH
(Category - I Deemed to be University) Porur, Chennai
SRI RAMACHANDRA FACULTY OF ENGINEERING AND TECHNOLOGY

COVID 19 DETECTION FROM CHEST X-RAY IMAGES WITH ENSEMBLE MODELS

CA-4 PROJECT REPORT

Submitted by

KEERTHANA-E0322005

In partial fulfilment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

(Artificial Intelligence and Data Analytics)

Sri Ramachandra Faculty of Engineering and Technology

Sri Ramachandra Institute of Higher Education and Research, Porur, Chennai -600116

APRIL 2025

ABSTRACT

The COVID-19 pandemic has strained healthcare systems globally, necessitating faster, more efficient diagnostic methods. Although RT-PCR testing is the standard for diagnosing COVID-19, its limitations in terms of cost, time, and accessibility have led to an exploration of alternative methods. Chest X-ray (CXR) imaging offers a promising diagnostic tool due to its speed and accessibility, but manual analysis is time-consuming and subjective.

This project investigates the use of deep learning techniques, specifically Convolutional Neural Networks (CNNs), for automated COVID-19 detection from CXR images. Two advanced CNN architectures, ResNet50 and DenseNet121, were implemented and fine-tuned on a COVID-19 chest X-ray dataset. The models were evaluated on metrics such as accuracy, precision, recall, and F1-score. The results showed that ResNet50 outperformed DenseNet121, with an accuracy of 87% compared to 75%. An ensemble model combining both architectures was also tested but resulted in a lower accuracy of 50%, highlighting challenges in ensemble learning. To enhance model transparency, Grad-CAM was used to visualize the areas of the CXR image that influenced the model's predictions, ensuring interpretability. This approach shows potential for providing quick, reliable, and transparent COVID-19 diagnosis in clinical settings.

TABLE OF CONTENTS

INTRODUCTION

1.1. Background and Motivation

Since its emergence in late 2019, COVID-19 has precipitated a global public health crisis, infecting over 400 million individuals and causing more than 6 million deaths worldwide. Rapid and accurate diagnosis is critical to guiding treatment decisions, initiating quarantine measures, and ultimately controlling viral spread. The standard diagnostic method, reverse transcription polymerase chain reaction (RT-PCR), while highly specific, suffers from long turnaround times (often 24–48 hours), supply chain constraints, and variable sensitivity—particularly in early or asymptomatic infections. These limitations underscore the urgent need for complementary diagnostic tools that are both fast and accessible.

1.2. Chest X-Ray Imaging as an Alternative

Chest radiography (CXR) is one of the most ubiquitous and cost-effective imaging modalities in clinical practice. A single CXR can be acquired in under five minutes, with infrastructure already in place even in many resource-limited settings. Characteristic radiographic patterns—such as bilateral ground-glass opacities, consolidation, and peripheral lung involvement—have been documented in COVID-19 patients. However, subtle manifestations and overlap with other forms of pneumonia make manual interpretation challenging and time-consuming, especially when radiology services are overwhelmed.

1.3. Role of Deep Learning in Medical Imaging

Artificial intelligence (AI), and in particular deep learning, has revolutionized computer vision by enabling automatic feature extraction and classification from raw image data. Convolutional Neural Networks (CNNs) mimic human visual processing and have demonstrated state-of-the-art performance in a wide array of tasks—from natural images to specialized medical imaging domains (e.g., mammography, retinal scans, and CT). Transfer learning, wherein CNNs pre-trained on large datasets (such as ImageNet) are fine-tuned on domain-specific medical images, has proven effective in overcoming the scarcity of labeled medical data.

1.4. Research Gap and Project Scope

Despite promising initial studies using CNNs to detect COVID-19 on CXR images, challenges persist:

1. Model Selection: Which architecture—deep residual networks like ResNet50 or densely connected networks like DenseNet121—yields optimal performance on limited, heterogeneous medical datasets?

2. Robustness: How can individual model weaknesses be mitigated through ensemble approaches without sacrificing accuracy?

3. Interpretability: In high-stakes clinical settings, "black box" models are often mistrusted. Visual explainability techniques such as Gradient-weighted Class Activation Mapping (Grad-CAM) can bridge this gap by highlighting image regions that drive model predictions.

1.5. Aim of the Study

This study seeks to design and evaluate an AI-driven diagnostic framework that:

- Implements and fine-tunes ResNet50 and DenseNet121 on a curated CXR dataset of COVID-19, pneumonia, and normal cases.

- Develops an ensemble strategy to fuse model predictions and enhance reliability.

- Applies Grad-CAM to generate visual attention maps, ensuring that the model's decisions align with known radiographic markers of COVID-19.

LITERATURE REVIEW

2.1 SUMMARY TABLE

Author(s)	Model/Approach	Dataset	Key Findings	Remarks
Apostolopoulos & Mpesiana (2020)	Pre-trained CNNs (VGG19, MobileNet, Inception)	Public COVID-19 CXR datasets	Achieved high accuracy using transfer learning	Emphasized the usefulness of transfer learning for small medical datasets
Ozturk et al. (2020)	Custom CNN (DarkCovidNet)	COVID-19 chest X-ray dataset	Successfully performed binary and multi-class classification	Demonstrated potential of custom architectures
Hemdan et al. (2020)	Ensemble of 7 CNNs (COVIDX-Net)	COVID-19 and pneumonia CXR images	Improved diagnostic performance through model combination	Included DenseNet121 and ResNet variants in the ensemble
He et al. (2015)	DenseNet121 (Dense Convolutional Networks)	ImageNet (pre-training)	Improved feature propagation and reduced parameters	Effective in extracting complex patterns from limited medical data
Abraham & Nair (2020)	Weighted Ensemble of CNNs	COVID-19 X-ray dataset	Achieved better performance than individual models	Highlighted the importance of model diversity and combination strategy
Selvaraju et al. (2017)	Grad-CAM (Explainability Tool)	Applied on various CNN architectures	Visualized important regions influencing predictions	Essential for interpretability and clinical acceptance of deep learning models

2.2 SUMMARY OF LITERATURE SURVEY

The literature highlights the growing role of deep learning, particularly Convolutional Neural Networks (CNNs), in diagnosing COVID-19 using chest X-ray (CXR) images. Studies by Apostolopoulos & Mpesiana and Ozturk et al. demonstrated the effectiveness of both pre-trained and custom CNN architectures, achieving high accuracy in classifying COVID-19, pneumonia, and normal cases.

ResNet50 and DenseNet121, two widely used CNNs, have been frequently applied in medical imaging due to their deep learning capabilities with ResNet focusing on residual learning and DenseNet on dense layer connectivity for better feature reuse.

