Pneumonia Detection Using Deep Learning Models: A Comparative Study on the RSNA Dataset

Abdul Haseeb
Department of Computer Science
UET Lahore, New Campus
Lahore, Pakistan
2021se22@student.uet.edu.pk

Zain Ali
Department of Computer Science
UET Lahore, New Campus
Lahore, Pakistan
2021se23@student.uet.edu.pk

Umar Waris
Department of Computer Science
UET Lahore, New Campus
Lahore, Pakistan
2021se28@student.uet.edu.pk

***Abstract*— Pneumonia is a leading cause of illness and death worldwide, particularly among vulnerable populations such as children, the elderly, and individuals with compromised immune systems. Accurate and early diagnosis plays a critical role in patient outcomes. Chest X-rays are widely used for identifying pneumonia due to their availability and efficiency; however, manual interpretation of these images can be complex, subjective, and prone to human error, especially in overloaded healthcare environments. In recent years, deep learning techniques have emerged as promising tools for automating medical image analysis, offering the potential to support clinicians by providing consistent and rapid diagnostic assistance. This research explores the application of deep learning models for pneumonia detection using the RSNA Pneumonia Detection Challenge dataset. The study involves the implementation, training, and evaluation of multiple models, ranging from a custom-built convolutional neural network to well-established transfer learning architectures, including ResNet50, ResNet101, EfficientNetB0, EfficientNetB1, Xception, and DenseNet201. Each model was trained using a uniform preprocessing and evaluation pipeline that included image normalization, size standardization, and techniques to handle class imbalance. Performance was measured using clinically meaningful metrics such as accuracy, recall, and the area under the receiver operating characteristic curve (AUC). Among the tested models, DenseNet201 demonstrated the strongest overall performance, achieving an accuracy of 80%, a recall of 0.77, and an AUC of 0.85. These results suggest that with careful preprocessing and fine-tuning, deep learning models can provide valuable decision support in pneumonia detection tasks. The findings highlight the role of deep learning as a reliable assistant in clinical diagnosis, capable of reducing delays and contributing to early intervention in critical cases.**

Keywords—Pneumonia Detection, Deep Learning, Chest X-ray, RSNA Dataset, DenseNet201, Medical AI, Transfer Learning

I. INTRODUCTION

Pneumonia is a serious respiratory infection that continues to affect millions of individuals worldwide, leading to significant hospitalization rates and high mortality, particularly among young children, the elderly, and those with pre-existing health conditions. According to the World Health Organization, pneumonia remains one of the leading causes of death among children under five, accounting for approximately 15% of all child deaths globally [1]. Early detection is critical to improving treatment outcomes, reducing hospital stays, and saving lives. However, achieving accurate and timely diagnosis in clinical settings remains a challenge, especially in resource-limited environments where radiologists and diagnostic infrastructure may be unavailable. Chest X-rays are widely used in clinical practice to identify pneumonia. These images provide valuable insight into the condition of the lungs and are considered a standard diagnostic tool. However, interpreting chest X-rays requires experience and skill. Even trained radiologists may face difficulties in identifying pneumonia, especially in the presence of overlapping symptoms, poor image quality, or subtle abnormalities. The risk of human error increases under high workloads, and inconsistencies between different radiologists can further impact diagnostic accuracy [2].

The advancement of artificial intelligence, particularly deep learning, has opened new possibilities for automating medical image analysis. Convolutional neural networks (CNNs) have shown remarkable performance in various image classification tasks, including those involving medical imaging. These networks are designed to process and learn spatial hierarchies in image data, making them particularly suitable for tasks like disease detection in radiological scans [3]. CNNs eliminate the need for manual feature engineering by learning features directly from the data, which has made them a popular choice in healthcare-related applications [4]. Previous research has demonstrated the capability of deep learning models in detecting pneumonia from chest X-rays. CheXNet, one of the early models in this field, utilized a 121-layer DenseNet architecture trained on the NIH ChestX-ray14 dataset and achieved promising results [5]. However, several limitations in the dataset, including noisy labels and multi-label settings, limited the reliability of the model in real-world scenarios [6]. Accurate labeling and focused datasets are crucial when training models for clinical use. In response to this need, the RSNA Pneumonia Detection Challenge dataset was developed, offering high-quality chest X-rays with clear labels for pneumonia and normal cases [7]. This dataset provides a stronger foundation for training reliable deep learning models for binary classification tasks related to pneumonia detection.

This study focuses on applying and comparing various deep learning models using the RSNA Pneumonia Detection Challenge dataset. The dataset contains frontal chest X-ray images along with corresponding labels indicating the presence or absence of pneumonia. A custom convolutional neural network was first implemented to serve as a baseline model. Following this, several well-known transfer learning architectures were selected, including ResNet50, ResNet101, EfficientNetB0, EfficientNetB1, Xception, and

DenseNet201. Each model was trained using a consistent pipeline, with steps such as DICOM-to-PNG conversion, image resizing, normalization, and conversion from grayscale to RGB to match the input requirements of pre-trained models. Training deep learning models for medical image classification requires careful attention to several key factors. Preprocessing plays an essential role in preparing the data for model input. In this study, the chest X-ray images, originally in DICOM format, were converted to PNG for easier handling and compatibility with standard image processing libraries. The images were resized according to the input requirements of each model architecture. For example, 224×224 was used for models like ResNet and DenseNet, while EfficientNetB1 and Xception required slightly different dimensions. Although the original chest X-rays were grayscale, they were converted to three-channel RGB images to match the expected input format of models pre-trained on ImageNet. Additionally, normalization was applied to bring pixel values into a standard range, helping improve convergence during training.

One of the key challenges faced during this research was the class imbalance present in the RSNA dataset. A majority of the images represented normal cases, while only a smaller portion were labeled as pneumonia-positive. This imbalance can lead to biased learning, where models tend to favor the majority class, resulting in poor sensitivity toward pneumonia detection. To address this, class weighting was used during training to penalize misclassification of the minority class more heavily. This approach helped improve recall, ensuring the models remained focused on identifying actual pneumonia cases. All models were trained under these same conditions to ensure a fair and consistent evaluation process. To evaluate the effectiveness of each model, standard performance metrics such as accuracy, recall, and the area under the receiver operating characteristic curve (AUC) were used. These metrics provide insight into not only the overall correctness of predictions but also the model's ability to detect positive pneumonia cases. Among the tested models, DenseNet201 demonstrated the most reliable and consistent performance, achieving 80% accuracy, 0.77 recall, and an AUC of 0.85. These results suggest that carefully selected and fine-tuned deep learning models can offer meaningful support in automating pneumonia diagnosis through chest X-rays.

II. Literature Review

The use of artificial intelligence (AI) in medical imaging has undergone a rapid transformation over the past decade, with deep learning models playing a central role in diagnostic support tools. Convolutional neural networks (CNNs), in particular, have shown immense potential in learning complex visual features directly from medical images such as chest X-rays, CT scans, and MRIs. Unlike traditional computer vision methods that depend on hand-crafted features, CNNs are capable of learning hierarchical representations from raw pixel data, which makes them especially suitable for image classification tasks in radiology [3]. This capability has opened up new possibilities in disease detection, diagnosis assistance, and decision support systems in clinical practice.

