DEVELOPMENT AND EVALUATION OF PNEUMFC NET: A NOVEL AUTOMATED LIGHTWEIGHT FULLY CONVOLUTIONAL NEURAL NETWORK MODEL FOR PNEUMONIA DETECTION

SHUBHRA PRAKASH[1] , B RAMAMURTHY[2]

[1]PhD Research Scholar, Department of Computer Science, Christ University Bengaluru, Karnataka, India

[2]Memeber IEEE, Associate Professor, Department of Computer Science, Christ University Bengaluru, Karnataka, India

E-mail: [1]shubhra.prakash@res.christuniversity.in, [2]ramamurthy.b@christuniversity.in

ABSTRACT

The aim of this study is to address the challenges of pneumonia diagnosis under constraint resources and the need for quick decision making. We present the PneumFC Net, a novel architectural solution where our approach focuses on minimizing the number of trainable parameters by incorporating transition blocks that efficiently manage channel dimensions and reduce number of channels. In contrast to using fully connected layers, which disregard the spatial structure of feature maps and substantially increase parameter counts, we exclusively employ only convolutional layer approach. In the study, X-ray image dataset is used to train and evaluate the proposed Convolutional Neural Network model. By carefully designing the architecture, the model achieves a balance between parameters and accuracy while maintaining comparable performance to pre-trained models. The results demonstrate the model's effectiveness in detecting pneumonia images reliably. In addition, the study examines the decision-making process of the model using Grad-CAM, which helps to identify important aspects of radiographic images that contribute to the positive pneumonia prediction. Furthermore, the study shows that the proposed model, Pneum FC Net not only has the highest accuracy of 98%, but the total trainable model parameters is only 0.02% of the next best model VGG-16, thus establishing the potential of this new robust Deep Learning model. This research primarily addresses concerns related to mitigating significant computational requirements, with a specific focus on implementing lightweight networks. The contribution of this work involves the development of resource-efficient and scalable solution for pneumonia detection.

Keywords: *PneumFC Net, Fully Convolutional Neural Network, Pneumonia Detection, Computer Aided Diagnosis*

1. INTRODUCTION

Pneumonia, is one of the leading cause of death among infants globally and contributes significantly to infant mortality rates. Traditional diagnosis of pneumonia involves a combination of physical examination and imaging techniques, primarily chest X-rays and computed tomography (CT) scans. Radiologists interpret these images, relying on their expertise to detect signs of pneumonia and guide patient care decisions. However, this process is subjective, as interpretations can vary among radiologists. Moreover, the reliance on human interpretation has limitations, and the growing demand for diagnostic imaging can strain the healthcare system. The limitations related to human factors, subjectivity, and technical constraints highlight the necessity for more objective, scalable, and efficient methods of interpreting diagnostic imaging in pneumonia diagnosis (1). CNNs are one of the powerful deep learning methods that can easily identify complex patterns from images (2,3) . A typical CNN consists of many layers of artificial neurons, often including different types of layers that perform specific tasks. They are widely used to diagnose and classify diseases through imaging, from skin lesions to brain tumours and, more recently, lung conditions such as pneumonia. However, many challenges remain. One of the obstacles is the computational resources required to train robust CNN models. When dealing with huge datasets or complicated models, training techniques are frequently complicated and time-consuming, requiring advanced hardware that is not always available.

Transfer learning models such as VGGNet, AlexNet, MobileNet, and ResNet have been extensively used for a variety of tasks, including image classification in the medical domain. While these models have proven their efficacy, these model architecture were designed to do multiclass classification on dataset like Imagenet which has more than 100 classes. These models are an overkill for applications where there are only two classes like pneumonia detection and where input image is often grayscale with single channel. Moreover due to the complex network architecture and more number of channels the size of the models are heavy which in turn requires good computational resources. Hence there is a growing need for lightweight models with fewer parameters, especially for tasks like X-ray image classification where there are resource constraints. In this article, we explore a novel approach to designing a lightweight X-ray image classification model that leverages the concept of progressive dimension reduction and channel count reduction through the use of max-pooling and point-wise convolution.

Additionally, CNN models tend to have a lack of detail behind the decisions, which means that although they can process images well, understanding the reasoning behind their decisions can be confusing. This kind of "black box" of CNN can make it difficult for them to be accepted by medical professionals. Despite these challenges, the success of CNN models in detecting pneumonia cannot be underestimated. They have consistently demonstrated the ability to provide high-quality analytics, quick decision-making, and the ability to handle large amounts of data.

The main idea of this study is that a compact and robust CNN model can provide predictions equal to or even higher than existing methods, even with a much smaller size. Our efforts will explore this hypothesis and address critical questions regarding the balance between model complexity and diagnostic accuracy.

The following are the manuscript's main contributions:

• A fully convolutional CNN model was trained for the classification of Pneumonia

• The input images for the proposed method were gathered from different X-ray image datasets and suitable pre-processing was performed

• Experiments were performed to come up with an architecture with 5 blocks similar to contemporary CNN architectures but with careful selection of number of channels and usage of transition blocks to keep the channels dimension and count in check at the same time retain as much spatial information as required for accurate prediction

• The classification performance of the proposed model was compared with transfer learning models (trained on same dataset) like VGG and ResNet to prove its efficacy while having an apples to apples comparison

• GradCAM method was used to attribute the predictions of the model back to individual pixels

• Another comparison of the result was performed with methods proposed in related works

The remaining sections are structured as follows: Section 2 shows results of literature survey. The proposed model, components of the model and architecture with model summary containing parameter count for each layer are described in Section 3. The results are presented and discussed in Section 4. Section 5 talks about future scope in section 6 concludes the manuscript [9].

