Computer Vision and Deep Transfer Learning Techniques in Creating an End-To-End Pipeline for The Identification of Pneumonia from Chest X-Ray

Priya L[1], Sathya.A[2], Prasannakumari.V[3]
Professor [1 and 3] Associate professor [2]
Department of Information Technology
Rajalakshmi Engineering college
Priya.l@rajalakshmi.edu.in
+91 9841036875

Abstract— Pneumonia is a respiratory illness characterized by lung inflammation, leading to air sacs filling with fluid or pus. Symptoms include coughing with pus or phlegm, fever, chills, and breathing difficulties. To improve diagnostic decision-making, a research study proposes a deep transfer learning approach using the VGG16 convolutional neural network algorithm for interpreting chest X-rays.Transfer learning is an optimization technique that allows models to quickly adapt and perform better by leveraging pre-existing knowledge. In this case, VGG16 is trained on a large dataset of general images, and then applied to the task of classifying chest X-rays into two groups: pneumonia-related alterations or no pneumonia. The aim is to enhance the accuracy and speed of pneumonia detection.The research anticipates that the deep transfer learning model based on VGG16 will outperform current state-of-the-art approaches and popular ensemble techniques. By utilizing transfer learning, the model can effectively utilize the learned weights and knowledge from the general image dataset, enabling it to rapidly adapt to the specific task of pneumonia detection. This paper aims to optimize and evolve the method more efficiently, leading to improved diagnostic outcomes for pneumonia detection from chest X-rays.

IndexTerms:* Pneumonia Detection, CNN, Deeplearning; Transferlearning;VGG16 (key words)

I. INTRODUCTION

In order to improve child health, it is essential to have effective diagnosis and treatment methods for pneumonia. Artificial intelligence (AI) and computer vision, particularly deep learning, have been increasingly utilized in various industries, including the medical field. The availability of vast amounts of data has facilitated the use of AI in making better, faster, and more consistent decisions.

Deep learning and AI have shown promise in medical fields that heavily rely on biomedical images and require large-scale image processing. Chest X-rays are one of the most commonly used medical imaging techniques due to their low cost and relatively low radiation exposure. However, the high volume of X-ray images generated daily in hospitals poses a significant diagnostic workload. In the UK's public healthcare system alone, over 22.5 million X-ray images were requested in 2019–20, surpassing other imaging modalities such as computed tomography (CT) and magnetic resonance imaging (MRI).

Pneumonia remains a major cause of child mortality, with 740,180 children under the age of 5 dying from the disease in 2019. Fortunately, pneumonia can be treated with low-cost medications and care. However, there is a shortage of radiologists, particularly in developed countries. Additionally, X-rays have limitations in terms of accuracy compared to more advanced imaging technologies like CT and PET scans, mainly due to their lower resolution and 2D projection.

To address these challenges, this research focuses on the application of deep transfer learning using the VGG16 convolutional neural network algorithm to interpret chest X-ray images. By leveraging transfer learning, the model can benefit from pre-existing knowledge and weights learned from a large dataset of general images, enabling it to quickly adapt to the specific task of pneumonia detection from X-ray images. The objective is to enhance the decision-making process for accurate diagnosis, aiding healthcare professionals in providing timely and effective treatment for pneumonia.

By utilizing deep learning and transfer learning techniques, the research aims to improve the accuracy and efficiency of X-ray interpretation, ultimately contributing to better child health outcomes in the diagnosis and treatment of pneumonia.

II. CONTRIBUTIONS

The contributions of this work address several significant challenges present in existing pneumonia detection systems, such as the vanishing gradient problem, excessive resource consumption, lengthy training processes, and moderate accuracy. Many existing systems have not adequately considered these drawbacks, prompting the development of a deep learning model for pneumonia classification using convolutional neural network (CNN) algorithms. The primary objective is to achieve the highest accuracy by comparing different CNN architectures.