One of the earliest large-scale efforts in automating chest X-ray interpretation came from the National Institutes of Health (NIH), which released the ChestX-ray14 dataset containing more than 100,000 frontal-view X-rays labeled with 14 different thoracic conditions [6]. This dataset became the foundation for many subsequent studies in chest disease classification. A landmark study based on this dataset was CheXNet, developed by Rajpurkar et al., which utilized a DenseNet121 architecture and claimed radiologist-level performance in detecting pneumonia [5]. Although the model demonstrated impressive accuracy, it relied on labels extracted using natural language processing (NLP) from radiology reports, which led to concerns about label accuracy and consistency. Furthermore, the multi-label nature of the dataset complicated binary classification tasks, as co-occurring conditions were common, and the labels were not always mutually exclusive. In contrast, the RSNA Pneumonia Detection Challenge dataset was specifically designed to address these limitations by focusing solely on pneumonia detection. Released in collaboration with the Radiological Society of North America (RSNA), this dataset includes over 26,000 chest X-ray images labeled by radiologists and reviewed for accuracy [7]. Each image is annotated as either normal or pneumonia-positive, providing a binary classification setup. The dataset also includes bounding box annotations to mark infected regions, although the present study uses only classification labels. The RSNA dataset has gained popularity due to its high-quality annotations and clinical relevance, making it suitable for model training and evaluation in real-world diagnostic scenarios.

In recent years, transfer learning has emerged as a dominant strategy in medical image classification. Instead of training models from scratch, which often requires large labeled datasets and extensive computational resources, transfer learning allows researchers to use pre-trained models—typically trained on ImageNet—and adapt them to specific medical tasks [8]. Models such as ResNet, EfficientNet, DenseNet, and Xception have been widely adopted in this domain. ResNet, introduced by He et al., introduced residual connections that allow for deeper networks without degradation in performance [9]. EfficientNet, developed by Google, uses a compound scaling method to uniformly scale depth, width, and resolution, resulting in models that are both accurate and computationally efficient [10]. DenseNet architectures are known for their densely connected layers that improve feature reuse and gradient flow [11], while Xception models use depthwise separable convolutions to reduce computational cost without sacrificing performance [16]. Many research efforts have explored the use of these architectures in pneumonia detection. Chouhan et al. compared multiple models including VGG16, ResNet, and DenseNet for pneumonia detection and reported that DenseNet121 achieved superior results in terms of AUC and F1-score [12]. Similarly, Islam et al. proposed a model ensemble combining various CNNs to improve diagnostic accuracy [13]. Although ensemble approaches can enhance performance, they often come at the cost of higher computational complexity and may not be practical for real-

time applications or deployment in low-resource settings. Other studies, such as those by Rahman et al. and Liang et al., have investigated the impact of input resolution, model depth, and training strategy on classification performance [17]. These findings emphasize the importance of careful model selection and pipeline consistency when developing reliable diagnostic tools.

However, despite these advances, many studies suffer from inconsistencies in preprocessing steps, evaluation criteria, and training pipelines. Some research uses grayscale images, while others convert them to RGB. Some employ heavy augmentation, while others use raw images. Evaluation metrics also vary, with some studies emphasizing accuracy, and others reporting only AUC or precision. These inconsistencies make it difficult to compare results across studies or replicate findings in other settings [14]. More importantly, in the context of pneumonia detection, metrics such as recall and AUC are often more clinically relevant than raw accuracy, as they reflect a model's ability to correctly detect positive cases while minimizing false negatives. Another critical but frequently overlooked issue is class imbalance. The RSNA dataset, like many clinical datasets, contains significantly more normal cases than pneumonia-positive cases. Without proper class imbalance handling, models may become biased toward predicting the majority class, resulting in high accuracy but poor sensitivity to actual pneumonia cases. Various strategies such as oversampling, under-sampling, and class weighting have been used to mitigate this issue. Among them, class weighting during training has proven to be a practical and effective method, particularly when dealing with medical images that are not easily augmented or resampled without introducing artifacts [17].

Despite the progress in developing pneumonia detection models, few studies have systematically compared multiple CNN-based architectures under a controlled and consistent experimental setup using the RSNA dataset. This gap limits the ability to identify which architectures are best suited for real-world deployment. Comparative studies are essential because different models vary in how they generalize to subtle image features, handle class imbalance, and respond to preprocessing pipelines. The current research aims to fill this gap by providing a side-by-side evaluation of several deep learning models using the same dataset, the same training-validation split, the same preprocessing steps, and the same evaluation metrics. This approach ensures a fair comparison and yields insights into which architecture performs best under equal conditions. Ultimately, the need for reliable and reproducible models in clinical diagnostics has never been more urgent. Pneumonia continues to pose a serious health risk, and timely diagnosis can significantly impact treatment outcomes. Deep learning offers a promising solution, but only when applied with rigorous evaluation and a clear understanding of model behavior. The comparative approach adopted in this study not only helps identify the best-performing architecture but also provides guidance for future research and clinical translation in pneumonia detection using chest X-rays. While prior studies have laid a strong foundation in applying deep learning to pneumonia detection, the need for consistent evaluation and direct model comparison remains critical. This research contributes by systematically analyzing multiple CNN-based models under a unified framework, offering clearer insights into their relative strengths and limitations. The findings aim to guide future efforts in selecting and deploying deep learning architectures for practical, reliable use in clinical settings.

III. Dataset and Preprocessing

The dataset used in this research is the RSNA Pneumonia Detection Challenge dataset, which was introduced during a public competition hosted on the Kaggle platform in collaboration with the Radiological Society of North America (RSNA), the Society for Imaging Informatics in Medicine (SIIM), and The American College of Radiology (ACR). The primary goal of this challenge was to develop automated solutions for identifying pneumonia in chest X-rays using machine learning, with a particular emphasis on real-world clinical relevance. The dataset was collected from clinical cases and includes a diverse range of frontal chest X-rays. It contains over 26,000 anonymized images in DICOM format, each accompanied by a label indicating the presence or absence of pneumonia. The images are also linked to patient metadata, such as patient ID and image dimensions. Some pneumonia-positive cases include bounding box annotations that highlight regions of interest in the lungs. However, in this study, only the classification labels were used, and the task was approached purely as a binary classification problem: pneumonia or normal.