2. RELATED WORKS

Khan et al. (4) proposed a lightweight Convolutional Neural Network (CNN) architecture designed for the detection of COVID-19 cases in X-ray and CT scan images. The model consisted of seven convolutional layers, which made it efficient and required less computational power during training. Each hidden layer utilized the Rectifier linear unit (ReLU) activation function to handle negative inputs and avoid vanishing gradients. A pooling layer was placed in the initial layers to determine the maximum value for the pixels. Following convolutional layers had 256 filters which subsequently reduced to 128, 64, and 32 filters. A dropout layer of 0.5 was inserted between two fully connected layers, and an output layer with softmax for categorical classification and sigmoid for binary classification was added after that.

Asif et al. (5) presented a shallow convolutional neural network (CNN) model and assessed its effectiveness in comparison to established transfer learning models such as Inception V3, Xception, MobileNet, NASNet, and DenseNet201. The suggested model exhibited significant improvements over the transfer learning counterparts; however, it featured a shallow architecture and incorporated a fully connected layer for classification.

Alduaiji et al. (6) proposed a CNN model design which used images of size 256x256x3. The CNN structure included four convolution layers with increasing filters (64, 64, 128, 256) and 2x2 kernels. The architecture integrated GAP as the final layer, enhancing class-specific activation maps and reducing overfitting risk. The model had 32.95 million parameters, where the major contribution came from dense layer at the end.

Nayak et al. (7) proposed an automated light CNN model where they used a GAP layer towards the end of the architectute. In the study the model was compared with other pretrained model. The model showed good AUROC score in comparison to other models but the ResNet model had better accuracy and F1 score.

Senan et al. (8) proposed an architecture where, output features from a light weight CNN model were combined with texture based features to enhance pneumonia detection. A dense layer was used at the end of the architecture for the classification task.

Souid et al. (9) used a modified MobileNet V2 architecture for thoracic radiographic predictions. The study utilized transfer learning with metadata, the study employed the NIH Chest-Xray-14 database, comparing their method's performance using AUC statistics. The average AUC was 0.811 with over 90% accuracy.

Oh et al. (10) in their study propose a patch-based deep neural network model, trained on small datasets, and decisions based on larger polls from random lung patches, and introduce a Grad-CAM saliency map for interpretation details

Khan et al. (11) developed a model called CoroNet, based on the Xception architecture and trained on a dataset of COVID-19 and Pneumonia x-ray images, achieved an accuracy of 89.6%, with a precision and recall rate of 93% and 98.2% for COVID-19.

Ozturk et al. (12) proposed a new model in their study for the automatic diagnosis of COVID-19, which achieved an accuracy of 98.08% for the classification of two variables (COVID vs No-Findings).

Nikolaou et al. (13) in their research developed a hybrid CNN using the EfficientNetB0 as a baseline model. The EfficientNetB0 was chosen due to its parameter efficiency, cost-effectiveness, and accurate classification of COVID-19 cases. The model included a fully dense layer for feature extraction. With around 5 million parameters, the CNN was faster and less likely to overfit, aided by 20% and 50% dropout rates. These actions improved the model's functionality while lowering the chance of overfitting during training and validation.

A recent study conducted by Hussein et al. (14) focuses on creating a lightweight convolutional neural network (CNN) that makes use of both convolutional and fully connected layers. Rather than depending more on the feature extractor, this method restricts the depth of the network and assigns the classification responsibility to the fully connected layer for categorization. Another relevant investigation, undertaken by Shi et al. (15) focuses on using chest X-ray (CXR) image data and makes use of a customized four-layer network with 11 × 11 or 3 × 3 kernels. Interestingly, they provide a pruning criterion that is weight-based in order to maximize network efficiency. Even with the pruning strategy used in this work to remove connections, there is still not enough depth in the network to enable reliable feature extraction.

In conclusion, it is clear that the development of a novel CNN model is necessary. When doing predictive tasks, this model should give feature extraction precedence over a strong dependence on the fully connected layers. Reducing parameters is also an important way to optimize the model because it improves computational efficiency. As a result, the optimal approach should keep model parameters minimal to guarantee usefulness and effective use of resources. This method will not only improves the suggested CNN model's feature extraction performance but also makes it more appropriate for practical application.

3. MATERIALS AND METHODS

X-ray images, which serve as the dataset for model construction and validation is used in this study. These images were obtained from a medical institution in China, and they show different manifestations of pneumonia. We have developed a simple CNN model designed to handle the complexity of pneumonia detection. In order to manage the trade-off between model size and diagnostic precision while taking into account the constrained resources of the actual clinical context, this model was carefully built. We partitioned our data set into separate portions for training, validation, and testing in order to assess the model performance. Along with the development and evaluation of the model, we also analyzed the details of the model's understanding of problem using visualization tools Grad-CAM by highlighting important areas in the X-ray image that affect the prediction of the model, thus providing insight into the inner workings of the model and improving reliability.