This research focuses on the classification of pneumonia-defected and normal images using CNN methods. The proposed system addresses the issue of vanishing gradients, which hinder the backpropagation of gradient information to the input layers of the model in multi-layer CNNs. To mitigate this problem, the VGG16 model is utilized, where only the last layer is necessary for execution, and the other layers can be disabled. This approach helps reduce resource consumption by employing a single GPU instead of distributing the model's neurons across multiple GPUs as done in the existing system. Additionally, the training process is accelerated, saving significant time.

The results of this research demonstrate a substantial improvement in accuracy, from 60.90% to 98.14%. By addressing the aforementioned challenges, the proposed system provides a more efficient and accurate approach to pneumonia detection. This advancement has the potential to enhance diagnostic capabilities, facilitate timely treatment decisions, and ultimately improve patient outcomes.

III. RelatedWork

The use of deep learning algorithms in medical image classification, specifically for disease identification, has become increasingly popular in research. Several studies have explored the application of these algorithms in pneumonia and COVID-19 detection.

Xiouyang et al. proposed a method utilizing a 3D convolutional neural network to detect COVID-19 from community-acquired pneumonia in the early stages. Their approach aimed to improve detection accuracy.

Muhammad et al. investigated the use of pre-trained deep learning methods in detecting COVID-19 through chest X-ray images. Their study demonstrated higher detection accuracy compared to other methods.

Sing-Ling Jhuo et al. addressed the prediction of influenza and associated pneumonia by considering metrological and pollution parameters, as well as acute upper respiratory infection (AURI) outpatient numbers. They employed a multilayer perceptron to identify the patient numbers of influenza and associated diseases.

Gengfei et al. focused on differentiating types of pneumonia using a Support Vector Machine classifier. Their goal was to assist doctors in accurate diagnosis.

Masahiro et al. aimed to reduce the workload of medical staff by using prediction methods based on electronic medical records.

Luka Racic et al. proposed a CNN methodology using AlexNet to support the decision-making process for pneumonia diagnosis.

Ko et al. addressed accurate lung opacity detection using an ensemble model combining Mask R-CNN and RetinaNet.

Sharma et al. focused on using CNN architectures to improve the speed and accuracy of pneumonia diagnosis using chest X-ray images.

Songtai Wan et al. utilized automatic pneumonia recognition algorithms and depth-wise convolutional neural networks to detect lung tissue necrosis and lung abscess causing pneumonia.

Pratik Patil et al. developed deep CNN models with an image enhancement algorithm to detect pneumonia at an early stage and decrease death rates.

In the proposed model, the limitations and drawbacks of the previous studies have been overcome, as explained in detail in the next section.

IV. PROPOSEDMETHODOLOGY

The proposed cognitive framework is based on structured two-dimensional Pre-trained Convolutional Neural Networks (CNNs) using Transfer Learning. The objective is to enhance the accuracy of pneumonia classification and improve the workflow. The proposed method involves training a Deep Learning VGG16 algorithm to classify pneumonia images. The workflow consists of the following modules:

Image Pre-processing: The input dataset, consisting of pneumonia and normal images, undergoes pre-processing techniques such as resizing, normalization, and augmentation to enhance the quality and diversity of the data.

Feature Extraction: The pre-processed dataset is fed into the pre-trained VGG16 CNN model. The model extracts high-level features from the images, leveraging the knowledge learned from a large dataset of general images.

Classification: The extracted features are then inputted into a classification module. This module utilizes machine learning algorithms, such as support vector machines (SVM), random forest, or logistic regression, to classify the images as either pneumonia or normal.

Model Evaluation: The performance of the trained model is assessed using evaluation metrics such as accuracy, precision, recall, and F1 score. This step helps determine the effectiveness of the proposed framework in accurately classifying pneumonia images.

The proposed cognitive framework aims to improve the accuracy and efficiency of pneumonia classification by utilizing the transfer learning capabilities of pre-trained CNN models. By leveraging the learned features and knowledge from the VGG16 model, the framework enhances the workflow and provides more accurate and reliable results in pneumonia diagnosis.