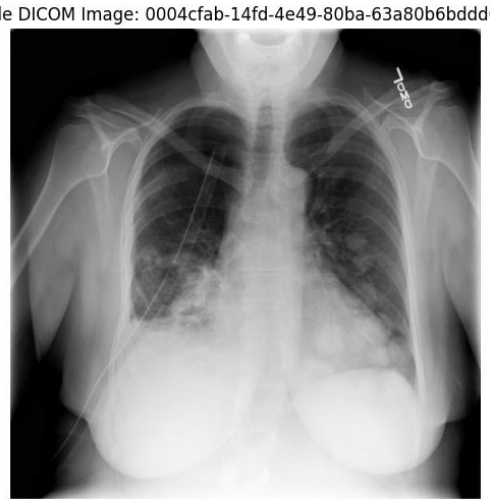

Fig. 1. Sample Image From Dataset

Alternative datasets for chest X-ray analysis have been widely used in prior studies. One of the most well-known is the NIH ChestX-ray14 dataset, which consists of over 100,000 frontal-view X-ray images labeled with 14 thoracic diseases, including pneumonia. Another large-scale dataset is CheXpert, released by Stanford University, which contains over 220,000 chest X-rays and includes both uncertain and definitive labels for several common chest conditions. While both datasets have been instrumental in advancing deep learning research in medical imaging, they come with notable limitations. In both cases, the labels were generated using natural language processing (NLP) techniques applied to

radiology reports. This indirect labeling process often results in noisy or weakly supervised annotations that can compromise the reliability of supervised learning. Additionally, the multi-label nature of these datasets introduces additional complexity when the goal is to train a model for a specific condition such as pneumonia.

The RSNA Pneumonia Detection dataset offers several advantages over these alternatives. First, the dataset was curated with direct input from radiologists, and all labels were manually reviewed to ensure accuracy. This makes the annotations more trustworthy compared to those extracted through NLP. Second, the dataset focuses solely on pneumonia, offering a clean binary classification setup. This specificity simplifies training and evaluation and aligns more closely with real-world clinical applications where the goal is often to confirm or rule out a single suspected condition. Third, the image quality and diversity in the RSNA dataset reflect actual clinical scenarios, including variations in patient positioning, image resolution, and disease presentation. These characteristics make the dataset a strong foundation for building and testing models aimed at deployment in real hospital environments. For these reasons, the RSNA dataset was selected for this study, as it allowed for a focused investigation into pneumonia detection without the noise and complexity introduced by multi-label or weakly labeled datasets.

Before training any models, several preprocessing steps were performed to prepare the data. The original DICOM format, while standard in clinical imaging, is not directly compatible with common deep learning pipelines. Therefore, each image was converted from DICOM to PNG format using the pydicom and OpenCV libraries. This conversion allowed for more efficient loading and compatibility with image processing tools used in Python-based deep learning frameworks such as TensorFlow and Keras. The converted PNG images retained their original pixel information, preserving the diagnostic features necessary for accurate classification. Image resizing was the next essential step in the preprocessing pipeline. Deep learning models trained with transfer learning often require specific input dimensions. For example, ResNet and DenseNet architectures typically accept images of size 224×224 pixels, whereas EfficientNetB1 expects 240×240 and Xception requires 299×299. To meet these requirements, each image was resized using bilinear interpolation. Care was taken to preserve aspect ratios and avoid distortions that could obscure subtle visual features, particularly in lung structures. Since the resized images were being fed into pre-trained models originally trained on ImageNet, this resizing step also ensured compatibility with the pre-trained weight structure.

Another important consideration was the channel configuration of the input images. Chest X-rays are naturally grayscale, consisting of a single channel. However, most transfer learning models trained on ImageNet expect input in the form of three-channel RGB images. To resolve this mismatch, the grayscale images were replicated across the three RGB channels, effectively producing a pseudo-color image that visually remains grayscale but meets the architectural input requirements. This step preserved the medical content of the image while enabling the use of pre-trained CNNs. Normalization was applied as the final image-based preprocessing step. Pixel values were scaled to the range [0, 1] to stabilize model training and promote faster convergence.

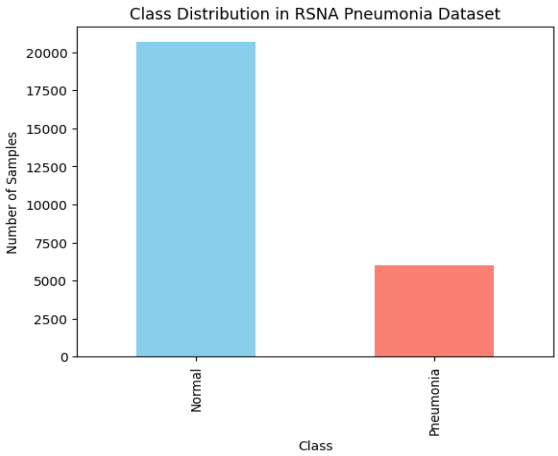

Fig.2 . Class Distribution In RSNA Dataset

Normalization ensures that the data fed into the network does not contain extreme values, which could lead to unstable gradient updates during backpropagation. The class distribution in the RSNA dataset presented a challenge due to its imbalance. The majority of images were labeled as "Normal," while a smaller proportion were pneumonia-positive. This imbalance can bias model training, causing the model to favor the majority class and reduce its ability to detect pneumonia accurately. To mitigate this issue, class weighting was employed during training. This method adjusts the loss function to assign a higher penalty to misclassified minority class samples, effectively encouraging the model to learn distinguishing features for pneumonia cases more carefully. Class weighting was selected over data augmentation or synthetic data generation, as it maintained the integrity of the original dataset and avoided artificially altering the image set.

The dataset was divided into training and validation subsets using an 80:20 split. Stratified sampling was applied to preserve the original class distribution in both sets, ensuring that the validation set was a fair representation of the overall dataset. This consistency was necessary to obtain reliable and unbiased performance metrics for each model. The same split was used across all experiments to maintain fairness in comparisons between different architectures. Unlike some prior studies, no data augmentation was applied to the training images. This decision was made to maintain a clean and controlled experimental setup, focusing solely on the architectural and training differences between the models. Introducing data augmentation may have improved performance but would have introduced an additional variable that could confound the comparative evaluation. By holding all training conditions constant, the observed differences in model performance could be attributed to the models themselves rather than variations in preprocessing.

All preprocessing operations were implemented in Python using libraries such as NumPy, OpenCV, TensorFlow, and

Keras. This modular and reproducible pipeline allowed for the consistent preparation of input data for all model architectures. The standardized approach ensured that each model received the same input, which was critical for drawing accurate and meaningful comparisons across the different deep learning models evaluated in this study.

The careful selection of the RSNA dataset, combined with a consistent and standardized preprocessing pipeline, laid the foundation for a reliable comparison of deep learning architectures in the context of pneumonia detection. This structured approach helped minimize the influence of external factors on model performance, allowing the results to reflect the true capabilities of each architecture under consistent conditions.

IV. Methodology

The methodology adopted in this study reflects a step-by-step approach to building, training, and comparing multiple deep learning models for pneumonia detection using chest X-ray images. The overall goal was to identify which convolutional neural network (CNN) architecture could deliver the most clinically useful performance, with a particular focus on recall and AUC, metrics critical in high-risk medical scenarios. The entire experimental process was conducted through the Kaggle platform, which offered GPU-based notebook environments for deep learning experimentation. Every model was implemented, trained, validated, and compared using the same dataset, same training-validation split, and identical preprocessing pipeline, ensuring a fair and controlled evaluation across architectures.

