1. INTRODUCTION

1.1 INTRODUCTION

Pneumonia is a common respiratory infection that can be fatal and affects millions of people globally. Improved detection techniques are essential for successful medical intervention because prompt and precise diagnosis is essential. Conventional techniques frequently depend on medical experts visually evaluating chest X-rays, which can be a laborious and subjective process. Convolutional Neural Networks (CNNs) have become highly effective tools in medical image analysis due to the quick development of artificial intelligence. They have shown remarkable effectiveness in automating the diagnosis of several diseases, including pneumonia.

This Document investigates the use of CNNs in the diagnosis of pneumonia using X-ray images of the chest. CNNs have the potential to improve diagnostic accuracy, decrease manual burden, and speed up decision-making in the healthcare industry by utilizing deep learning. We explore the essential elements of putting a CNN model for pneumonia diagnosis into practice, including training, evaluation, preprocessing, data collection, and model design. The application of CNNs in clinical settings has the potential to enhance patient outcomes and expedite healthcare procedures as researchers and practitioners work to improve these methods.

Everywhere, pneumonia affects both young people and the elderly. Modern computer vision solutions are now readily available to engineers and academics thanks to the rapid rise in popularity of neural networks. The technology of Deep Learning has made it feasible for us to automate analysis processes with the aid of Artificial Intelligence. Many people have a greater risk of contracting pneumonia, particularly in developing and economically poor nations where most people lack access to a nourishing food. According to the World Health Organization, diseases brought on by air pollution account for almost 4 million premature deaths annually.

The goal of this project is to construct an artificial intelligence (AI) network that can conduct linear operations and activations on each pixel value after receiving it as input for a particular X-ray image. Next, by multiplying each of the aforementioned procedures by the number of nodes and each layer of the neural network. You have millions of operations all of a sudden.

They can perform these jobs more effectively if they put in the necessary effort. The goal is to create a Deep Learning model that can determine a patient's status without using chest X-ray pictures to determine whether or not they have pneumonia.

Pneumonia, which is defined as an infection-related lung inflammation, is still a major cause of morbidity and mortality worldwide, particularly in groups that are more susceptible. The conventional diagnosis method entails having skilled radiologists carefully review chest X-rays. But in addition to being labor-intensive, this manual method is also vulnerable to the inherent ambiguity of human perception.

The introduction of artificial intelligence, especially CNNs, has caused a paradigm change in the area of medical imaging in recent years. CNNs are excellent at deriving hierarchical representations from complicated data, which makes them suitable for image recognition and classification applications. CNNs have demonstrated encouraging outcomes in the field of pneumonia diagnosis, providing the possibility to complement and, in certain situations, exceed human diagnostic abilities.

Global health emergencies, like the COVID-19 pandemic, have highlighted the need for reliable and automated pneumonia detection systems even more, since timely diagnosis of respiratory infections is essential for efficient public health care.

The purpose of this paper is to present a thorough analysis of the critical function that CNNs perform in automating the diagnosis of pneumonia from chest X-ray pictures. We hope to contribute to the understanding and application of cutting-edge technologies that have the potential to transform pneumonia diagnosis and, as a result, improve patient outcomes globally by clarifying the critical steps involved in the process, from data acquisition to model evaluation.

HAP increases the likelihood of developing antibiotic resistance, which complicates treatment. Pneumonia claims the lives of about 800,000 children under five each year. More than 2200 persons pass away. Pneumonia affects almost 1400 kids out of every 100,000. According to the Global Burden of Disease Study, pulmonary diseases including pneumonia were the second leading cause of mortality in 2013. Roughly 35 percent of patients in European hospitals and 27.3% of patients globally have been affected by pneumococcal illness. A recent analysis from the John Hopkins Bloomberg College of Public Health states that, among children under the age of five, pneumonia and diarrhea combined killed over 2.97 lac people in India in 2015.

Prior to getting into the specifics of CNN-based strategies, it is important to recognize the methods and techniques already in use for pneumonia identification. In the past, Malaria diagnosing mainly depended on microscopic and doctors manual interpretation of blood cells. This method is effective, but it requires a great degree of expertise and experience and is fundamentally subjective and time-consuming. The machine learning vector machine support requires hand-engineered and requires lot of expertise in the domain knowledge it contains more complex issues and give the false diagnosis in the machine learning malaria diagnosis and the other existing system is microscopic method which gives false diagnosis and the accuracy is very low in these existing systems.

1.2 Existing System

Prior to getting into the specifics of CNN-based strategies, it is important to recognize the methods and techniques already in use for pneumonia identification. In the past, diagnosing pneumonia mainly depended on radiologists' and doctors' manual interpretation of chest X-rays. This method is effective, but it requires a great degree of expertise and experience and is fundamentally subjective and time-consuming.

Disadvantages:

- Need for Large Datasets
- Interpretability
- Data Imbalance
- Computational Complexity
- Overfitting
- Ethical and Bias Concerns
- Resource Intensiveness
- Limited to 2D Images

1.3 Proposed System

This proposed Convolution Neural Network (CNN) model aims to address the problem of image classification, where the goal is to classify image into predefined categories. The primary objectives are to improve classification accuracy, reduce over-fitting, and enhance the model's ability to capture intricate patterns in images.