3.1 Dataset

The dataset used in this study consists of X-ray images of the chest, which are divided into two groups: pneumonia and normal. The database was provided by Guangzhou Children's Hospital in Guangzhou, China (16). A pneumonia group has images that show different levels and severity of the disease, while a normal group includes images without pneumonia. This data collection, given its source from a well-known hospital and its variety, serves as a reliable basis for the development and evaluation of our simple CNN model for the analysis of pneumonia. In study, the images used had varying dimensions, although the majority measured 255x255 pixels with a bit depth of 24, and they were all in PNG format.

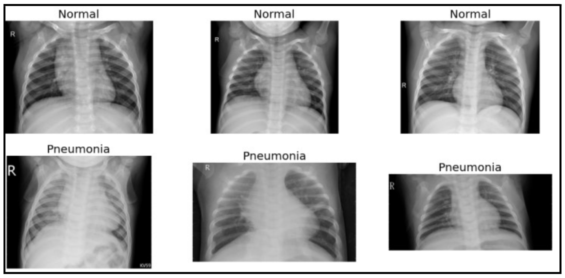

Figure 1: Sample images from dataset

Table 1: Train and test split for dataset

Class label	Training	Testing
Normal	1340	3874
Pneumonia	224	241

3.2 Preprocessing

In the study all the images were resized by scaling them to 224 x 224 pixels, increasing the consistency of the dataset and making model training easier and the results comparable. Finally, we apply image normalization, which helps to bring consistency in pixel values across different images and enhances model's ability to learn meaningful patterns. The normalized pixel values are calculated by following formula (17):

normalized_value = (pixel_value - mean_value) / standard_deviation (1)

3.3 Deep Neural Network Classifier

The proposed lightweight CNN model is designed to classify pneumonia cases into two categories: pneumonia and normal. Its primary objective is to detect pneumonia in patients using chest X-ray images while addressing the constraints of computational power and memory requirements. The parts that follow will give a general overview of this CNN model and go into detail on its construction and evaluation of results.

3.3.1 Convolutional Layer

The CNN model's core layers, which extract features, employ a collection of filters (kernels) with fixed sizes (such as 3x3, 5x5, or 7x7). To find pertinent features, these filters are convolved over the input images. Each block in the input image must match the size of the filter in order to complete the convolution operation, which entails element-wise multiplying each block by the filter. A feature map, which is the output matrix's matching position, is created by adding the results to create a single output value. Different filters are used to create a variety of feature maps and each represent a different aspect of the original image. An activation function like ELU or ReLU is used to add nonlinearity to the convolution operation's output (18).

3.3.2 Batch Normalization

Batch normalization is a critical technique that addresses the issue of internal covariate shift. It involves normalizing the intermediate feature maps within a batch of training samples to have zero mean and unit variance. By applying BatchNorm, the network becomes less sensitive to variations in the distribution of inputs, leading to improved training stability and faster convergence. The BatchNorm layer is typically inserted after the convolutional or fully connected layers in a CNN. BatchNorm has been shown to enhance the performance of CNNs by reducing the effects of internal covariate shift, enabling more effective gradient flow during back propagation, and improving the generalization capability of the network (19).

We can define the normalization formula of Batch Norm as:

$$Z^n = (\frac{Z - m_z}{s_z}) \qquad (2)$$

where m_z_z and $s_z$zrepresent the mean and standard deviation

3.3.3 Exponential Linear Unit (ELU)

Exponential Linear Unit (ELU) is an activation function commonly used in Convolutional Neural Networks (CNNs) to introduce nonlinearity. ELU addresses the limitations of ReLU by mitigating the "dying ReLU" problem, where neurons can become non-responsive and produce zero outputs for negative inputs, leading to dead activation units. ELU overcomes this issue by allowing negative values, which can help preserve gradient flow and improve network learning (20).

The ELU activation function is defined as follows:

$$f(x) = \{x, if\ x > 0\ and \\ -\alpha X (exp(-x) - 1), if\ x < 0\} \tag{3}$$

3.3.4 Max Pooling

Convolutional neural networks (CNNs) frequently use the downsampling method known as max pooling to minimise the spatial dimensions of feature maps while retaining the most important features. To construct downsampled feature maps, it divides the feature maps into non-overlapping rectangular regions (often with a stride) and chooses the maximum value inside each zone. The input feature map is traversed by a sliding window during the max pooling procedure. The position in the output feature map that corresponds to each value is taken from the window's maximum value. The strongest and most representative traits are effectively kept while using max pooling, while less important aspects are eliminated (21).

3.3.5 Global Average Pooling

Global Average Pooling (GAP) is a technique used in Convolutional Neural Networks (CNNs) for spatial downsampling and dimensionality reduction of feature maps. GAP accomplishes pooling over the whole spatial dimensions of the feature maps, in contrast to max pooling or average pooling, which work on discrete portions of the feature maps. The feature maps are initially spatially averaged in the GAP procedure by averaging each feature map's width and height. The feature maps' spatial dimensions are reduced to a single value per channel by this pooling technique. The final results After the a few convolutional layer, a transition layer, is introduced. This layer performs max pooling with a kernel size of 2 and stride of 2, reducing the spatial dimensions of the feature maps. Followed by a 1x1 Convolution to reduce the number of channels (as shown in figure 2) and are then entered into classification tasks or used as features for later layers. GAP can capture global context and lessen the impact of local fluctuations or noise by averaging the entire feature map. Also, GAP provides a degree of spatial invariance, as it discards specific spatial information and emphasizes the overall distribution of features. One notable characteristic of GAP is its ability to generate a fixed-length feature vector regardless of the input size or spatial dimensions of the feature maps. This makes GAP particularly suitable for tasks like image classification, where the network's output is a fixed-size vector representing the image's class probabilities (22).