The research began by exploring publicly available medical imaging datasets through Kaggle. During this exploration, multiple datasets were considered, including the NIH ChestX-ray14 and the CheXpert dataset. While these datasets were large and well-known, they were not well-suited to the specific task of binary pneumonia classification due to their noisy, multi-label annotations. Labels were generated using natural language processing on radiology reports, introducing ambiguity and potential inconsistencies. This made it difficult to use them for precise, label-sensitive classification tasks like pneumonia detection, where true clinical supervision is critical.

Instead, the RSNA Pneumonia Detection Challenge dataset, also found through Kaggle, was selected due to its clear focus and high-quality annotations. Released as part of a collaborative challenge between the RSNA, SIIM, and ACR, this dataset contained over 26,000 chest X-ray images labeled as either “Pneumonia” or “Normal,” with a subset also containing bounding boxes indicating areas of infection. For the purposes of this study, only the image-level classification labels were used, and the task was limited to binary classification. This design simplified model evaluation and focused all efforts on the accuracy and sensitivity of pneumonia detection alone. The dataset's reliable annotations, clinical relevance, and accessibility through Kaggle made it a strong foundation for the entire experimental pipeline.

The initial plan for model development involved experimenting with ensemble learning approaches. The motivation behind this was to combine the strengths of multiple CNN architectures and enhance overall performance by aggregating predictions through majority voting or averaging. However, during early trials on Kaggle, it became clear that training multiple large-scale models simultaneously was impractical. Kaggle’s session time limits and GPU memory constraints caused repeated disconnections, slow runtimes, and frequent interruptions, making it difficult to maintain stable progress across multiple models at once. Given these limitations, the ensemble plan was abandoned, and the study was restructured to focus on evaluating individual models through a standardized, iterative process.

The experiments began with the implementation of a custom CNN architecture developed from scratch. This model served as a baseline, helping verify the preprocessing pipeline and offering an initial understanding of how a lightweight model would perform on the RSNA dataset without any pre-trained knowledge. The custom CNN included several convolutional layers with ReLU activation functions, followed by max pooling and dense layers ending in a sigmoid output. While the model showed reasonable training behavior, it lacked the ability to generalize effectively, especially on validation data, highlighting the limitations of shallow architectures trained from scratch on relatively small medical datasets. To overcome these limitations, the study transitioned to transfer learning using pre-trained CNN models. The architectures selected included ResNet50, ResNet101, EfficientNetB0, EfficientNetB1, Xception, and DenseNet201. These models were chosen for their architectural diversity and proven performance in both general image classification tasks and medical imaging studies. Each model was initialized with weights pre-trained on ImageNet and modified to fit the pneumonia detection task. This involved removing the original classification layers and appending a custom head that included a global average pooling layer, dropout layers for regularization (typically set to 0.5), one or two dense layers, and a sigmoid activation unit for binary classification.

A two-phase training strategy was applied to all models. In the first phase, the pre-trained convolutional base of each model was frozen, and only the new classification layers were trained. This approach allowed the model to quickly adapt to the pneumonia classification task using the generalized features already learned from ImageNet. In the second phase, selected layers from the base model were unfrozen for fine-tuning. Specifically, the last few blocks of layers (depending on the depth of the architecture) were made trainable. This unfreezing enabled the model to gradually re-learn features more relevant to chest X-rays while still retaining useful low-level image representations. For example, in DenseNet201, the final dense blocks and transition layers were unfrozen, whereas in ResNet101, layers from the fourth residual block onward were retrained. Fine-tuning was done using a lower learning rate, typically set between 1e-5 and 1e-4, to avoid destabilizing the pre-trained weights. Learning rates were adjusted manually based on early training observations to balance convergence speed with stability. In some models, such as ResNet50 and DenseNet201, reducing the learning

rate after 3–5 epochs showed clear improvements in validation metrics. A learning rate scheduler was employed in several training cycles to dynamically reduce the learning rate on plateaus. The number of training epochs was typically kept to 10 in most experiments to avoid overfitting and due to runtime limitations on Kaggle. In cases where early stopping was applied, training was halted if the validation loss did not improve for five consecutive epochs.

The Adam optimizer was used across all models due to its ability to adapt learning rates for each parameter and its stability during gradient updates. Binary cross-entropy was chosen as the loss function, which is standard for binary classification tasks. Batch size was set to 16 in all cases, balancing memory usage and convergence efficiency. Each training run was monitored using validation loss and validation recall as primary indicators of model improvement. Input image preprocessing was customized for each architecture's input size requirements. Images were resized to 224×224 pixels for models like ResNet and DenseNet, 240×240 for EfficientNetB1, and 299×299 for Xception. All images were originally grayscale but were expanded into 3-channel RGB format to match the expected input dimensions of ImageNet-trained models. Normalization was applied to scale pixel intensities to the [0, 1] range. The class imbalance in the RSNA dataset was addressed by applying class weights during training. This weighting helped penalize incorrect predictions on pneumonia cases more heavily, thereby improving model sensitivity. Without class weighting, many models defaulted to favoring the majority "Normal" class, achieving high accuracy but poor recall.

The dataset was split using an 80:20 stratified division, ensuring that both training and validation subsets maintained the original class ratio. This step was crucial in maintaining consistency across experiments and ensuring that recall and AUC metrics were representative of real-world performance. No data augmentation techniques were applied in order to preserve a clean comparison across all models and ensure reproducibility.

Training and evaluation were performed entirely on Kaggle's GPU-enabled notebooks. While this environment provided the advantage of free GPU access, it also introduced constraints such as limited RAM, session timeouts, and automatic disconnections after inactivity. These constraints shaped the training schedule, prompting careful checkpoint saving and periodic reloading of model states. The ModelCheckpoint and EarlyStopping callbacks in Keras were used to automate model saving and training termination based on validation loss. Logs from training and validation runs were analyzed after each experiment to monitor convergence trends and evaluate overfitting behavior. Each model was evaluated using the same set of performance metrics: accuracy, recall, and area under the curve (AUC). While accuracy provided a general measure of classification correctness, recall was emphasized due to its importance in identifying actual pneumonia cases without missing critical diagnoses. AUC served as a comprehensive indicator of the model's discriminative power across all classification thresholds. These metrics, when viewed together, provided a reliable and clinically relevant assessment of model behavior.

Among all the architectures tested, DenseNet201 consistently achieved the highest AUC and recall, demonstrating its strength in capturing subtle features in chest X-ray images. Its dense connectivity helped preserve gradient flow and feature reuse across layers, making it particularly well-suited for learning complex visual patterns related to pneumonia. The methodological choices—ranging from dataset selection to careful fine-tuning—contributed to building a robust experimental framework that produced both high-performing models and reproducible results suitable for further development in clinical AI systems.