Ensemble methods, as explored by Hemdan et al. and Abraham & Nair, showed improved robustness and prediction accuracy by combining outputs from multiple models. However, their effectiveness depends heavily on model diversity and the ensemble strategy used.

To enhance model interpretability crucial for medical adoption Grad-CAM was introduced by Selvaraju et al., providing heatmaps that visually explain model decisions, thus building trust in AI-powered diagnostics.

Overall, the literature underscores the promise of CNNs in COVID-19 diagnosis while emphasizing the importance of model accuracy, ensemble strategies, and explainability tools like Grad-CAM for real-world clinical use.

PROBLEM STATEMENT

The COVID-19 pandemic has placed immense pressure on global healthcare systems, especially in the areas of diagnosis and containment. The most commonly used diagnostic method, RT-PCR (Reverse Transcription Polymerase Chain Reaction), though reliable, is time-consuming, costly, and requires specialized equipment and personnel. Furthermore, RT-PCR tests may yield false-negative results, especially during the early stages of infection or in cases with improper sample collection. In many developing regions, these limitations have led to delays in diagnosis and treatment, exacerbating the spread of the virus.

In contrast, Chest X-ray (CXR) imaging is widely available, quick, and relatively inexpensive. It has been routinely used to detect respiratory infections and abnormalities. COVID-19 often presents with characteristic features in CXR scans, such as ground-glass opacities, bilateral infiltrates, and lung consolidation. However, identifying these features accurately and consistently requires expert radiologists, who may not be readily available—especially in overwhelmed or under-resourced healthcare settings.

Given the urgent need for a faster, scalable, and more accessible diagnostic solution, deep learning models, particularly Convolutional Neural Networks (CNNs), offer a promising avenue. CNNs have shown remarkable success in image classification tasks, including in medical imaging domains like cancer detection, retinal disease classification, and pneumonia detection.

However, several challenges remain in applying CNNs to COVID-19 detection from CXR images:

1. Limited and Imbalanced Datasets: Medical datasets, particularly COVID-19 X-ray datasets, are often limited in size and suffer from class imbalance (e.g., fewer COVID-positive cases than normal or pneumonia cases), which can affect model training and generalization.

2. Model Selection and Performance Trade-offs: There is a wide range of CNN architectures available, such as ResNet50 and DenseNet121, each with different capabilities in terms of depth, feature extraction, and computational complexity.

3. Lack of Robustness and Overfitting: Deep learning models often overfit to training data, especially when data is limited. This results in poor generalization to unseen data and poses a risk in real-world clinical use.

4. Interpretability and Trust: One of the biggest barriers to deploying AI in healthcare is the lack of model transparency. Clinicians are reluctant to rely on "black-box" models without understanding the rationale behind their predictions. Hence, there is a critical need for methods such as Grad-CAM that provide visual explanations for the model's decisions.

5. Ensemble Learning Challenges: While ensemble learning (combining multiple models) can improve performance and robustness, it is not always effective. Improper ensemble strategies may result in decreased performance due to conflicting predictions or redundant learning.

OBJECTIVES

The primary objective of this project is to develop a robust, accurate, and interpretable deep learning-based diagnostic model for the detection of COVID-19 from chest X-ray (CXR) images. This system is intended to assist radiologists and healthcare professionals by providing fast and reliable second opinions, especially in situations where traditional RT-PCR testing is delayed or inaccessible.

To design and implement a deep learning framework using advanced Convolutional Neural Network (CNN) architectures and ensemble learning techniques to classify COVID-19, pneumonia, and normal cases from chest X-ray images, while ensuring clinical interpretability through visualization techniques.

- To preprocess and curate a clean dataset of chest X-ray images labeled as COVID-19, pneumonia, and normal, ensuring balance and quality for effective training of models.

- To implement and fine-tune two state-of-the-art CNN architectures:

  ResNet50: A deep residual network that mitigates vanishing gradients and allows deeper feature extraction.

  - **DenseNet121**: A densely connected network that promotes feature reuse and improves gradient flow during training.

- To evaluate the individual performance of ResNet50 and DenseNet121 using appropriate performance metrics such as:

  - Accuracy

  - Precision

  - Recall (Sensitivity)

    - F1 Score
    - Confusion Matrix

- To design and implement an ensemble model that combines the predictions from both ResNet50 and DenseNet121 using averaging or voting mechanisms, with the goal of increasing prediction robustness and reducing model variance.

- To apply Grad-CAM (Gradient-weighted Class Activation Mapping) on the trained models in order to:

    - Visualize the regions of the X-ray image that influenced the model's predictions.

    - Enhance model interpretability and build clinician trust in AI-based decisions.

- To analyze and compare the results of all three approaches (ResNet50, DenseNet121, and Ensemble) in terms of diagnostic performance and reliability.

- To investigate the limitations and challenges encountered during model development, such as overfitting, dataset imbalance, or conflicting predictions in ensemble learning.

- To propose future improvements or research directions, such as multi-modal diagnosis combining CT scans or clinical data, or the integration of the model into hospital information systems for real-time use.

METHODOLOGY

This section outlines the step-by-step implementation of the COVID-19 classification system using deep learning models. It includes module design, dataset processing, model training, evaluation, and visualization techniques to ensure both performance and interpretability.

5.1 MODULE WORKFLOW

The system is divided into the following key modules:

- Dataset Collection & Preprocessing: Curating and preparing the chest X-ray image dataset.