3.3.6 Softmax and Negative Log Loss

Softmax is an activation function that converts a vector of real numbers into a probability distribution. Negative Log Loss is a commonly used loss function to measure the dissimilarity between predicted probabilities and the true labels. In the context of classification tasks, the negative log loss calculates the average logarithmic loss over all samples in the dataset. The negative log loss penalizes incorrect predictions more strongly, encouraging the network to learn accurate class probabilities. Minimizing the negative log loss during training helps the network converge towards the optimal set of weights that produce accurate predictions.

3.3.7 Architecture of Proposed Model

The proposed CNN model is designed to perform image classification tasks. It consists of multiple convolutional layers, pooling layers, and a GAP layer. The model architecture is implemented in Python using the PyTorch library. The model follows a squeeze and expand way of adding channels and consists of two major groups of layers namely convolutional layers and Transition Layers as shown in figure 3. Each convolutional layer has some input channels on which a kernel of fixed size is convolved and is followed by batch normalization and activation function (ELU). In these convolutional layers, we have flexibility in specifying the number of input and output channels, kernel size (default:3x3), stride(default:1), and padding(default:0).

is then followed by batch normalization and the ELU activation function.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

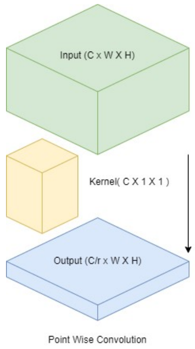

Figure 2: Point wise convolution

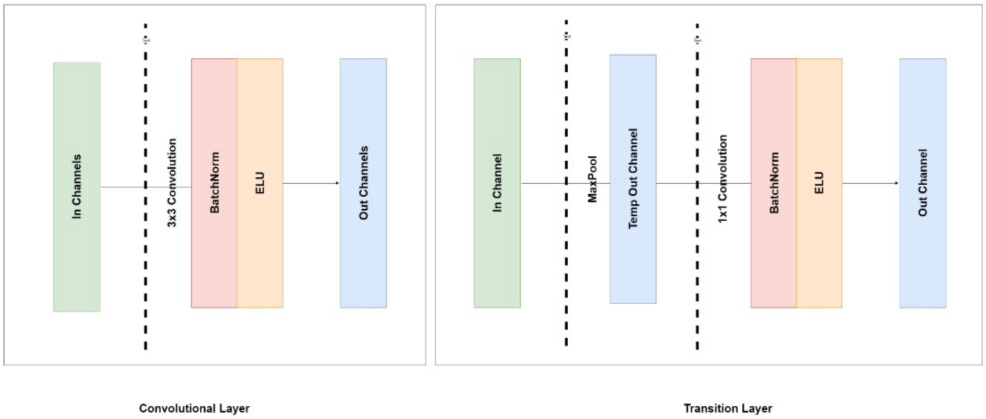

Figure 3: Model architecture: convolution and transition layer

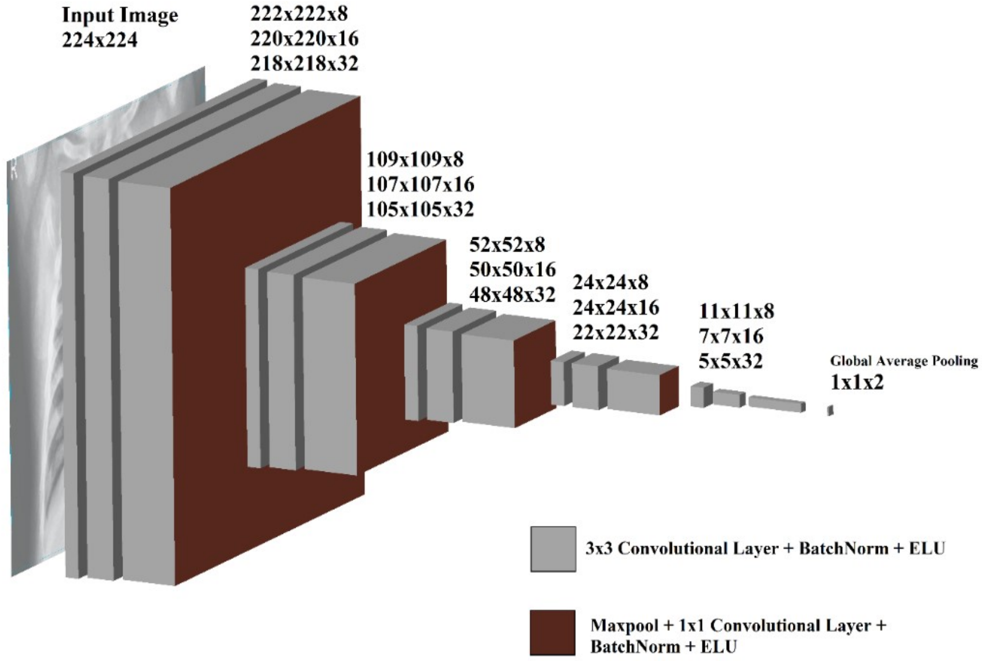

Figure 4: Proposed architecture of the proposed CNN model

Table 2: Model summary of the proposed cnn model for pneumonia detection

Layer	Kernel Size	BatchNorm	Activation	Input Shape	Output Shape	Parameters	Block
Input	-	-	-	-	[3, 224, 224]	0	
Conv2d	3x3	Yes	ELU	[3, 224, 224]	[8, 222, 222]	216 + 16	1
Conv2d	3x3	Yes	ELU	[8, 222, 222]	[16, 220, 220]	1152 + 32	1
Conv2d	3x3	Yes	ELU	[16, 220, 220]	[32, 218, 218]	4608 + 64	1
MaxPool2d	2x2	-	-	[32, 218, 218]	[32, 109, 109]	0	1
Conv2d	1x1	Yes	ELU	[32, 109, 109]	[8, 109, 109]	256 + 16	1
Conv2d	3x3	Yes	ELU	[8, 109, 109]	[16, 107, 107]	1152 + 32	2
Conv2d	3x3	Yes	ELU	[16, 107, 107]	[32, 105, 105]	4608+64	2
MaxPool2d	2x2	-	-	[32, 105, 105]	[32, 52, 52]	0	2
Conv2d	1x1	Yes	ELU	[32, 52, 52]	[8, 52, 52]	256 + 16	2
Conv2d	3x3	Yes	ELU	[8, 52, 52]	[16, 50, 50]	1152 + 32	3
Conv2d	3x3	Yes	ELU	[16, 50, 50]	[32, 48, 48]	4608+64	3
MaxPool2d	2x2	-	-	[32, 48, 48]	[32, 24, 24]	0	3
Conv2d	1x1	Yes	ELU	[32, 24, 24]	[8, 24, 24]	256 + 16	3
Conv2d(with Padding)	3x3	Yes	ELU	[8, 24, 24]	[16, 24, 24]	1152 + 32	4
Conv2d	3x3	Yes	ELU	[16, 24, 24]	[32, 22, 22]	4608+64	4
MaxPool2d	2x2	-	-	[32, 22, 22]	[32, 11, 11]	0	4
Conv2d	1x1	Yes	ELU	[32, 11, 11]	[8, 11, 11]	256 + 16	4
Conv2d	3x3	Yes	ELU	[8, 11, 11]	[16, 7, 7]	3200+32	5
Conv2d	3x3	Yes	ELU	[16, 7, 7]	[32, 5, 5]	4608+64	5
AvgPool2d	5x5	-	-	[32, 5, 5]	[32, 1, 1]	64	5
Conv2d	1x1	Yes	ELU	[32, 1, 1]	[2, 1, 1]	64	5
					Total Parameters	32712	