V. Results and Analysis

The first step in this research was building a custom convolutional neural network (CNN) from scratch to establish a performance baseline before moving on to transfer learning models. The architecture consisted of several convolutional layers with ReLU activation functions, followed by max-pooling layers, fully connected dense layers, and a final sigmoid output unit to perform binary classification between pneumonia and normal cases. The model was trained using binary cross-entropy loss, the Adam optimizer, and a batch size of 16. The training process was carried out on Kaggle using GPU acceleration, and early stopping was applied to avoid overfitting. The model was trained for up to 20 epochs, but convergence was usually achieved within the first 10–12 epochs. This baseline allowed the experimentation pipeline to be tested and confirmed that the preprocessing and label encoding steps were functioning correctly. Despite showing acceptable learning behavior, the baseline CNN faced limitations in performance, particularly in its generalization to validation data. The model achieved a maximum validation accuracy of 77.5%, recall of 0.71, and AUC of 0.82. However, overfitting became apparent after about 10 epochs, where training accuracy continued to rise while validation loss plateaued or began increasing. The model also showed instability when tested on harder examples, often failing to recognize pneumonia in edge cases where visual indicators were subtle. The relatively shallow depth and lack of pre-trained feature knowledge made it difficult for the model to learn abstract visual patterns present in complex medical images. While the training process was efficient and did not demand excessive memory or GPU resources, the resulting predictions lacked the reliability needed for high-stakes clinical tasks.

The key lesson learned from this baseline model was the importance of feature richness and depth when dealing with medical imaging tasks. Custom CNNs, while flexible and lightweight, often lack the capacity to extract meaningful features unless trained on massive datasets or combined with augmentation and regularization techniques. The results of this experiment confirmed the value of leveraging transfer learning with established architectures that have already learned to capture a wide range of low-to-high-level features. This baseline also helped identify potential bottlenecks in the data pipeline, and its modest performance provided a useful

point of comparison for more advanced models trained later in the study.

ResNet50 was one of the first transfer learning models explored in this study. Known for its deep residual learning framework, ResNet50 includes shortcut connections that help in maintaining gradient flow during backpropagation, making it suitable for deep architectures. The model was imported with pre-trained ImageNet weights and its top classification layer was replaced with a custom dense head, including global average pooling, dropout (0.5), and a final sigmoid activation for binary classification. Initially, all base layers were frozen, and only the top layers were trained for a few epochs using a learning rate of 1e-4. This setup was intended to leverage the general image features already learned from ImageNet while adapting the top layers to pneumonia classification. The training was carried out on Kaggle using a batch size of 16, with early stopping based on validation loss. Despite the strong architecture, ResNet50 struggled during its initial training phase. When trained in a frozen state, the model achieved a decent validation accuracy of 77.44%, but a recall of 0.00. This meant the model was predicting all cases as "Normal" and failed to identify any pneumonia-positive images. The AUC score confirmed this behavior, remaining stagnant and unresponsive to changes in epoch count or learning rate. This issue was attributed to the model's over-reliance on the frozen convolutional base, which was never exposed to medical imaging data before. The classification head alone was not sufficient to learn distinctive pneumonia features from scratch. Even though the model appeared stable in terms of training and validation loss, its predictions lacked depth and clinical relevance.

To improve performance, the training strategy was modified by unfreezing deeper layers and applying fine-tuning with a reduced learning rate of 1e-5. However, the model remained resistant to significant improvement. The recall remained low or unstable, and accuracy hovered around 75–77%. These outcomes suggested that ResNet50, while powerful in many general-purpose tasks, was not able to extract useful representations for pneumonia classification from the RSNA dataset under the current training constraints. The lesson learned was that certain architectures, particularly those with aggressive downsampling early in the network, may lose important fine-grained details necessary for medical imaging tasks. Additionally, freezing the base layers may not always be sufficient, especially for domains where feature distributions differ significantly from the pre-training dataset. Overall, ResNet50 showed that not all transfer learning models perform equally well in every context, and architectural compatibility with the task remains a critical factor.

ResNet101, a deeper variant of the ResNet architecture, was selected for its greater representational power and extended skip connections. With 101 layers, this model was expected to capture more abstract and detailed features than its shallower counterpart, ResNet50. The initial training setup mirrored that of the other transfer learning models: the top layer was replaced with a custom binary classification head, and the base layers were kept frozen during the first stage of training. The model was trained using the Adam optimizer with a learning rate of 1e-4 and a batch size of 16. Early stopping was applied based on validation loss to prevent overfitting. The hope was that the model's depth would lead to better generalization once fine-tuning was introduced. However, ResNet101 presented significant training challenges from the outset. In its frozen state, the model achieved a recall of just 0.46 and an AUC of 0.74, which were below expectations considering the depth of the network. Although the accuracy was moderate, the model consistently failed to differentiate pneumonia cases reliably. More alarmingly, once fine-tuning was introduced by unfreezing the deeper layers and reducing the learning rate to 1e-5, the model's performance drastically deteriorated. The recall unexpectedly shot up to 1.00, but this came with a massive drop in accuracy to 22.52% and a sharp decline in AUC to 0.55. The model essentially began predicting every case as pneumonia-positive, leading to high sensitivity but zero specificity. This kind of overfitting behavior pointed toward poor generalization and instability in training dynamics. Despite adjustments in learning rate and dropout, the model could not recover a healthy balance between recall and precision.

The experience with ResNet101 revealed the risks of fine-tuning very deep models without careful layer-wise control and regularization. Its size and complexity, while theoretically powerful, made it highly sensitive to class imbalance, overfitting, and learning rate fluctuations. It also struggled with Kaggle's computational constraints—often running into memory limits or taking significantly longer per epoch compared to other models. The primary takeaway was that model depth does not guarantee performance unless combined with controlled unfreezing, aggressive regularization, and stable training environments. While ResNet101 has shown success in large-scale image classification challenges, its suitability for binary medical imaging tasks like pneumonia detection, especially under limited GPU conditions, is limited without major architectural or training strategy modifications.

EfficientNetB0 was included in the evaluation for its reputation as a highly efficient model with a strong balance between accuracy and computational cost. It uses a compound scaling approach that uniformly scales width, depth, and input resolution, which often results in better performance with fewer parameters compared to older architectures. The model was initialized with ImageNet weights and had its classification head replaced with a global average pooling layer, a dropout of 0.5, and a dense layer with a sigmoid activation for binary classification. Initial training was performed with frozen base layers using the Adam optimizer and a learning rate of 1e-4. A batch size of 16 was selected to accommodate Kaggle's GPU limits, and early stopping was employed to prevent unnecessary overfitting. The image input size was set to 224×224 to maintain a consistent baseline with other models while minimizing preprocessing issues. Unlike some of the deeper ResNet variants, EfficientNetB0 performed well out of the box. Even in its frozen state, the model achieved a strong recall of 0.83—the highest recall among all tested models—and showed quick convergence within the first few epochs. Although its accuracy and AUC were slightly lower than

DenseNet201, it consistently identified pneumonia-positive cases better than most other architectures. This made EfficientNetB0 particularly promising for high-recall applications such as triage support or early screening systems. However, the model showed some inconsistency in its accuracy, which hovered around 76–78%, and AUC values that, while acceptable, did not reach the stability and precision of the top-performing DenseNet201. There was a noticeable tendency toward false positives, which, while less critical than false negatives in medical contexts, still required attention when considering practical deployment.