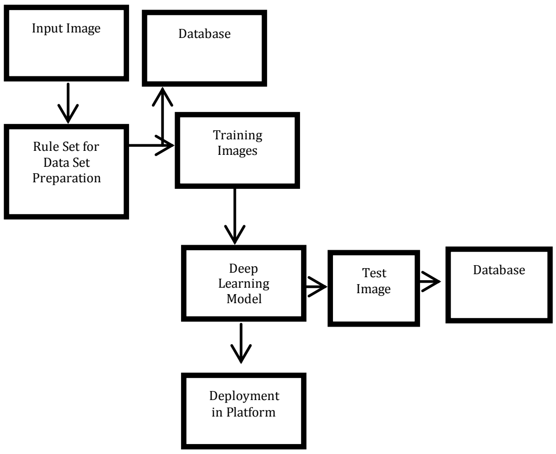


Figure 1 System Architecture

A. Image Pre-Processing

In this proposed work, the Chest X-Ray image datasets from Kaggle are utilized. The dataset consists of a total of 3000 images, which are divided into three subsets: training, validation, and testing. The dataset follows an 80/10/10 split, meaning that 80% of the data is used for training the deep learning models, 10% is used for validation, and the remaining 10% is used for testing.

To prepare the images for the deep learning models, a pre-processing step is performed. The chest X-ray images Shown in Figure 2 are taken as input and resized from the original size of (1024, 1024) pixels to the default size of the VGG16 model, which is (224, 224) pixels. Additionally, the images are transformed into the RGB color space by applying the three primary colors: red, green, and blue. The output of this pre-processing module is a set of resized RGB images ready for further analysis and classification.

By performing these pre-processing steps, the images are standardized and prepared in a consistent format that aligns with the requirements of the VGG16 model. This ensures compatibility and optimal performance during the training and evaluation stages of the proposed framework.

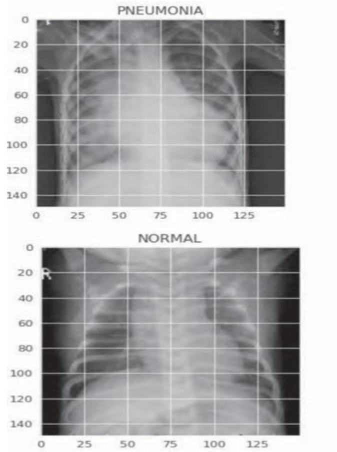


Figure 2 ChestX-ray labeled as:A)pneumonia B)Normal

Pseudo code for the pneumonia detection is given below for your reference.

BEGIN

Initialize the VGG-16 model;

Set weights= ImageNet;

Convert the format of input data;

Set size= (224,224) and shape= size,[3] Input the chest X-ray images; // training data

REPEAT

Calculate the output; // softmax activation function

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Calculate the loss function; // categorical cross entropy

$$CE = \frac{1}{M}\sum_{p}^{M} -log\left(\frac{e^{s_p}}{\sum_{j}^{C} e^{s_j}}\right)$$

Back updates the net weights; //Adam optimizer

UNTIL (reaches the specified epochs);

Loss function with comparison of predicted probability distribution;END

Overall algorithm process is shown in Figure 3

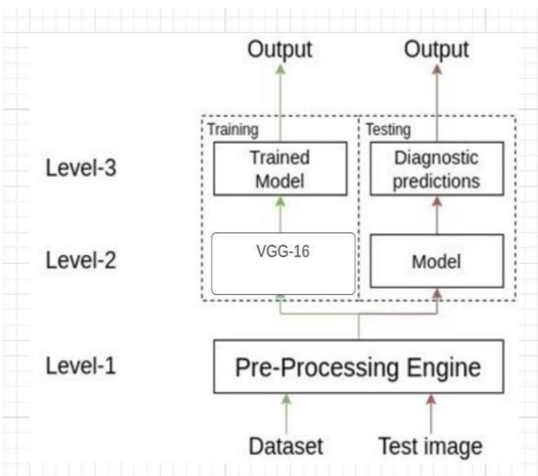


Figure 3 Overall algorithm process

B. Feature Extraction

During the feature extraction stage, the trained model is finalized. CNN models, such as VGG-16, are already equipped with pre-trained feature extraction capabilities, making them suitable for this task. In the proposed framework, statistical results have indicated that the VGG-16 model is the optimal choice for the feature extraction stage.