The proposed architecture begins with a 224x224 image and employs a sequence of three convolutional layers with outputs of 8, 16, and 32 channels. These convolutional layers, depicted as grey blocks, perform feature extraction on the input image. The width of the blocks represents the number of channels, while the dimensions of the image are denoted length and breadth of the block. As a result of these three layers, the image size reduces to 218x218 with 32 channels. Subsequently, a transitional layer is applied, which incorporates maxpooling to downsample the feature maps while simultaneously reducing the depth back to 8. This transitional layer prepares the architecture for another cycle of increasing depth, accomplished through subsequent convolutional layers with outputs of 16 and 32 channels. This cycle is repeated four times until the image size decreases to 11x11 with 8 channels.

Following these repeated convolutional layers, another stack of convolutional layers is applied. However, instead of using a transitional layer, a global average pooling technique is employed. This global average pooling operation calculates the average value for each feature map, resulting in a fixed-length vector representation of the input, specifically 1x1x2. This vector is then reshaped and fed into a softmax function, which produces the final probability distribution over the classes. The step by step operation and parameter calculation is shown in the model summary as shown in table II.

As it is evident from the architecture that the model only uses convolutional layers that prioritizes feature extraction over reliance on fully connected layers for predictive tasks. Additionally, the total parameter count is only 32712 which makes this model very light weight.

3.3.8 Model Training and Testing

For training and testing firstly, the dataset is divided into two parts: training and testing data. PyTorch's DataLoader module is utilized to efficiently load and preprocess the data. In the training phase, the CNN model is trained using the training data. The chosen optimizer is 'SGD' (Stochastic Gradient Descent), which updates the model parameters to minimize the loss function during backpropagation. The learning rate is set to 0.01, allowing the optimizer to control the step size during parameter updates. Additionally, a momentum value of 0.9 is specified, which helps accelerate convergence during training. The training process is performed iteratively over a fixed number of epochs (in this case, 20). Each epoch consists of multiple batches, and the batch size is set to 32. The model is trained by feeding the batches through the network, computing the loss, and backpropagating the gradients to update the model parameters. This process continues until all epochs are completed. Finally, the trained model is evaluated using the testing data. The test data is fed through the model, and the predicted outputs are compared with the ground truth labels.

Table 3: Hyperparameters for model training

Hyperparameter	Value
Optimizer	SGD
Learning Rate	0.01
Momentum	0.9
Epochs	20
Batch Size	32

3.3.9 Performance Evaluation

In this investigation, assessing the performance of the suggested model forms a crucial aspect as it verifies the model's ability to diagnose pneumonia accurately. Primarily, the model's accuracy is determined, quantified as the proportion of correct predictions relative to total predictions made. Nevertheless, considering the potential inaccuracies in datasets, the evaluation also employs precision, recall, F1 scores as reliable metrics.