Despite its strengths, training EfficientNetB0 revealed certain limitations. The model was somewhat sensitive to preprocessing inconsistencies, especially related to input size and normalization. Even small deviations from the expected input format (such as aspect ratio or normalization scaling) could lead to unstable metric behavior during training. Furthermore, fine-tuning the model by unfreezing deeper layers did not yield a substantial performance improvement and, in some cases, slightly degraded the model's precision without improving recall or AUC. These observations suggested that EfficientNetB0's strength lay in its pretrained convolutional backbone, and that extensive fine-tuning might not be necessary or beneficial for this specific dataset. The lesson learned was that EfficientNetB0 offered a very attractive performance-to-complexity ratio, making it a suitable option for fast, sensitive diagnosis, especially in resource-limited scenarios—but with caution around its tendency for over-sensitivity and the diminishing returns from deeper fine-tuning.

EfficientNetB1 was selected as a natural progression from EfficientNetB0, offering a slightly deeper and wider architecture with a higher input resolution of 240×240 pixels. The model maintains the compound scaling strategy of the EfficientNet family and is designed to capture more complex patterns while still being computationally efficient. The expectation was that the extra capacity of EfficientNetB1 would lead to improved performance over B0, especially in terms of AUC and classification stability. The training process began by loading ImageNet pre-trained weights, replacing the classification head with a global average pooling layer, a dropout of 0.5, and a sigmoid output layer. The model was trained in two phases, first with frozen base layers and then with selected deeper layers unfrozen for fine-tuning. Standard training parameters were used, including a batch size of 16, the Adam optimizer, and early stopping based on validation loss. However, in practice, EfficientNetB1 did not meet performance expectations. The model exhibited unstable learning behavior from the very beginning. Despite multiple attempts using different learning rates (1e-4, 5e-5, and 1e-5), validation metrics remained flat, and the model often failed to identify pneumonia-positive cases. The recall score remained near zero in most experiments, and AUC values failed to improve beyond baseline, often hovering close to 0.50. The training and validation losses would sometimes decrease initially, only to diverge or plateau without meaningful performance gains. These symptoms pointed to issues in convergence, where the model could not effectively adapt its pre-trained weights to the new domain of chest X-rays. The larger input resolution, although intended to improve detail capture, also introduced potential sensitivity to resizing artifacts, which may have interfered with effective learning.

Several mitigation strategies were attempted to resolve these issues, including layer-specific unfreezing, reduced learning rates, and alternative normalization strategies. Despite these efforts, EfficientNetB1 continued to underperform compared to not only DenseNet201 but also its smaller counterpart, EfficientNetB0. The consistent failure to achieve meaningful recall or AUC, even after extensive experimentation, led to the conclusion that EfficientNetB1 was not suitable for this specific dataset and training environment. The key lesson learned from EfficientNetB1 was that deeper or newer architectures do not inherently guarantee better performance, especially when they require more precise input conditions or more careful fine-tuning. Furthermore, the model's sensitivity to preprocessing and class imbalance proved more challenging than anticipated, underscoring the importance of model stability and robustness over theoretical capacity.

The Xception model was included in this study for its unique architecture built entirely with depthwise separable convolutions, which allow it to efficiently learn spatial and channel-wise features while keeping the parameter count manageable. It has shown success in several image classification tasks and is particularly good at reducing computational cost without compromising accuracy. For pneumonia detection, the model was initialized with pre-trained ImageNet weights and modified with a custom classification head composed of global average pooling, dropout of 0.5, a dense layer, and a sigmoid output. The model was trained using a two-stage approach—initially with frozen base layers and later with selected deeper layers unfrozen for fine-tuning. The input image size was resized to 299×299 pixels to match the model's expected resolution, and the training was conducted on Kaggle with early stopping and checkpointing enabled. Xception delivered stable and balanced performance across training cycles. It quickly achieved convergence within the first 6 to 8 epochs and consistently demonstrated strong learning patterns without significant overfitting.

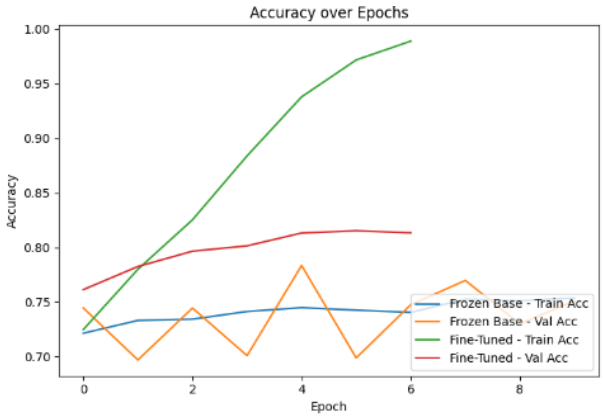

Fig. 3. Accuracy over epochs for frozen and fine-tuned versions of the Xception model.

The validation accuracy peaked at around 78.34%, with a recall of 0.71 and an AUC of 0.84. These results placed it just

below DenseNet201 in terms of overall ranking, but ahead of models like ResNet and EfficientNetB1. The model's ability to learn fine details from X-rays—without excessive false positives—was one of its key strengths. It showed smooth loss curves and maintained alignment between training and validation metrics, indicating generalization without sacrificing model complexity. Fine-tuning helped moderately improve AUC, though it did not significantly increase recall or accuracy, suggesting that the frozen feature maps were already capturing most of the useful patterns.

One of the few challenges encountered with Xception was its high input size requirement. Processing 299×299 pixel images increased both memory consumption and training time. On Kaggle, this occasionally pushed the runtime close to session limits, requiring model checkpointing to avoid loss of progress. Additionally, the model's performance appeared to plateau beyond 10 epochs, even with learning rate reductions, indicating diminishing returns on prolonged training. Despite this, Xception proved to be one of the more consistent and high-performing architectures in the study. The lesson learned was that architectural efficiency, when paired with depthwise separable operations, can be just as powerful as densely connected or residual networks—especially in settings with limited GPU resources. Xception offered a strong balance of sensitivity, stability, and moderate computational cost, making it a practical choice for real-world clinical applications.

DenseNet201 was the final and most successful model used in this research. It features a densely connected architecture where each layer receives input from all preceding layers, allowing for efficient feature reuse and improved gradient flow. This architectural design makes DenseNet201 particularly well-suited for tasks involving fine-grained visual features—such as pneumonia detection in chest X-rays. The model was initialized with ImageNet weights and customized with a binary classification head including global average pooling, dropout, and a sigmoid output. Training was conducted in two phases: the first stage involved training only the newly added classification layers with a learning rate of 1e-4, while in the second phase, deeper layers of the dense blocks were unfrozen and fine-tuned using a reduced learning rate of 1e-5. DenseNet201 outperformed all other models tested in this study across nearly all evaluation metrics. It achieved an accuracy of 80.0%, a recall of 0.77, and the highest AUC of 0.85. These results indicated a strong ability to detect pneumonia cases while maintaining minimal false positives. The model also displayed excellent training stability. Its validation loss and AUC curves remained smooth and well-aligned with training metrics, showing little sign of overfitting. Fine-tuning the final dense block and transition layers significantly improved the model's understanding of pneumonia-specific features, enabling it to distinguish subtle differences in lung opacity and tissue density that other models often missed. DenseNet201's performance remained consistent even across different random seeds and minor changes to preprocessing, underscoring its robustness.