Based on this finding, the VGG-16 base architecture is employed to build the model. This involves utilizing the pre-trained weights and architecture of VGG-16 as the foundation. By leveraging the learned features from a large dataset of general images, the VGG-16 model can effectively extract relevant features from the input data.

The use of VGG-16 as the base architecture offers several advantages, including its ability to capture intricate details and patterns in images. It has proven to be successful in various computer vision tasks, including image classification. By using VGG-16 as the feature extraction model, the proposed framework aims to leverage its capabilities and optimize the accuracy of pneumonia classification.

By building on the strengths of the VGG-16 architecture and its pre-trained features, the proposed model is expected to enhance the accuracy and efficiency of pneumonia classification, ultimately improving the overall performance of the cognitive framework.

1. Vgg-16 Model Flow Diagram

The VGG16 architecture was proposed by Karen Simonyan and Andrew Zisserman from the University of Oxford. It is a widely used convolutional neural network model known for its excellent performance in image classification tasks. The "16" in VGG16 represents the total number of layers in the architecture, which includes both convolutional and pooling layers.

The VGG16 architecture primarily consists of 3x3 convolutional layers and 2x2 max pooling layers. The use of smaller kernel sizes for convolution helps capture finer details and patterns in the input images. The pooling layers aid in downsampling and reducing the spatial dimensions of the feature maps.

The VGG16 model has a significant number of parameters, approximately 138 million, which are learned during training on the ImageNet dataset. The ImageNet dataset contains millions of labeled images across 1,000 different classes. The model achieves an accuracy of 92.7% on this dataset, demonstrating its effectiveness in image classification tasks.

By leveraging the pre-trained weights and architecture of VGG16, the proposed model in your work benefits from the knowledge and features learned from the vast ImageNet dataset. This enables the model to effectively extract relevant features and improve the accuracy of pneumonia classification in the proposed framework.

2. Feature Extractor

In the VGG16 architecture, the feature extraction process begins by passing the resized RGB image as input to the first convolutional layer. The number of filters in this layer can be assigned up to 512. The purpose of these filters is to detect spatial patterns, such as edges, within the input image.

To identify changes in edge intensity values, the small filters move across the image from top to bottom and from left to right. As they traverse the image, they extract features by detecting and highlighting edges and other spatial patterns.

After the convolutional layer, the output moves on to the max pooling layer. In this layer, a filter covers a specific region of the image, and the maximum pixel value within that region is selected. This process helps reduce the spatial dimensions of the feature maps while retaining the most significant features.

This process of convolution and max pooling is repeated consistently throughout the VGG16 architecture. The result is a series of feature maps that represent a lower-dimensional representation of the original input image, while preserving the most important features.

The output of the feature extraction module, consisting of these reduced-dimensional feature maps, can then be further processed and classified to determine the presence of pneumonia or normalcy in the input image.

3. Proposed Model Architecture

In the proposed diagram Figure 4, the chest X-ray images from the dataset undergo preprocessing steps before being passed into the VGG16 model, as depicted in Figure 5. The purpose of preprocessing is to prepare the images for classification and feature extraction.

After preprocessing, the preprocessed images are inputted into the VGG16 model. The model performs both image classification and feature extraction tasks simultaneously. The VGG16 model consists of multiple layers, including convolutional and pooling layers, which analyze the input images to extract relevant features and classify them into different categories.

Once the image classification and feature extraction processes are completed, the results from the VGG16 model are compiled. This could involve gathering the extracted features, classification predictions, or any other relevant information generated by the model during the processing of the chest X-ray images.