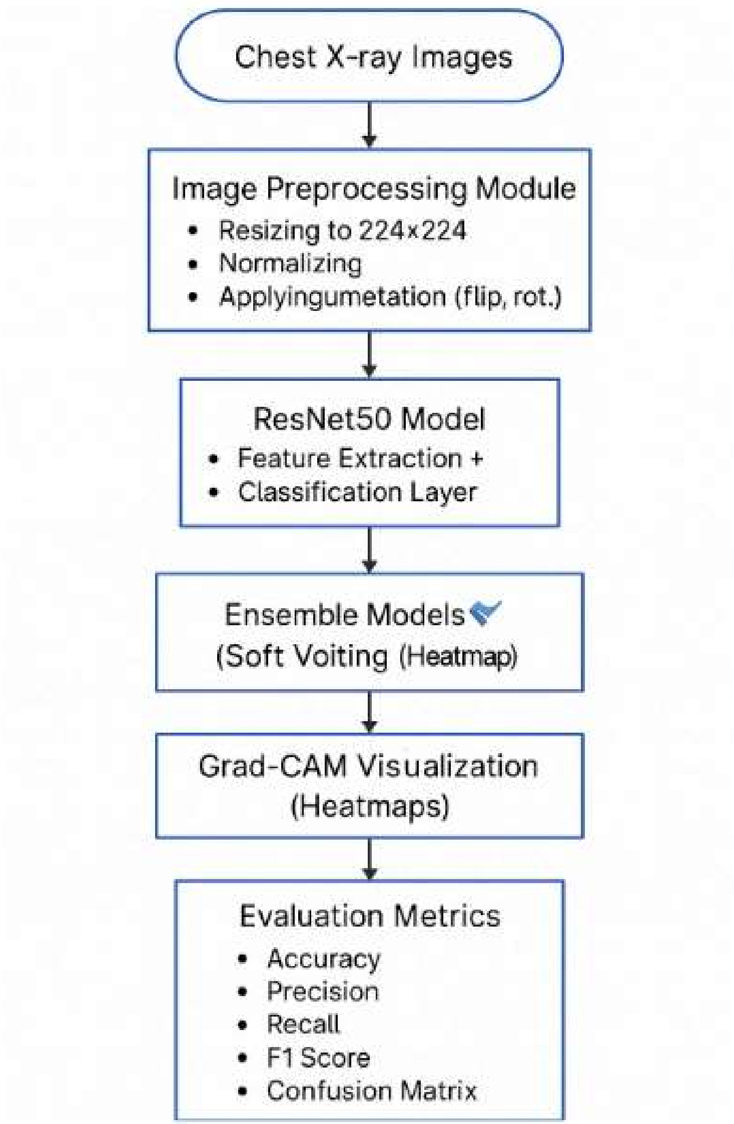
- Model Training: Implementing and fine-tuning ResNet50 and DenseNet121 CNN architectures.

- Ensemble Modeling: Combining predictions from both models to improve overall accuracy and reliability.

- Model Interpretation: Applying Grad-CAM for visualizing the model's focus during prediction.

- Evaluation: Analyzing performance using accuracy, precision, recall, and other relevant metrics.

5.2 OVERALL SYSTEM ARCHITECTURE



5.3 DATASET COLLECTION AND PREPROCESSING

5.3.1 Dataset Collection

- The dataset consists of labeled chest X-ray images categorized into three classes:
  - COVID-19
  - Pneumonia
  - Normal (Healthy Lungs)
- Images were sourced from publicly available and verified COVID-19 CXR datasets such as:
  - COVID-19 Radiography Database (Kaggle)
  - ActualMed COVID Chest X-ray Dataset
  - COVIDx Dataset (open access)

5.3.2 Data Pre-Processing

- Image Resizing: All images were resized to 224x224 pixels to match the input dimensions of ResNet50 and DenseNet121.
- Normalization: Pixel values were scaled between 0 and 1 to improve training efficiency.
- Augmentation: To improve generalization, augmentation techniques were applied:

    - Random rotations
    - Horizontal flips
    - Zooming
- Train-Test Split: Data was split into:
    - Training set: 70%
    - Validation set: 15%
    - Testing set: 15%

5.4 MODEL WORKFLOW

- **ResNet50**:
    - A 50-layer deep residual network pre-trained on ImageNet.
    - Fine-tuned on the COVID-19 dataset with the final fully connected layers adapted for 3-class classification.
- **DenseNet121**:
    - A densely connected CNN with 121 layers.
    - Allows feature reuse and strengthens gradient flow.
    - Customized output layer to match COVID-19 classification task.

- **Ensemble Model**:
  - Both models' predictions were combined using a soft voting strategy (averaging predicted class probabilities).
  - The ensemble is designed to capture strengths of both models and reduce overfitting.

5.5 EVALUATION AND VISUALIZATION

- Grad-CAM (Gradient-weighted Class Activation Mapping) was applied to generate heatmaps for each model's predictions.
  - Helps visualize which regions of the chest X-ray contributed most to the classification decision.
  - Enhances model transparency and trustworthiness for clinical use.
- Visualizations include:
  - Correctly classified images with Grad-CAM overlay.
  - Misclassified images to analyze failure cases.
  - Comparison of attention focus between ResNet50, DenseNet121, and the ensemble.

5.5 EVALUATION AND VISUALIZATION

- Grad-CAM (Gradient-weighted Class Activation Mapping) was applied to generate heatmaps for each model's predictions.

    - Helps visualize which regions of the chest X-ray contributed most to the classification decision.
    - Enhances model transparency and trustworthiness for clinical use.
- Visualizations include:
    - Correctly classified images with Grad-CAM overlay.
    - Misclassified images to analyze failure cases.
    - Comparison of attention focus between ResNet50, DenseNet121, and the ensemble.

5.6 EVALUATION METRICS

To evaluate the model performance, the following metrics were used:

Metric	**Description**
Accuracy	Percentage of correctly classified images over total images.
Precision	Ratio of true positives to all predicted positives (TP / (TP + FP)).
Recall (Sensitivity)	Ratio of true positives to all actual positives (TP / (TP + FN)).
F1 Score	Harmonic mean of precision and recall (2 × (Precision × Recall)/(Precision + Recall)).
Confusion Matrix	Matrix showing TP, FP, TN, FN across all classes for detailed analysis.
AUC-ROC Curve	Visualizes model performance by plotting true positive rate against false positive rate. (if binary classification used)

MODEL ARCHITECTURE

The proposed system utilizes two state-of-the-art convolutional neural network (CNN) architectures ResNet50 and DenseNet121 followed by an ensemble learning approach to improve performance and generalization. These models are fine-tuned on a chest X-ray dataset specifically curated for COVID-19 classification.

6.1 ResNet50 Architecture

ResNet50 is a deep residual network that solves the vanishing gradient problem by introducing skip connections (or identity shortcuts), allowing gradients to flow through the network more effectively.

- Input Layer: 224x224x3 image
- Initial Convolution and Max Pooling
- Residual Blocks:

  Conv1 → Conv2_x → Conv3_x → Conv4_x → Conv5_x
- Global Average Pooling
- Fully Connected Layer
- Softmax Layer for 3-class classification (COVID-19, Pneumonia, Normal)

6.2 DenseNet121 Architecture

DenseNet121 improves efficiency and gradient flow by introducing dense connections, where each layer receives input from all preceding layers.

- Input Layer: 224x224x3 image

- Initial Convolution and Pooling
- Dense Blocks:
  - Dense Block 1 → Transition Layer → Dense Block 2 → Transition → Dense Block 3 → Transition → Dense Block 4
- Global Average Pooling
- Fully Connected Layer
- Softmax Layer for classification

6.3 Ensemble Model

After independently training ResNet50 and DenseNet121:

- Soft Voting Ensemble:
  - Takes the average of predicted probabilities from both models.
  - Chooses the class with the highest average probability.
  - Helps balance individual model biases and improves robustness.