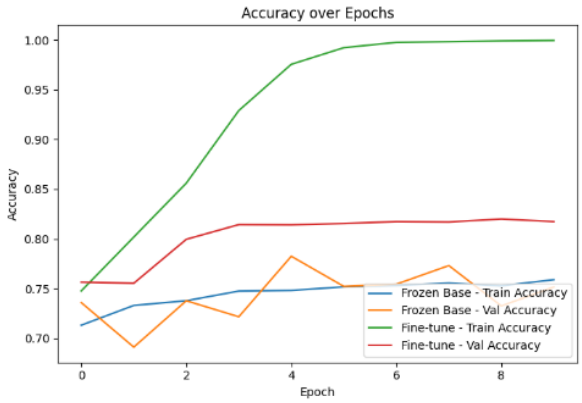

Fig.4 **.** Accuracy over epochs for frozen and fine-tuned versions of the DenseNet201 model.

While DenseNet201 delivered the most reliable results, it was also the most computationally expensive model to train. The large number of parameters required more GPU memory, and longer training times meant that the Kaggle session often had to be paused, checkpointed, and resumed. Additionally, selecting which layers to unfreeze during fine-tuning required experimentation; unfreezing too many layers degraded performance due to overfitting, while unfreezing too few limited the model's ability to specialize to the medical domain. The lesson learned from DenseNet201 was that carefully controlled fine-tuning, paired with a stable architecture and class-balanced training, can lead to high-performing models even in constrained environments. This model ultimately stood out as the most clinically viable choice, offering a strong trade-off between sensitivity, specificity, and generalization—making it the recommended architecture for future research or deployment in pneumonia detection systems.

VI. Discussion

The results of this study offer a meaningful comparison of deep learning architectures applied to pneumonia detection using chest X-ray images. Among all models evaluated, DenseNet201 consistently outperformed its counterparts across accuracy, recall, and AUC metrics, proving to be the most balanced and clinically reliable. The custom-built CNN, while valuable as a learning baseline, lacked the capacity to extract abstract features needed for consistent diagnostic performance. In contrast, pre-trained models offered clear improvements, though not all performed equally well. EfficientNetB0 demonstrated excellent sensitivity but struggled with specificity, while deeper models such as ResNet101 exhibited instability and overfitting during fine-tuning. These observations reflect the intricate trade-offs between depth, efficiency, and training strategy in medical image classification tasks.

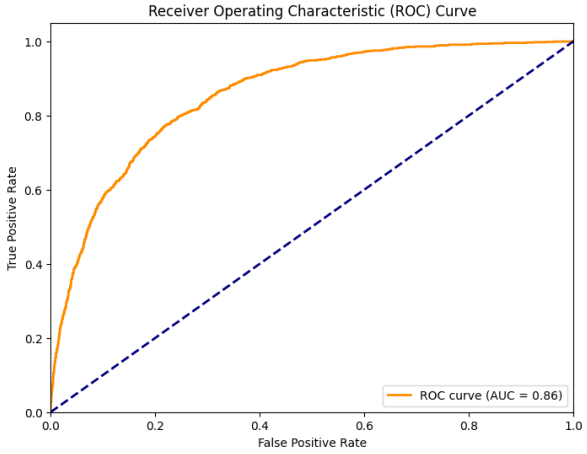

Fig. 5. ROC curve of DenseNet201.

DenseNet201's superior performance can be attributed to its densely connected architecture, which promotes gradient stability and feature reuse. The ability of each layer to directly access all previous feature maps allowed the model to effectively capture both low-level details and high-level patterns associated with pneumonia, such as lung opacity or localized consolidation. It responded well to controlled fine-tuning and class imbalance mitigation through weighted loss, which enhanced its recall without sacrificing precision. On the other hand, models like ResNet50 and ResNet101 suffered either from insufficient adaptation during frozen-layer training or from overfitting when deeper layers were unfrozen. In the case of ResNet101, fine-tuning led to a recall of 1.00 but disastrous overall accuracy and AUC, revealing the challenges of training very deep models without aggressive regularization and architectural tuning. From a clinical perspective, recall and AUC are more important than pure accuracy. The goal in pneumonia detection is not just to classify correctly on average, but to ensure that actual pneumonia cases are not missed. DenseNet201's recall of 0.77 and AUC of 0.85 suggests that the model is not only capable of identifying most pneumonia-positive cases but also confident in doing so across different thresholds. High recall ensures fewer false negatives, essential for timely treatment, while a high AUC reflects reliable decision-making under variable thresholds. EfficientNetB0, despite having the highest recall of 0.83, was slightly less stable in AUC, raising concerns about specificity in real-world deployment. These distinctions are essential when selecting models for clinical integration, where false negatives carry greater risks than false positives.

Compared to prior studies, including those based on the NIH ChestX-ray14 dataset and models like CheXNet, this study benefits from a cleaner and more focused dataset (RSNA) and a consistent training pipeline. Many past models used weakly labeled data or attempted to solve multi-label classification problems, which often muddled evaluation and interpretation. By focusing on a binary classification task using radiologist-verified labels, the models in this study had the opportunity to learn clearer feature boundaries. Furthermore, this study evaluated multiple architectures under a single, shared setup, which is often missing from prior work. These controlled conditions make the performance differences more meaningful and reproducible.

Despite its strengths, the study faced several limitations. Kaggle's runtime and GPU memory constraints limited batch sizes, total epochs, and the ability to train larger ensembles. For example, the original plan to implement an ensemble learning strategy had to be dropped because training multiple models in parallel led to repeated session timeouts. Similarly, advanced techniques such as Grad-CAM, data augmentation, or bounding box-based localization were not implemented in this version due to feasibility constraints. These omissions limited the interpretability and spatial understanding of the model predictions, although classification performance remained strong. It's also worth noting that the models were validated only on a subset of the RSNA dataset, and external validation using completely different datasets was not performed, which limits generalizability. A number of technical lessons emerged from this research. First, freezing base layers in transfer learning is a double-edged sword. While it allows quick adaptation using pre-trained weights, it may prevent deeper domain-specific features from being learned, particularly in models with aggressive downsampling. Fine-tuning, when done carefully with layer-wise control and reduced learning rates, greatly improved performance in DenseNet201 and Xception. Second, not all models responded equally to class weighting. In EfficientNetB1 and ResNet101, class weighting alone was insufficient to overcome poor convergence or overfitting. Third, preprocessing consistency played a critical , roleespecially with models like EfficientNet, where input size and normalization had to be handled precisely to avoid erratic training behavior. These nuances emphasize the importance of model-specific handling rather than a one-size-fits-all approach.