4. RESULTS AND DISCUSSION

For this section, a thorough evaluation was conducted on the proposed lightweight CNN models for Pneumonia detection using commonly used performance measures. The implementation of the models was carried out in Python programming language. The results obtained from the proposed models are reported and discussed in the subsequent subsections. Furthermore, a comparative analysis is presented, highlighting the differences between our proposed model and the existing models.

4.1 Accuracy and Loss vs Epochs

Throughout the training phase of a given classifier algorithm, accuracy and loss metrics are commonly used to assess the performance of the model over successive epochs. These metrics serve the purpose of controlling overfitting and understanding the state of predictions. It is generally assumed that accuracy and loss are inversely related, meaning that lower loss values correspond to higher accuracy values.

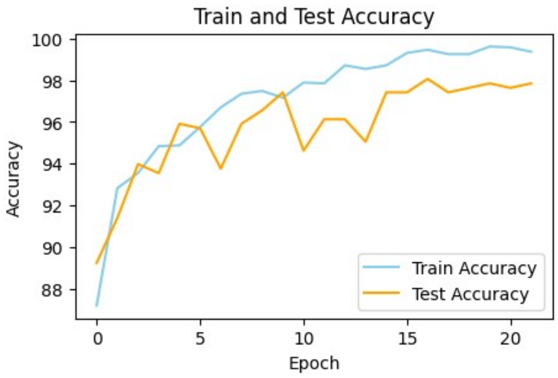

Figure.5: Train and test accuracy over epochs

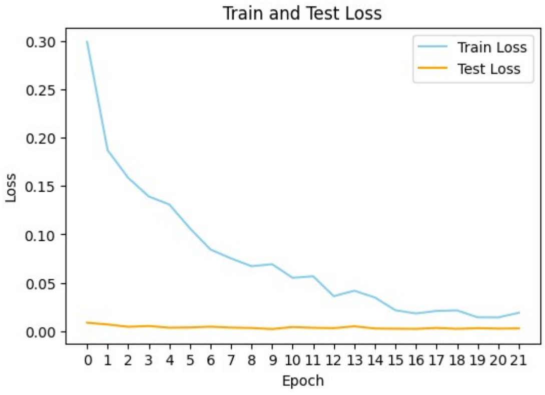

Figure 6: Train and test loss over epochs

4.2 Confusion Matrix Analysis

The confusion matrix reveals that out of 224 instances where the true label was 0, the model correctly predicted 221 of them as 0 (TP), but erroneously classified 3 instances as 1 (FP). Similarly, out of 241 instances where the true label was 1, the model correctly predicted 234 as 1 (TP) but misclassified 7 instances as 0 (FN).

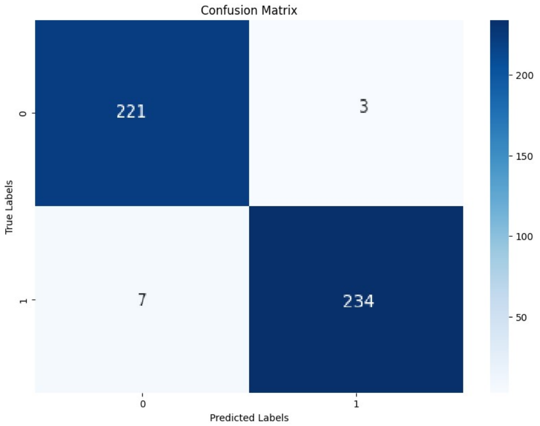

Figure 7: Confusion matrix

4.3 Receiver Operating Characterstic (ROC) Curve

As seen in the graph below AUC of the ROC curve is close to 1, it indicates that the model has excellent discriminative ability and is able to effectively separate the positive and negative instances which is highly desirable

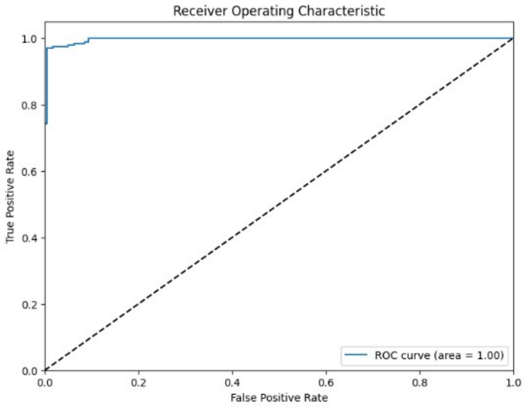

Figure 8: ROC Curve

4.4 GradCAM Output

Shown below are the GradCAM output for the Pneumonia Images. The highlighted region show the importance of the region in prediction of pneumonia.

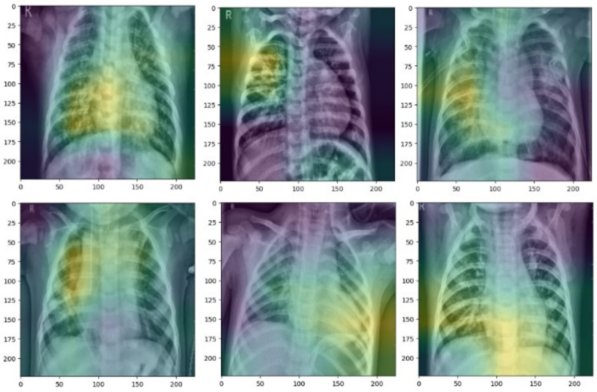

Figure 9: GradCAM output

4.5 Model Performance

In the table 4 the accuracy, precision, recall, F1 score, and model parameters of different models have been compared. These transfer learning models were trained on the same data as Pneum FC Net. It can be observed that the Pneum FC Net model stands out as an accurate and compact model compared to others. In terms of accuracy, the Pneum FC Net achieves an accuracy of 0.9806, which is higher than all the other models. This indicates that the Pneum FC Net model performs exceptionally well in accurately classifying pneumonia cases. Additionally, the proposed model demonstrates competitive precision, recall, and F1 scores, showcasing its ability to maintain a good balance between correctly identifying positive instances (precision) and capturing all relevant positive instances (recall).