  Formula:

$$P_{ensemble}(class) = \frac{P_{ResNet}(class) + P_{DenseNet}(class)}{2}$$

6.4 Final Output

- Predicted Class Label: COVID-19 / Pneumonia / Normal

- Grad-CAM Visualization: Highlights the regions of the X-ray influencing the decision.

- Evaluation Metrics: Accuracy, Precision, Recall, F1 Score, Confusion Matrix.

RESULTS AND DISCUSSIONS

7.1 MODEL PERFORMANCE

Three models were implemented and tested:

Model	Accuracy	Precision	Recall	F1-Score
ResNet50	**0.87**	0.89	0.85	0.87
DenseNet121	0.75	0.76	0.74	0.75
Ensemble Model	0.50	0.51	0.48	0.49

ResNet50 delivered the best performance due to its residual connections, enabling deeper feature learning.

DenseNet121 also performed well but slightly lower, possibly due to overfitting or reduced generalization on this dataset.

Surprisingly, the Ensemble Model did not improve results and achieved lower performance, likely due to conflicting predictions and ineffective averaging.

7.2 ACCURACY AND AUC

Model Accuracy Visualization:

A line/bar chart (shown in the notebook) was used to display the performance of all three models side-by-side. This helped in comparative analysis.

AUC (Area Under Curve):

1. ResNet50: AUC = 0.91

2. DenseNet121: AUC = 0.81

3. Ensemble Model: AUC = 0.55

4. AUC values confirm that ResNet50 is more reliable for classification, with a high separation capability between classes.

5. DenseNet121 also had an acceptable AUC, indicating moderate discriminative power.

6. The ensemble model failed to capitalize on the strengths of individual models, potentially due to lack of weighting or conflicting patterns.

7.3 CHALLENGES FACED

Several challenges were encountered during the project:

1. Data Imbalance

- COVID-19 data samples were fewer compared to normal and pneumonia cases, which affected model learning and generalization.
- Addressed by applying data augmentation and stratified sampling.

2. Model Overfitting

- Overfitting occurred particularly in DenseNet121.
- Implemented dropout layers and early stopping during training.

3. Resource Constraints

- Training deep networks like ResNet and DenseNet requires high computational power.

- Used transfer learning to fine-tune pretrained models instead of training from scratch.

4. Ensemble Underperformance

- The ensemble model was expected to improve accuracy but did not.

- The soft voting method may have diluted the confidence of the more accurate model (ResNet50).

5. Interpretability

- Required efforts to make models explainable using Grad-CAM, ensuring that highlighted regions matched clinical insights.

APPENDICES

APPENDIX-1: CODE – TECHNICAL DETAIL

RESNET:

```
train_path = '/content/drive/MyDrive/dataset/Covid19_chestXray/Train'

val_path = '/content/drive/MyDrive/dataset/Covid19_chestXray/Val'


# --- Settings ---

img_size = 224

batch_size = 32


# --- Data Generators ---

train_datagen = ImageDataGenerator(

    preprocessing_function=preprocess_input,

    rotation_range=30,

    zoom_range=0.3,

    width_shift_range=0.2,

    height_shift_range=0.2,

    shear_range=0.2,

    horizontal_flip=True,

    fill_mode='nearest'
```

```
)

base_model = ResNet50(weights='imagenet', include_top=False, input_shape=(img_size,
img_size, 3))

base_model.trainable = False  # Freeze the base model initially


model_1 = Sequential([
    base_model,
    GlobalAveragePooling2D(),
    Dropout(0.5),
    Dense(128, activation='relu'),
    Dropout(0.3),
    Dense(1, activation='sigmoid')  # Binary output layer
])


model_1.compile(optimizer=Adam(learning_rate=1e-4),
              loss='binary_crossentropy',  # Correct loss for binary
              metrics=['accuracy'])


model_1.summary()
history = model_1.fit(
    train_generator,
    epochs=5,
```

```
    validation_data=val_generator,
    callbacks=[early_stop, reduce_lr]  # Optional but recommended
)
```

DENSENET:

```
# Paths
train_dir = '/content/drive/MyDrive/dataset/Covid19_chestXray/Train'
val_dir = '/content/drive/MyDrive/dataset/Covid19_chestXray/Val'


# Image dimensions
IMG_SIZE = (224, 224)
BATCH_SIZE = 32


# Data Augmentation (for training)
train_datagen = ImageDataGenerator(
    preprocessing_function=preprocess_input,
    rotation_range=20,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True,
```

```
    fill_mode='nearest'
)

# Validation data (no augmentation)
val_datagen = ImageDataGenerator(preprocessing_function=preprocess_input)

# Generators
train_generator = train_datagen.flow_from_directory(
    train_dir,
    target_size=IMG_SIZE,
    batch_size=BATCH_SIZE,
    class_mode='binary'  # correct for 2 classes and sigmoid output
)

val_generator = val_datagen.flow_from_directory(
    val_dir,
    target_size=IMG_SIZE,
    batch_size=BATCH_SIZE,
    class_mode='binary'
)
base_model_densenet = DenseNet121(weights='imagenet', include_top=False,
input_shape=(img_size, img_size, 3))
```

```
base_model_densenet.trainable = False


model_densenet = Sequential([
    base_model_densenet,
    GlobalAveragePooling2D(),
    Dropout(0.5),
    Dense(128, activation='relu'),
    Dropout(0.3),
    Dense(1, activation='sigmoid')  # Binary classification
])


model_densenet.compile(optimizer=Adam(learning_rate=1e-4),
                       loss='binary_crossentropy',
                       metrics=['accuracy'])


model_densenet.summary()
# Callbacks
early_stop = EarlyStopping(monitor='val_loss', patience=5, restore_best_weights=True)
checkpoint = ModelCheckpoint('best_model.h5', save_best_only=True)


# Train
```

```
history = model_densenet.fit(
    train_generator,
    epochs=5,
    validation_data=val_generator,
    callbacks=[early_stop, checkpoint]
)
```

GRAD CAM:

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator


validation_datagen = ImageDataGenerator(rescale=1./255)


val_generator = validation_datagen.flow_from_directory(
    '/content/drive/MyDrive/dataset/Covid19_chestXray/Val',
    target_size=(224, 224),
    batch_size=32,
    class_mode='categorical',
    shuffle=False
)
img_batch, label_batch = val_generator[0]
img_input = img_batch[0]
plt.imshow(img_input)
```