Finally, the compiled model, which incorporates the results from the VGG16 model, is utilized in conjunction with a test chest X-ray image. The purpose is to determine whether the diagnosis of the test image is pneumonia or normal. The model's classification predictions and extracted features are used to make this determination, leveraging the knowledge and patterns learned during the training phase.

.

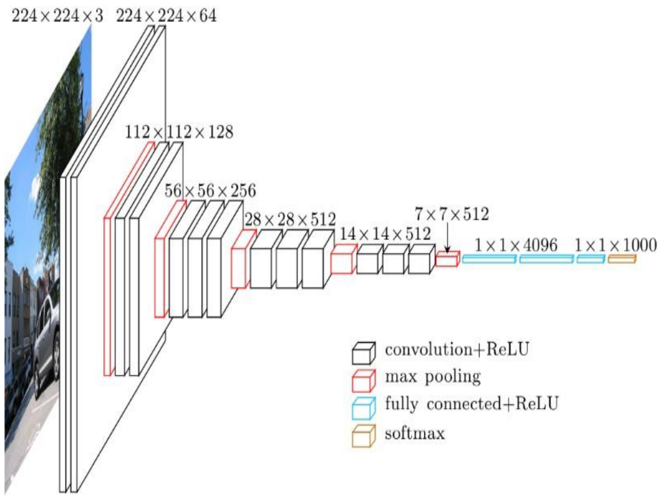


Figure 4 FeatureExtractioninVGG-16Architecture

4. FEATURESELECTION

In the module you described, feature selection is performed by passing the feature map obtained from the VGG16 model into the flatten() function. The purpose of this step is to convert the 2-dimensional array representation of the feature map into a single long continuous vector.

By flattening the feature map, the spatial information is discarded, and the output becomes a one-dimensional representation of the features. This transformation allows for easier handling and processing of the features during subsequent stages.

The flatten() function reduces the dimensionality of the feature map, effectively converting it into a subset of features represented as a single vector. This process helps to retain the important features while discarding the spatial arrangement information.

By performing feature selection through flattening, the model focuses on the relevant and discriminative features extracted from the input chest X-ray images. This condensed feature representation facilitates further analysis and classification tasks, enabling more efficient and accurate predictions of pneumonia or normalcy in the subsequent stages of the framework.

C. IMAGE CLASSIFICATION

After the feature extraction and selection stages, the VGG-16 model's fully connected layer is utilized for image classification in this system. The fully connected layer, also known as the dense layer, is responsible for connecting every neuron from the previous layer to every neuron in the subsequent layer. In this case, the fully connected layer is followed by a Rectified Linear Unit (ReLU) activation function.

The ReLU activation function determines whether a neuron should activate or not by setting all negative values in the matrix to zero, while keeping positive values unchanged. Mathematically, the ReLU activation function can be defined as A(x) = max(0, x), where x is the input value.

The output values from the ReLU activation function are then passed to the output dense layer, which utilizes the softmax activation function. The softmax function outputs values between 0 and 1, and the sum of these values is always equal to 1, resembling a probability distribution. For example, if the softmax assigns a value of 0.7 to class 0 (normal) and 0.3 to class 1 (pneumonia), the output will be class 0 since it has the higher value.

In this system, class 0 represents the normal class, while class 1 represents the pneumonia class. By applying the softmax activation function in the output dense layer, the system generates probabilities for each class, enabling the classification of input images as either normal or pneumonia based on the assigned probabilities.

As a whole, the fully connected layer with ReLU activation and the output dense layer with softmax activation contribute to the final image classification process, providing a decision-making mechanism to distinguish between normal and pneumonia cases based on the assigned probabilities.

MODEL COMPILATION

In this module, the model is compiled by specifying a loss function and an optimizer to minimize the loss during the training process. The loss function is responsible for quantifying the difference between the expected outcome and the output generated by the machine learning model.