Table 4: Comparison of proposed model with transfer learning model (size / parameter count in millions)

Model	Accuracy	Precision	Recall	F1	Size
Pneum FC Net (Proposed)	0.98	0.98	0.98	0.98	0.032 M
MobileNet V2	0.96	0.96	0.96	0.96	22.26 M
Resnet34	0.95	0.95	0.95	0.95	21.28 M
AlexNet	0.97	0.97	0.97	0.97	57.01 M
VGG-16	0.97	0.97	0.97	0.97	134.26 M

The distinguishing feature of the Pneum FC Net model lies in its notably reduced parameter count. With only 32,712 parameters, it surpasses all other models under consideration in terms of compactness. This advantage becomes particularly evident when comparing it to larger models such as AlexNet with 57,012,034 parameters, VGG-16 with 134,268,738 parameters, and even MobileNetV2 with 2,226,434 parameters. Remarkably, the proposed model achieves commendable accuracy levels while maintaining significantly smaller model size.

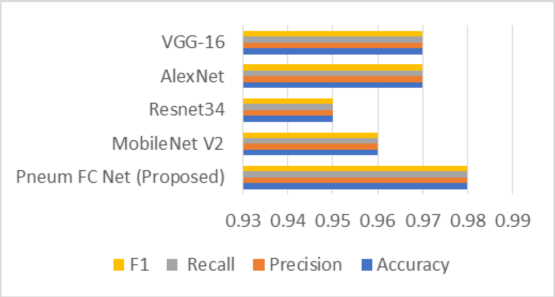

Figure 10: Comparison of proposed model with transfer learning models

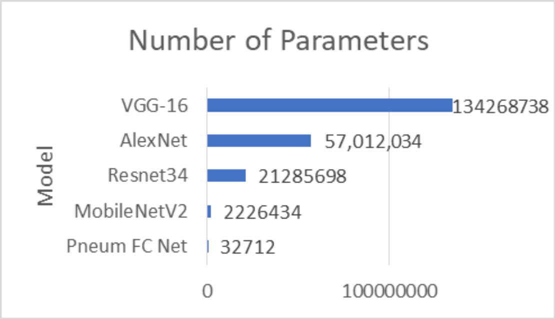

Figure 11: Parameter count for models

4.6 Model Comparison

In this section, we evaluate the performance of our proposed model in relation to existing CNN based approaches in the field for pneumonia detection. Our study demonstrates notable advancements, albeit with modest improvements, over the other studies.

Table 5: Model Comparison

	Accuracy	Precision	Recall	F1Score
Pneum FC Net	0.98	0.981	0.98	0.98
Souid et al. (9)	0.94	0.991	0.38	0.558
Oh et al. (10)	0.889	0.834	0.859	0.844
Khan et al. (11)	0.896	0.90	0.899	0.898
Ozturk et al. (12)	0.87	0.89	0.85	0.87

In summary, the thorough evaluation of our proposed lightweight CNN models for Pneumonia detection has provided valuable insights into their performance using a range of performance measures including accuracy and loss versus epochs, confusion matrix analysis, ROC curve, GradCAM output, and comprehensive model comparisons. Notably, the Pneum FC Net model emerged as a standout performer, achieving an accuracy of 0.9806 and demonstrating remarkable precision, recall, and F1 scores. The model's distinctiveness lies in its significantly reduced parameter count, exemplified by its 32,712 parameters, which surpasses larger models while maintaining commendable accuracy levels. The model comparison against existing CNN-based approaches showcased notable advancements, signaling the potential of our approach to enhance the effectiveness of medical image classification in the context of pneumonia detection. These findings contribute to the ongoing discourse in the field and underscore the significance of our proposed model in achieving accurate and compact solutions for medical image classification tasks.

5. CONCLUSION

In conclusion, our research has effectively addressed the pressing need for an advanced lightweight pneumonia detection model. Recognizing the limitations of existing models, we undertook the development of a fully convolutional neural network (CNN) which prioritizes feature extraction over heavy reliance on the fully connected layer and meets the computational efficiency requirements. Our chosen method involved a meticulous process of model development and evaluation. Our CNN model showcased superior performance, outperforming pre-trained models across various metrics crucial for pneumonia detection. The need for a reliable and efficient solution was met through the successful implementation of our approach. The results obtained from our model, supported by receiver operating characteristic (AUC-ROC) analysis, demonstrated its robust discriminatory capabilities, effectively distinguishing between pneumonia and non-pneumonia cases. This outcome not only fulfills the primary objective of accurate detection but also adds a layer of confidence in the model's diagnostic abilities. In aligning with the need for interpretability in AI systems, we incorporated the Gradient-Weighted Class Activation Mapping (Grad-CAM) method. This strategic inclusion allows for a visual understanding of the model's decision-making process, serving as a crucial bridge between artificial intelligence and human expertise. In essence, our research journey, driven by the identified need, navigated through methodical model development, and culminated in outstanding results. The implications can extend beyond pneumonia detection, offering a promising outlook for the broader field of medical AI.