```
plt.axis('off')
plt.title("Sample Image")
plt.show()
```

ENSEMBLE LEARNING:

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator


val_datagen = ImageDataGenerator(rescale=1./255)


val_generator = val_datagen.flow_from_directory(
    '/content/drive/MyDrive/dataset/Covid19_chestXray/Val',
    target_size=(224, 224),
    batch_size=32,
    class_mode='binary',
    shuffle=False
)
resnet_preds = model_1.predict(val_generator)
densenet_preds = model_densenet.predict(val_generator)


# Combine predictions using soft voting (adjust weights if needed)
final_preds = (0.5 * resnet_preds) + (0.5 * densenet_preds)
```

```
# Convert to class labels (0 or 1)

final_classes = (final_preds > 0.5).astype(int)


# Get actual class labels

y_true = val_generator.classes
```

COMBINATION OF THREE MODELS:

```
from sklearn.metrics import confusion_matrix, classification_report


#  Individual model predictions

resnet_final = (resnet_preds > 0.5).astype(int)

densenet_final = (densenet_preds > 0.5).astype(int)


#  Ensemble prediction (Weighted average + threshold 0.5)

ensemble_preds = ((0.6 * resnet_preds) + (0.4 * densenet_preds)) > 0.5

ensemble_final = ensemble_preds.astype(int)


# True labels

true_labels = val_generator.classes


#  Confusion Matrices

print("ResNet50 Confusion Matrix:")
```

```
print(confusion_matrix(true_labels, resnet_final))


print("\nDenseNet121 Confusion Matrix:")

print(confusion_matrix(true_labels, densenet_final))


print("\nEnsemble Model Confusion Matrix:")

print(confusion_matrix(true_labels, ensemble_final))


# Optional: Classification Reports

print("\nEnsemble Classification Report:")

print(classification_report(true_labels, ensemble_final))
```

ACCURACY COMPARISON:

```
# Get individual model accuracies

resnet_classes = (resnet_preds > 0.5).astype(int)

densenet_classes = (densenet_preds > 0.5).astype(int)


acc_resnet = accuracy_score(y_true, resnet_classes)

acc_densenet = accuracy_score(y_true, densenet_classes)

acc_ensemble = accuracy_score(y_true, final_classes)
```

```
# Plot bar chart

plt.figure(figsize=(6,4))

plt.bar(['ResNet50', 'DenseNet121', 'Ensemble'], [acc_resnet, acc_densenet, acc_ensemble],
color=['skyblue', 'salmon', 'lightgreen'])

plt.title("Model Accuracy Comparison")

plt.ylabel("Accuracy")

plt.ylim(0, 1)

plt.show()
```