In this system, the optimizer used is Adam. Adam (short for Adaptive Moment Estimation) is an optimization algorithm that is commonly used with deep learning models. It helps improve the ability of the convolutional neural network (CNN) in classification tasks by efficiently adjusting the model's parameters during the training process.

The loss function employed in this system is categorical cross-entropy. Categorical cross-entropy is suitable for multi-class classification problems, such as distinguishing between normal and pneumonia cases in this context. It calculates the difference between the predicted class probabilities and the true class labels, aiming to minimize this difference during training.

Additionally, the accuracy metric is defined in this system. Accuracy is a widely used metric for evaluating classification models. It measures the fraction of correct predictions made by the model over the total number of predictions. In the context of this system, accuracy helps assess how well the model performs in correctly classifying chest X-ray images as either normal or pneumonia cases.

By defining the loss function, optimizer, and accuracy metric, the model is prepared for the training process. The optimization algorithm and loss function work together to update the model's parameters, aiming to minimize the loss and improve the model's performance in classification tasks. The accuracy metric provides a measure of the model's success in correctly predicting the class labels of the input images.

Formula for accuracy: TP+TN÷(TP+TN+FP+FN)

Here , TP – true positive, TN – true negative, FP –false positive, FN-false negative.

Model summary of the proposed model is shown below Figure 5.

```
model.summary()

Model: "functional_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 224, 224, 3)]     0
_________________________________________________________________
block1_conv1 (Conv2D)        (None, 224, 224, 64)      1792
_________________________________________________________________
block1_conv2 (Conv2D)        (None, 224, 224, 64)      36928
_________________________________________________________________
block1_pool (MaxPooling2D)   (None, 112, 112, 64)      0
_________________________________________________________________
block2_conv1 (Conv2D)        (None, 112, 112, 128)     73856
_________________________________________________________________
block2_conv2 (Conv2D)        (None, 112, 112, 128)     147584
_________________________________________________________________
block2_pool (MaxPooling2D)   (None, 56, 56, 128)       0
_________________________________________________________________
block3_conv1 (Conv2D)        (None, 56, 56, 256)       295168
_________________________________________________________________
block3_conv2 (Conv2D)        (None, 56, 56, 256)       590080
_________________________________________________________________
block3_conv3 (Conv2D)        (None, 56, 56, 256)       590080
_________________________________________________________________
block3_pool (MaxPooling2D)   (None, 28, 28, 256)       0
_________________________________________________________________
block4_conv1 (Conv2D)        (None, 28, 28, 512)       1180160
_________________________________________________________________
block4_conv2 (Conv2D)        (None, 28, 28, 512)       2359808
```

Figure 5 Model summary

D. PREDICTIONMODEL

In this module, the model is saved using the Save Model function. This functionality allows for saving the model architecture, weights, and traced TensorFlow sub-graphs of the call functions. When the saved model is loaded again, the built-in layers and custom objects are reconstructed, enabling the model to be used for predictions.

The model is saved in the HDF5 format, which is capable of preserving the model architecture and compile() information in a JSON file. This format ensures that all the necessary components of the model are stored for later use.

Once the model is saved, it can be loaded into a Django framework for building a prediction model. The loaded model can be utilized to make predictions using the predict() function. This function takes an input image as an argument and predicts whether the image corresponds to pneumonia or not.

By integrating the saved model into a Django framework and implementing the predict() function, the model can be deployed in a web-based environment. Users can input an image, and the model will use its learned parameters and architecture to make predictions on whether the image represents pneumonia or not.

Overall, this module ensures the persistence and reusability of the trained model by saving it in HDF5 format. The model can then be integrated into a Django framework to create a prediction system, allowing users to obtain predictions on new chest X-ray images.

RESULTs and Discussions

The result evaluation involves analyzing various parameters such as training loss, training accuracy, validation loss, validation accuracy, and module accuracy. These metrics provide insights into the performance and effectiveness of the trained models. Figure 6 displays the results of the training process and accuracy loss.