REFERENCES:

[1]. Labhane G, Pansare R, Maheshwari S, Tiwari R, Shukla A. Detection of Pediatric Pneumonia from Chest X-Ray Images using CNN and Transfer Learning. In: 2020 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things (ICETCE). 2020. p. 85–92.

[2]. Taha AM, Ariffin D, Abu-Naser SS. A Systematic Literature Review of Deep and Machine Learning Algorithms in Brain Tumor and Meta-Analysis. Journal of Theoretical and Applied Information Technology. 2023;101(1):21–36.

[3]. Alkayyali ZK, Idris S, Abu-Naser SS. A Systematic Literature Review of Deep and Machine Learning Algorithms in Cardiovascular Diseases Diagnosis. Journal of Theoretical and Applied Information Technology. 2023;101(4):1353–65.

[4]. Khan HA, Gong X, Bi F, Ali R. Novel Light Convolutional Neural Network for COVID Detection with Watershed Based Region Growing Segmentation. Journal of Imaging. 2023 Feb;9(2):42.

[5]. Asif S, Zhao M, Tang F, Zhu Y. A deep learning-based framework for detecting COVID-19 patients using chest X-rays. Multimedia Systems. 2022 Aug 1;28(4):1495–513.

[6]. Alduaiji N, Algarni A, Abdalaha Hamza S, Abdel Azim G, Hamam H. A Lightweight CNN and Class Weight Balancing on Chest X-ray Images for COVID-19 Detection. Electronics. 2022 Jan;11(23):4008.

[7]. Nayak SR, Nayak J, Sinha U, Arora V, Ghosh U, Satapathy SC. An Automated Lightweight Deep Neural Network for Diagnosis of COVID-19 from Chest X-ray Images. Arab J Sci Eng. 2023 Aug 1;48(8):11085–102.

[8]. Senan EM, Alzahrani A, Alzahrani MY, Alsharif N, Aldhyani THH. Automated Diagnosis of Chest X-Ray for Early Detection of COVID-19 Disease. Comput Math Methods Med. 2021;2021:6919483.

[9]. Souid A, Sakli N, Sakli H. Classification and Predictions of Lung Diseases from Chest X-rays Using MobileNet V2. Applied Sciences. 2021 Jan;11(6):2751.

[10].Oh Y, Park S, Ye JC. Deep Learning COVID-19 Features on CXR Using Limited Training Data Sets. IEEE Transactions on Medical Imaging. 2020 Aug;39(8):2688–700.

[11].Khan AI, Shah JL, Bhat MM. CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest x-ray images. Computer Methods and Programs in Biomedicine. 2020 Nov 1;196:105581.

[12].Ozturk T, Talo M, Yildirim EA, Baloglu UB, Yildirim O, Rajendra Acharya U. Automated detection of COVID-19 cases using deep neural networks with X-ray images. Computers in Biology and Medicine. 2020 Jun 1;121:103792.

[13].Nikolaou V, Massaro S, Fakhimi M, Stergioulas L, Garn W. COVID-19 diagnosis from chest x-rays: developing a simple, fast, and accurate neural network. Health Inf Sci Syst. 2021 Oct 12;9(1):36.

[14].Hussein HI, Mohammed AO, Hassan MM, Mstafa RJ. Lightweight deep CNN-based models for early detection of COVID-19 patients from chest X-ray images. Expert Syst Appl. 2023 Aug 1;223:119900.

[15].Shi Y, Tang A, Xiao Y, Niu L. A lightweight network for COVID-19 detection in X-ray images. Methods. 2023 Jan;209:29–37.

[16].Kermany DS, Goldbaum M, Cai W, Valentim CCS, Liang H, Baxter SL, et al. Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning. Cell. 2018 Feb 22;172(5):1122-1131.e9.

[17].Gibson E, Li W, Sudre C, Fidon L, Shakir DI, Wang G, et al. NiftyNet: a deep-learning platform for medical imaging. Computer Methods and Programs in Biomedicine. 2018 May 1;158:113–22.

[18].Alzubaidi L, Zhang J, Humaidi AJ, Al-Dujaili A, Duan Y, Al-Shamma O, et al. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. Journal of Big Data. 2021 Mar 31;8(1):53.

[19].Ismael AM, Şengür A. Deep learning approaches for COVID-19 detection based on

chest X-ray images. Expert Systems with Applications. 2021 Feb 1;164:114054.

[20].Mittal A, Kumar D, Mittal M, Saba T, Abunadi I, Rehman A, et al. Detecting Pneumonia Using Convolutions and Dynamic Capsule Routing for Chest X-ray Images. Sensors. 2020 Jan;20(4):1068.

[21].Uddin A, Talukder B, Monirujjaman Khan M, Zaguia A. Study on Convolutional Neural Network to Detect COVID-19 from Chest X-Rays. Mathematical Problems in Engineering. 2021 Sep 11;2021:e3366057.

[22].Yamashita R, Nishio M, Do RKG, Togashi K. Convolutional neural networks: an overview and application in radiology. Insights Imaging. 2018 Aug 1;9(4):611–29.