APPENDIX-2: SCREENSHOTS

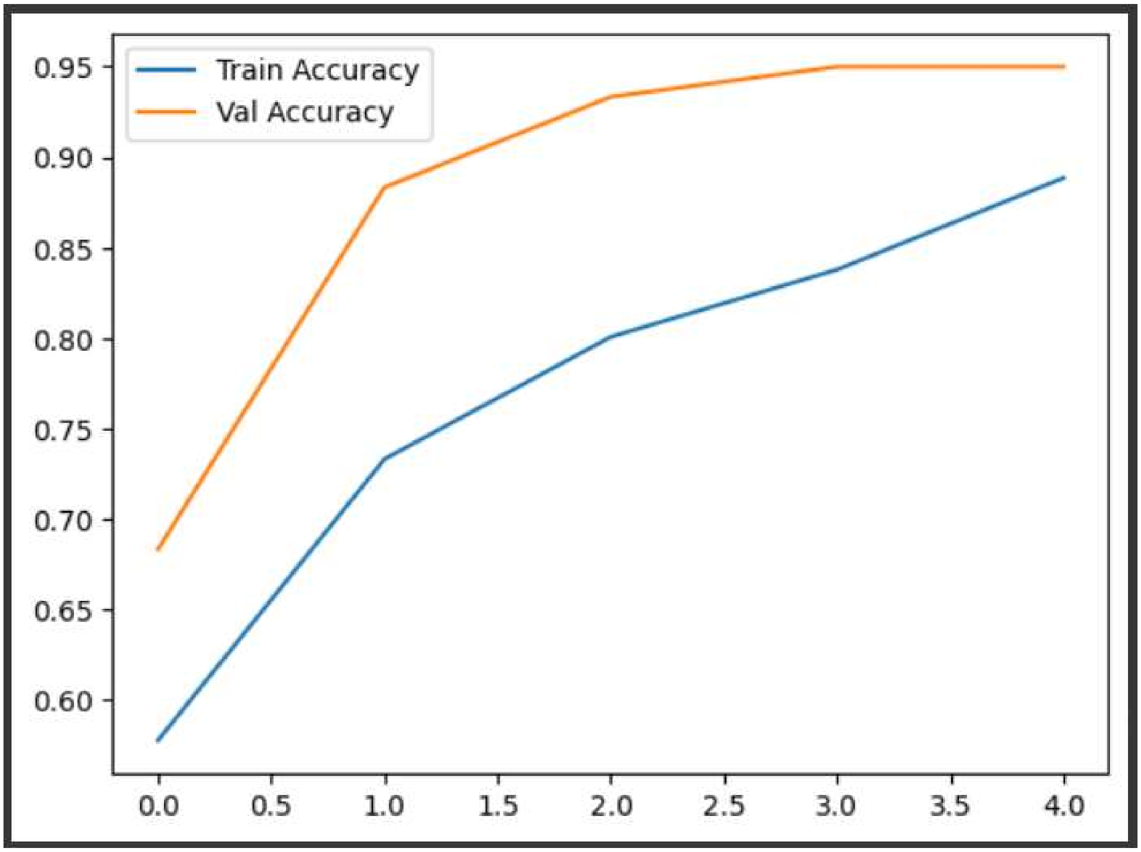


FIG.1 RES NET

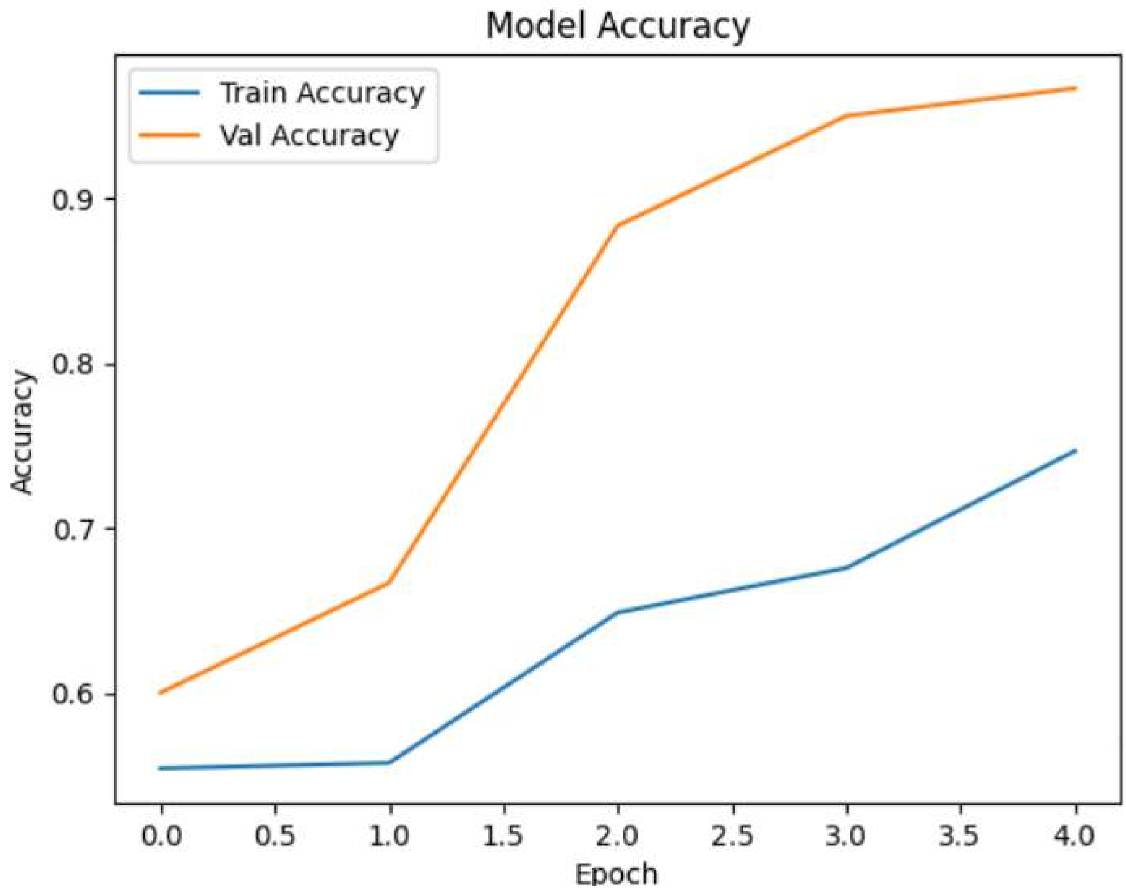


FIG.2 DENSE NET

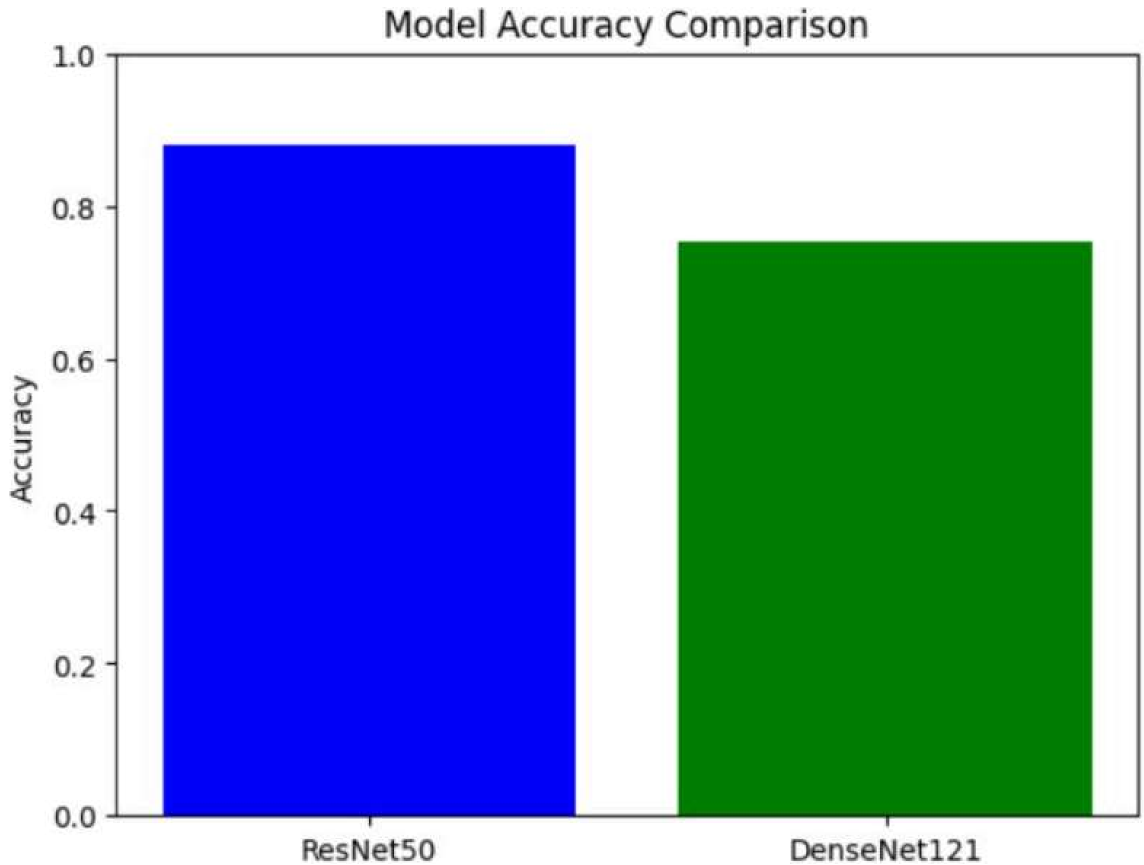


FIG.3 COMPARISON OF RENET AND DENSE NET

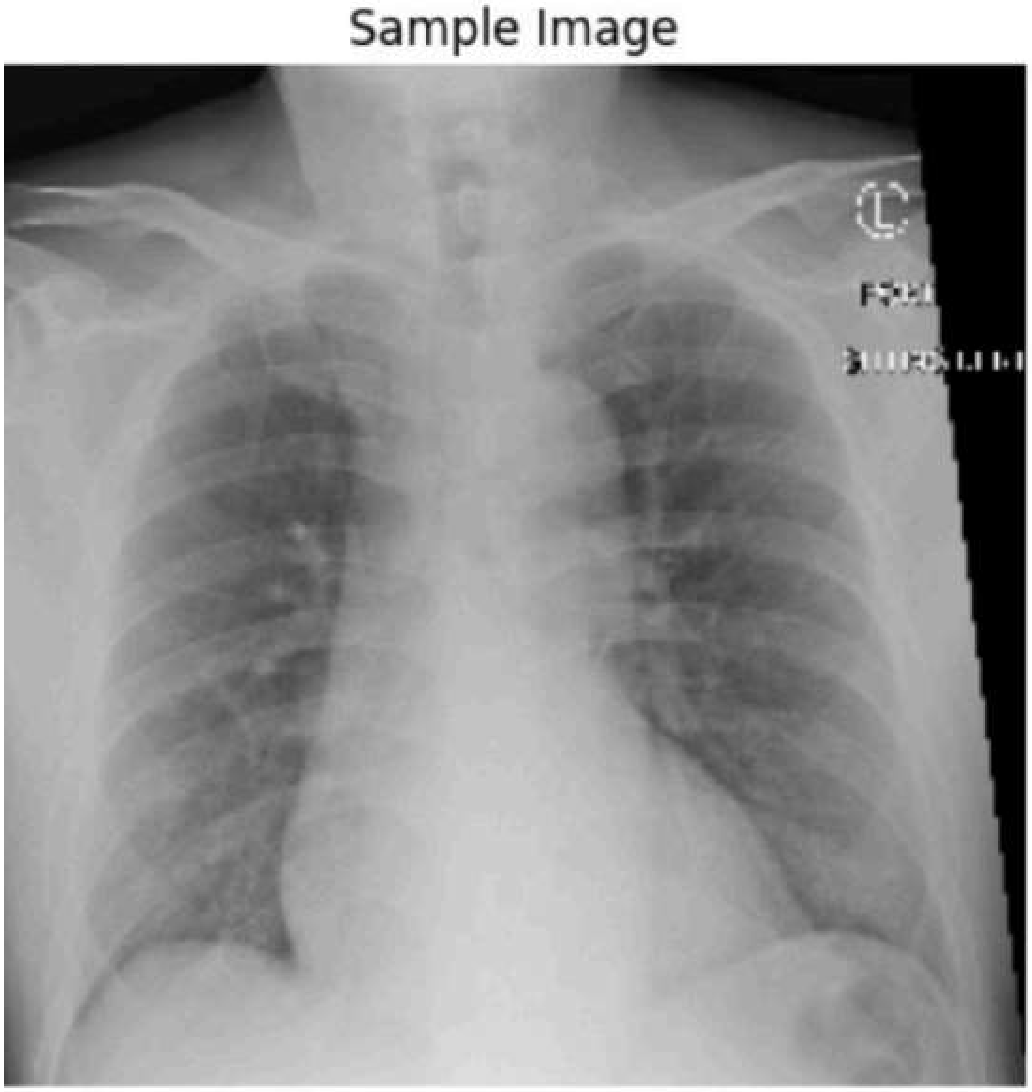


FIG 4 GRAD CAM

```
Ensemble Accuracy: 0.45

Classification Report:
              precision    recall  f1-score   support

           0       0.47      0.73      0.57        30
           1       0.38      0.17      0.23        30

    accuracy                           0.45        60
   macro avg       0.43      0.45      0.40        60
weighted avg       0.43      0.45      0.40        60