From the figure, it is evident that the transfer learning model utilizing the VGG-16 algorithm achieves the highest model accuracy of 98.14%, which is approximately 99%. This indicates that the model is capable of accurately classifying pneumonia and normal cases with a high level of confidence. In contrast, the AlexNet model achieves a lower accuracy of 60%, but it has the potential to be further trained to improve its accuracy up to 80%.

The figure also illustrates the loss values associated with the models. The transfer learning model exhibits a low loss value of 0.012, indicating that it is able to minimize the discrepancy between predicted and actual values effectively. On the other hand, the AlexNet model shows a higher loss value of 0.67, indicating that it has a larger discrepancy between predicted and actual values compared to the transfer learning model.

Based on these results, it can be concluded that the transfer learning model with the VGG-16 algorithm is more efficient and accurate in classifying pneumonia and normal cases compared to the AlexNet model. The transfer learning approach allows the model to leverage pre-trained knowledge and features from a large dataset, enabling it to achieve higher accuracy and lower loss values.

These findings demonstrate the superiority of the transfer learning model and highlight its potential for accurate pneumonia classification, which can contribute to improved diagnostic capabilities and patient care.

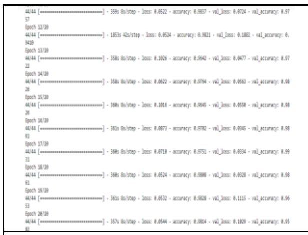
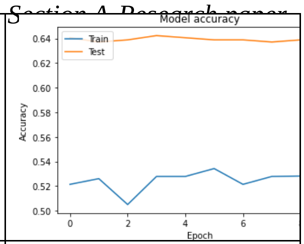
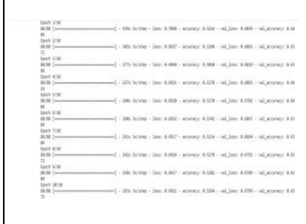
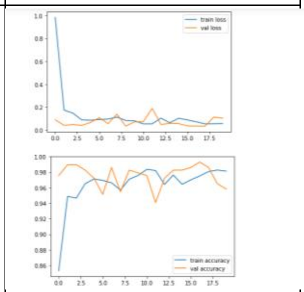
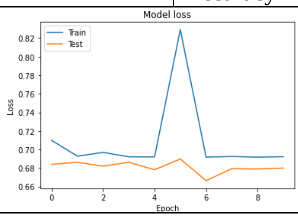
	
a. Transfer Learning Accuracy and Loss	b. Model accuracy
	
c. Alex Net Accuracy and Loss	d. Plotting of Transfer Learning Model and Plotting of Alex Net model Accuracy
	
e. Plotting of Net model Loss	

Figure 6 Results of the proposed model

V. CONCLUSION AND FUTURE WORK

In this paper, a deep learning technique is employed to classify chest X-rays based on the presence or absence of pneumonia characteristics. The proposed model utilizes the VGG-16 model and is implemented using Python programming.

The results of the study indicate a high accuracy of 99%, which suggests that the model is successful in accurately identifying pneumonia in the X-ray images. This demonstrates the effectiveness of the deep learning approach and the ability of the model to learn and extract relevant features from the input data.

However, the paper also acknowledges the need for further training and improvement, especially for special categories. This implies that while the model achieves a good overall accuracy, it may require additional fine-tuning or specific training in order to accurately classify certain subcategories or more challenging cases.

The statement highlights the ongoing nature of deep learning research and the need for continuous refinement and optimization to enhance the model's performance. By focusing on specialized training for specific categories or challenging cases, the accuracy and reliability of the prediction can be further improved.

Overall, the study demonstrates the potential of deep learning techniques, specifically the VGG-16 model, in effectively classifying pneumonia based on chest X-ray images. The high accuracy achieved indicates the model's capability to assist in pneumonia diagnosis. However, ongoing efforts are necessary to address specific challenges and optimize the model's performance for different categories within the scope of pneumonia detection.