Looking ahead, several paths exist for extending this work. Visual explanation techniques such as Grad-CAM could be implemented to help clinicians understand what features the models are focusing on during prediction. This would be especially useful for building trust in AI-assisted diagnostics. The bounding box annotations provided in the RSNA dataset could be used for object detection tasks, enabling not just classification but also localization of pneumonia-affected regions. Additionally, the original ensemble idea could be revisited in a more powerful environment to explore whether combined architectures can yield further improvements in recall and AUC. Another potential area of extension is testing the trained models on other datasets such as CheXpert to evaluate generalizability. Ensuring that models perform well beyond their training domain is critical for real-world adoption.

VII. Conclusion and Future Work

Pneumonia remains a major global health concern, with timely and accurate diagnosis being critical to effective treatment and patient recovery. The widespread use of chest X-rays in clinical diagnosis provides an opportunity for deep learning models to assist in detecting pneumonia, especially in settings where radiological expertise is limited. This study explored how various convolutional neural network (CNN) architectures could be applied to the task of pneumonia

detection using chest X-ray images. A comparative analysis was carried out using the RSNA Pneumonia Detection Challenge dataset, chosen for its clinical relevance and high-quality radiologist-verified labels. The goal was to identify a robust model capable of not only classifying images accurately but also minimizing false negatives—a key concern in medical diagnostics.

The methodology involved training a custom CNN model from scratch as a baseline, followed by a series of transfer learning experiments using six widely recognized pre-trained models: ResNet50, ResNet101, EfficientNetB0, EfficientNetB1, Xception, and DenseNet201. Each model was trained under the same preprocessing pipeline and evaluated using consistent metrics, including accuracy, recall, and AUC. These metrics were selected for their diagnostic relevance, particularly recall, which indicates a model's ability to detect actual pneumonia cases. The training was carried out using Kaggle notebooks, where computational constraints played a role in shaping the design and feasibility of each experiment.

Among the models tested, DenseNet201 emerged as the top performer, achieving the best balance across all three evaluation metrics. Its dense connectivity and effective gradient flow allowed for robust feature learning and generalization to validation data. EfficientNetB0 demonstrated high sensitivity, making it useful in high-recall clinical scenarios, but its precision and AUC fell short compared to DenseNet201. ResNet101 and EfficientNetB1, despite their depth, underperformed significantly—often showing instability or overfitting. The baseline CNN served as a useful reference but lacked the architectural capacity to compete with more advanced models. Overall, the results confirmed the value of transfer learning, especially when combined with careful fine-tuning and balanced training. In clinical applications, a model's ability to detect true positives is more valuable than raw accuracy. DenseNet201's strong recall and AUC scores reflect its reliability in detecting pneumonia cases without missing subtle manifestations of the disease. Its stability during training and low rate of false negatives make it a viable candidate for integration into diagnostic pipelines. The study highlights the importance of architectural choice, layer-freezing strategy, learning rate tuning, and class balancing techniques in developing effective AI tools for medical imaging.

Despite the positive results, this study has several limitations that open up directions for future work. The most immediate improvement involves incorporating data augmentation techniques, such as random rotation, flipping, or contrast adjustments, which could help increase model robustness and reduce overfitting. While computational constraints on Kaggle restricted the use of ensemble models, future work could explore model fusion techniques to improve classification stability further. Similarly, explainability methods like Grad-CAM could be applied to visualize which parts of the chest X-ray contributed to a model's prediction, enhancing transparency and clinician trust. The current study focused exclusively on classification. However, the RSNA dataset also includes bounding box annotations that can be leveraged for object detection or localization tasks. Future work could explore detection models that not only classify but also localize pneumonia-affected regions within the lungs. Additionally, segmentation models could be used to provide detailed heatmaps, offering even more insight into disease progression and severity. Such spatial understanding would be valuable in triage, treatment planning, and follow-up assessments.

Another critical direction involves testing the developed models on external datasets like CheXpert or real hospital data to evaluate their generalizability across different populations, equipment types, and clinical settings. Cross-dataset validation would help ensure that the models are not overfitted to the RSNA dataset and are robust enough for real-world deployment. This step is essential for moving from academic research into clinical practice, where model performance must remain consistent across environments.
Lastly, practical deployment pathways should be explored. This includes building lightweight inference pipelines that can be deployed on cloud servers or edge devices like hospital PACS systems or mobile diagnostic units. Collaborating with radiologists to obtain real-world feedback and performing prospective clinical trials could further validate the system's usefulness in everyday practice. With thoughtful integration and continuous improvement, deep learning systems like the ones explored in this study can serve as valuable diagnostic aids in the fight against pneumonia.

Model Variant	Accuracy (%)	Recall
Basic CNN	77.5	0.71
Fine-Tuned CNN	80.76	0.32
Fine-Tuned CNN (Retry)	75.33	0.74
ResNet50 (Frozen)	77.44	0.00
ResNet101 (Frozen)	74.61	0.46
ResNet101 (Fine-Tuned)	22.52	1.00
EfficientNetB0	72.6	0.83
EfficientNetB1	77.48	0.00
Xception	78.34	0.71
DenseNet201 (Final)	**80.0**	**0.77**

Table 1: Performance Metrics of Deep Learning Models for Pneumonia Detection

VIII. REFERENCES

[1] D. S. Kermany, M. Goldbaum, W. Cai et al., "Identifying medical diagnoses and treatable diseases by image-based deep learning," *Cell*, vol. 172, no. 5, pp. 1122–1131.e9, 2018.

[2] P. Rajpurkar, J. Irvin, K. Zhu et al., "CheXNet: Radiologist-level pneumonia detection on chest X-rays with deep learning," *arXiv preprint*, arXiv:1711.05225, 2017.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2012, pp. 1097–1105.

[4] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 2980–2988.

[5] J. Irvin, P. Rajpurkar, M. Ko et al., "CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison," *Proc. AAAI Conf. Artif. Intell.*, vol. 33, pp. 590–597, 2019.

[6] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "ChestX-ray8: Hospital-scale chest X-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 3462–3471.

[7] RSNA, "RSNA Pneumonia Detection Challenge," *Kaggle*, 2018. [Online]. Available: https://www.kaggle.com/c/rsna-pneumonia-detection-challenge

[8] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, 2010.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 770–778.

[10] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2019, pp. 6105–6114.

[11] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 4700–4708.

[12] S. Chouhan, V. K. Singh, A. Khamparia et al., "A novel transfer learning based approach for pneumonia detection in chest X-ray images," *Applied Sciences*, vol. 10, no. 2, p. 559, 2020.

[13] M. T. Islam, M. A. Aowal, A. T. Minhaz, and K. Ashraf, "Abnormality detection and localization in chest X-rays using deep convolutional neural networks," *arXiv preprint*, arXiv:1705.09850, 2017.

[14] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu et al., "Convolutional neural networks for medical image analysis: Full training or fine tuning?" *IEEE Trans. Med. Imaging*, vol. 35, no. 5, pp. 1299–1312, 2016.

[15] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Med. Image Comput. Comput. Assist. Interv. (MICCAI)*, 2015, pp. 234–241.

[16] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 1251–1258.

[17] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 618–626.