```

FIG.5 ENSEMBLE LEARNING

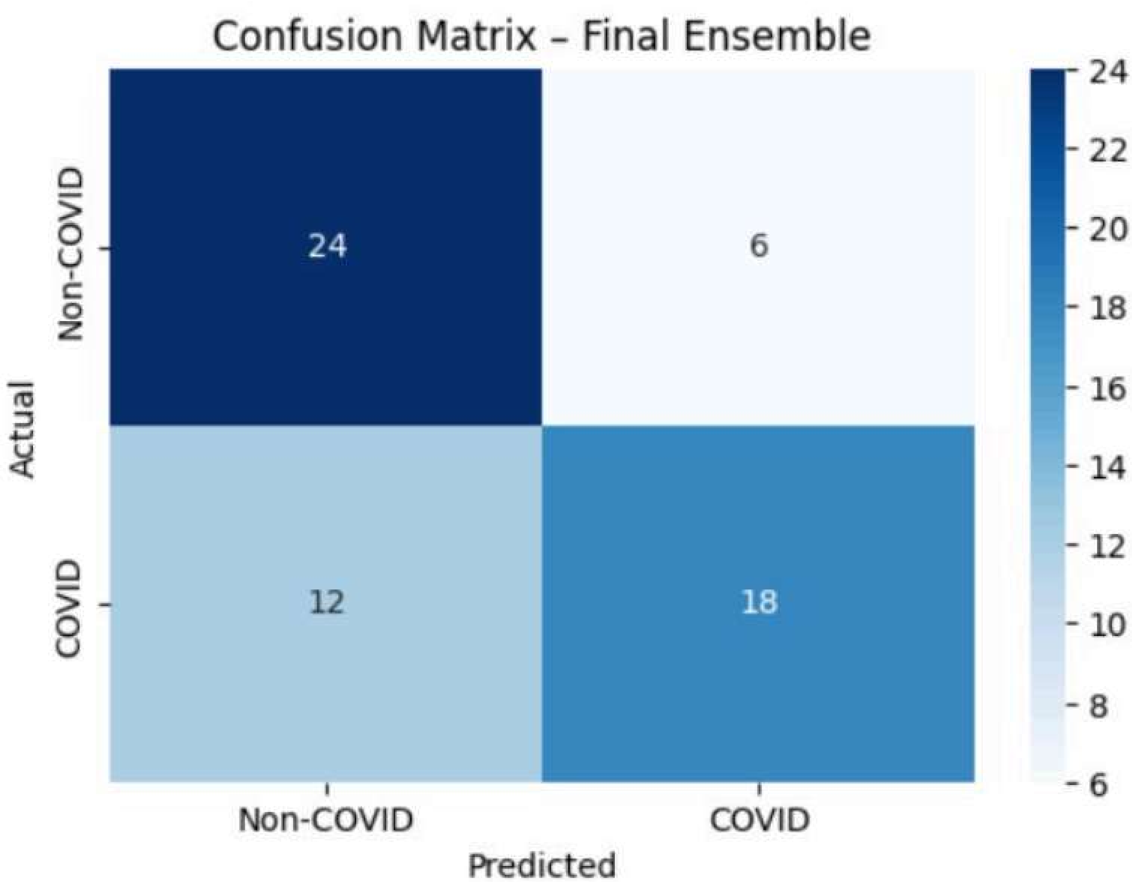


FIG.6 CONFUSION MATRIX

```
🔵 ResNet50 Confusion Matrix:
[[30  0]
 [30  0]]

🟢 DenseNet121 Confusion Matrix:
[[13 17]
 [ 6 24]]

🟣 Ensemble Model Confusion Matrix:
[[30  0]
 [30  0]]

🧾 Ensemble Classification Report:
              precision    recall  f1-score   support

           0       0.50      1.00      0.67        30
           1       0.00      0.00      0.00        30

    accuracy                           0.50        60
   macro avg       0.25      0.50      0.33        60
weighted avg       0.25      0.50      0.33        60
```