REFERENCES

[1] X. Ouyang et al., "Dual-Sampling Attention Network for Diagnosis of COVID-19 From Community Acquired Pneumonia," in IEEE Transactions on Medical Imaging, vol. 39, no. 8, pp. 2595-2605, Aug. 2020, doi: 10.1109/TMI.2020.2995508.

[2] M. E. H. Chowdhury et al., "Can AI Help in Screening Viral and COVID-19 Pneumonia?," in IEEE Access, vol. 8, pp. 132665-132676, 2020, doi: 10.1109/ACCESS.2020.3010287.

[3] S. -L. Jhuo, M. . -T. Hsieh, T. -C. Weng, M. -J. Chen, C. -M. Yang and C. -H. Yeh, "Trend Prediction of Influenza and the Associated Pneumonia in Taiwan Using Machine Learning," 2019 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), 2019, pp. 1-2, doi: 10.1109/ISPACS48206.2019.8986244.

[4] Gengfei Ling1, Congcong Cao2" Automatic Detection andDiagnosis of Severe Viral Pneumonia CT Images Based onLDA-SVM", IEEESensors Journal2020,Vol. 20(20),pp: 11927-11934

[5] M. Hayashitani et al., "Field Trial of Aspiration Pneumonia Predicion based on Electronic Medical Records," 2019 IEEE International Conference on Healthcare Informatics (ICHI), 2019, pp. 1-2, doi: 10.1109/ICHI.2019.8904811.

[6] L. Racic, T. Popović, S. cakicand S. Sandi, "Pneumonia Detection Using Deep Learning Based on Convolutional Neural Network," 2021 25th International Conference on Information Technology(IT),2021,pp. 1-4, doi: 10.1109/IT51528.2021.9390137.

[7] H. Sharma, J. S. Jain, P. Bansal and S. Gupta, "Feature Extraction and Classification of Chest X-Ray Images Using CNN to Detect Pneumonia," 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), 2020, pp. 227-231, doi: 10.1109/Confluence47617.2020.9057809.

[8] Priya, L., Sathya, A., ThangaRevathi, S. (2021). Deep Learning in Healthcare. In: Suresh, A., Paiva, S. (eds) Deep Learning and Edge Computing Solutions for High Performance Computing. EAI/Springer Innovations in Communication and Computing. Springer, Cham. https://doi.org/10.1007/978-3-030-60265-9_8

[9] Patil and H. Patil, "X-ray Imaging Based Pneumonia Classification using Deep Learning and Adaptive Clip Limit based CLAHE Algorithm," 2020 IEEE 4th Conference on Information & Communication Technology (CICT), 2020, pp. 1-4, doi: 10.1109/CICT51604.2020.9312089

[10] Priya, L., Anand, S. Object recognition and 3D reconstruction of occluded objects using binocular stereo. Cluster Comput 21, 29–38 (2018). https://doi.org/10.1007/s10586-017-0891-7

[11] Anand, S., & Priya, L. (2020). A Guide for Machine Vision in Quality Control (1st ed.). Chapman and Hall/CRC. https://doi.org/10.1201/9781003002826

[12] S. Wan, C. -Y. Hsu, J. Li and M. Zhao, "Depth-Wise Convolution with Attention Neural Network (DWA) for Pneumonia Detection," 2020 International Conference on Intelligent Computing, Automation and Systems

(ICICAS), 2020, pp. 136-140, doi: 10.1109/ICICAS51530.2020.00035.

[13] H. Ko, H. Ha, H. Cho, K. Seo and J. Lee, "Pneumonia Detection with Weighted Voting Ensemble of CNN Models," 2019 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD), 2019, pp. 306-310, doi: 10.1109/ICAIBD.2019.8837042.

[14] L.Priya, A.Sathya, K.Poornimathi, J.Anitha "A Novel Intelligent Diagnosis and Disease Prediction Algorithm in Green Cloud Using Machine Learning Approach" Journal of Green Engineering (JGE) Volume-10, Issue-7, July 2020