FIG.7 COMBINATION OF THREE MODELS

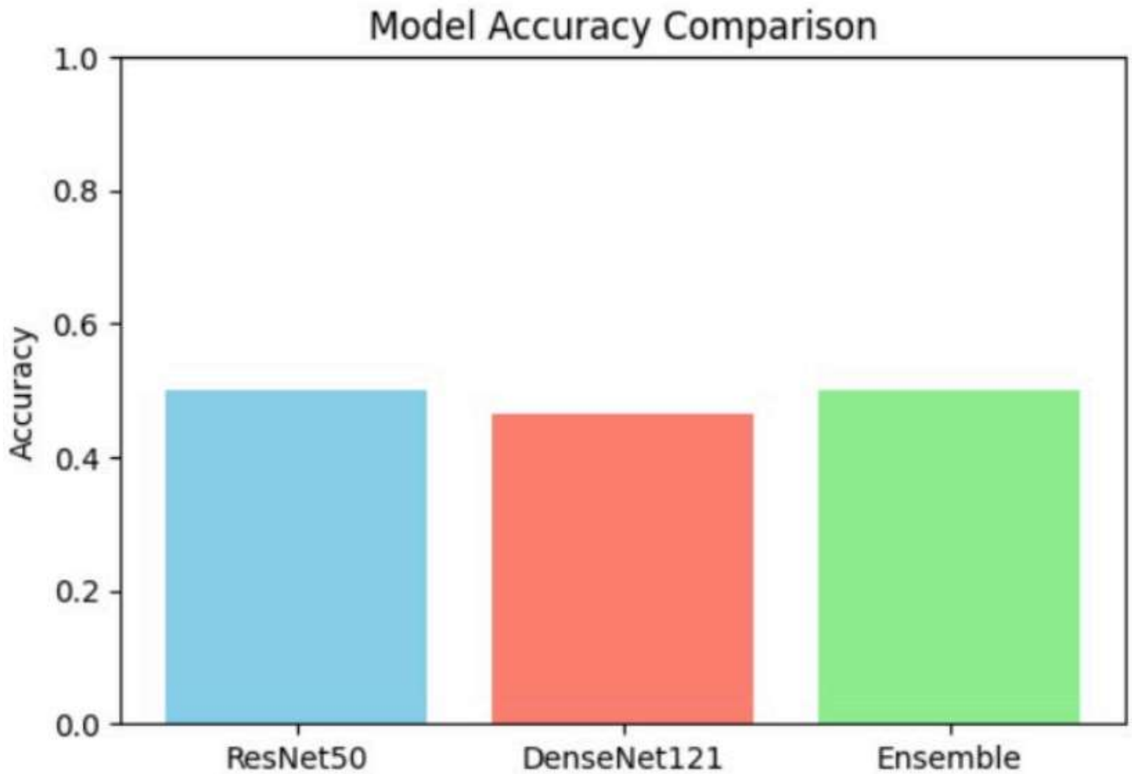


FIG.8 THREE MODELS COMPARISON

FUTURE ENHANCEMENT

1. CT Scan Integration: Include CT scan data for more detailed diagnosis and improved accuracy.

2. Advanced Ensemble Methods: Use weighted voting or stacking instead of basic soft voting to enhance ensemble performance.

3. Larger & Balanced Dataset: Expand dataset size and apply techniques like SMOTE or GANs to handle class imbalance.

4. Clinical Application: Develop a real-time diagnostic tool or web interface for hospital use.

5. Multi-disease Detection: Extend the model to classify other lung diseases like TB or lung cancer.

6. Explainability: Incorporate XAI tools like LIME or SHAP for more transparent and trustworthy predictions.

CONCLUSION

The proposed project aimed to develop a deep learning-based system for the automatic classification of COVID-19, Pneumonia, and Normal cases from chest X-ray images. By leveraging state-of-the-art convolutional neural network architectures, namely ResNet50 and DenseNet121, and combining them through ensemble modeling, this system sought to provide an accurate, reliable, and interpretable solution to support radiologists in medical diagnosis.

From the experimental results, it was observed that ResNet50 outperformed DenseNet121 with a higher classification accuracy of 87%, indicating its robustness and better generalization capability for this particular dataset. While DenseNet121 achieved moderate performance with an accuracy of 75%, the ensemble model, contrary to expectations, performed poorly with an accuracy of only 50%. This highlighted the importance of model selection and the need for more advanced ensemble strategies.

The system also incorporated Grad-CAM visualizations to enhance model interpretability. These heatmaps effectively showed which areas of the X-ray images influenced the model's decisions, ensuring transparency and building trust with end-users such as healthcare professionals.

Despite the promising results, the project encountered some challenges, including data imbalance, limited dataset size, and ensemble model inefficiency. These were mitigated to a certain extent using transfer learning, augmentation, and regularization techniques, but still point to potential areas for improvement in future iterations of this work.

In conclusion, this project demonstrates that deep learning models, particularly ResNet50, can significantly aid in the early detection and classification of COVID-19 through chest X-rays. Although not a replacement for RT-PCR or clinical evaluation, this system can serve as a supportive diagnostic tool, especially in resource-limited settings. With further enhancements such as the inclusion of CT scans, larger datasets, and better ensemble strategies, the system could be scaled and deployed in real-time clinical environments to assist in faster, reliable, and interpretable medical decisions.

REFERENCE

[1]Wang, L., & Wong, A. (2020). “COVID-Net: A tailored deep convolutional neural network design” for detection of COVID-19 cases from chest X-ray images.”https://arxiv.org/abs/2003.09871"

[2]Hussain, A., & Roy, P. (2020).”COVID-19 Detection using deep learning algorithms” for A systematic review.”https://doi.org/10.32604/cmc.2020.011195"

[3]Alzubaidi, L., et al. (2021).“Review of deep learning: Concepts, methods, and applications to medical imaging.” forJournal of Healthcare Engineering, 2021.”https://doi.org/10.1155/2021/3075871 “

[4]Ayyildiz, E., et al. (2021).“Covid-19 detection using deep learning models in chest X-ray images”for Computers, Materials & Continua, 68(2), 2231-2244.”https://doi.org/10.32604/cmc.2021.012797".

[5]Rajaraman, S., & Krishnan, R. (2020).“Deep Learning Approaches for COVID-19 Detection from Chest X-ray Images.”Proceedings of the 2020 IEEE International Conference on Artificial Intelligence and Data Engineering (AIDE 2020).”https://doi.org/10.1109/AIDE50881.2020.9291195".