The suggested system addresses the shortcomings of current methods for detecting pneumonia by incorporating several novel approaches that improve diagnosis efficiency, interpretability, and accuracy. One noteworthy aspect is the incorporation of transfer learning, which uses pre trained CNN models like VGG16 to leverage features extracted from large datasets. This method not only improves the model's performance but also solves the problem of the scarcity of data on pneumonia. The system also uses a multi-modal fusion method, which combines other imaging techniques or clinical data to provide a more comprehensive view for increased accuracy.

Advantages:

- High Accuracy and Sensitivity
- Automatic Feature Extraction
- Transfer Learning for Limited Data
- Efficient Processing of Medical Images
- Automation of Diagnosis
- Scalability
- Integration with Clinical Data
- Speed
- Potential for Early Detection
- Cost-Effective

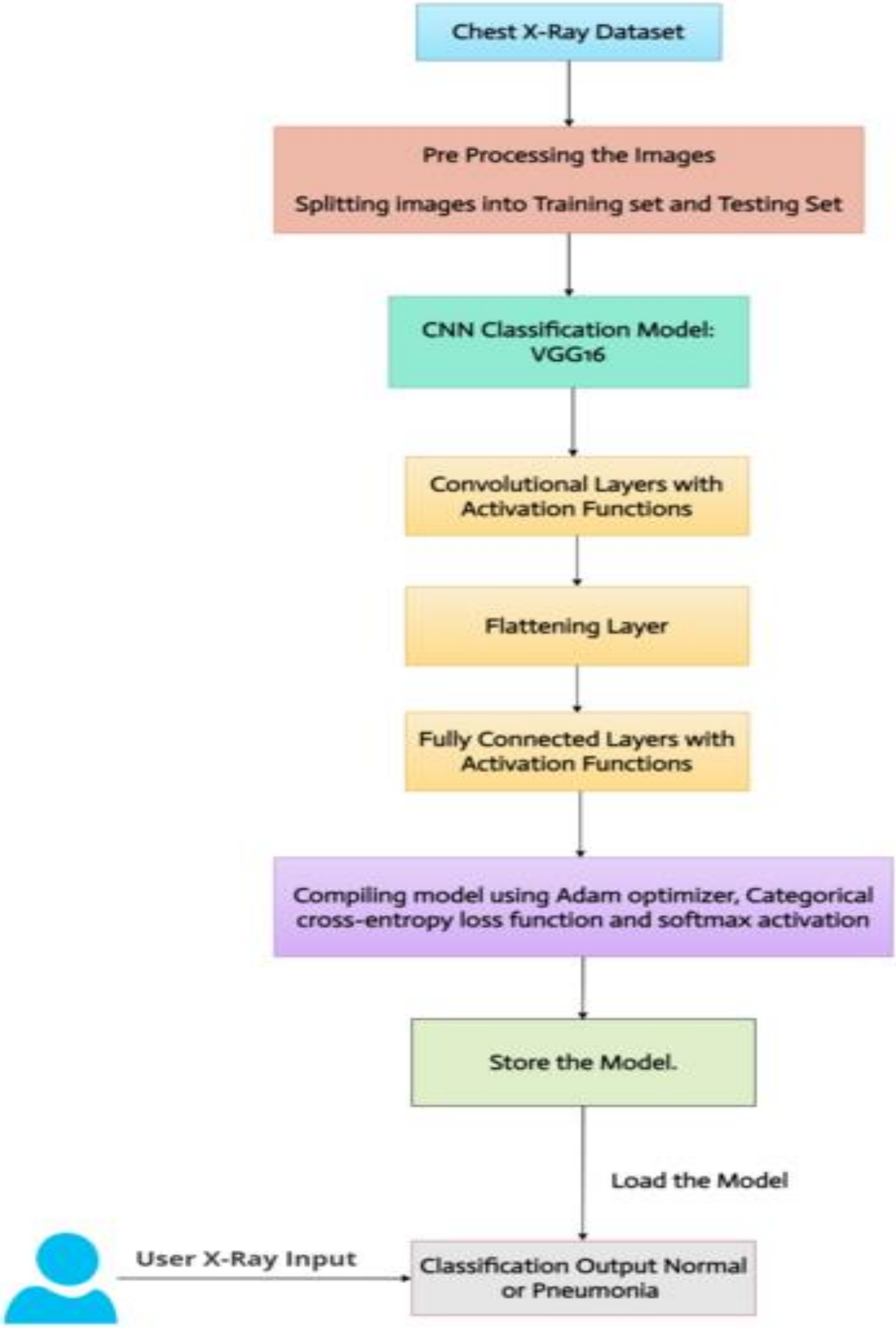

Chest X-Ray Dataset
Pre Processing the Images
Splitting images into Training set and Testing Set
CNN Classification Model:
VGG16
Convolutional Layers with
Activation Functions
Flattening Layer
Fully Connected Layers with
Activation Functions
Compiling model using Adam optimizer, Categorical
cross-entropy loss function and softmax activation
Store the Model.
Load the Model
User X-Ray Input
Classification Output Normal
or Pneumonia

Figure:1.1 proposed System

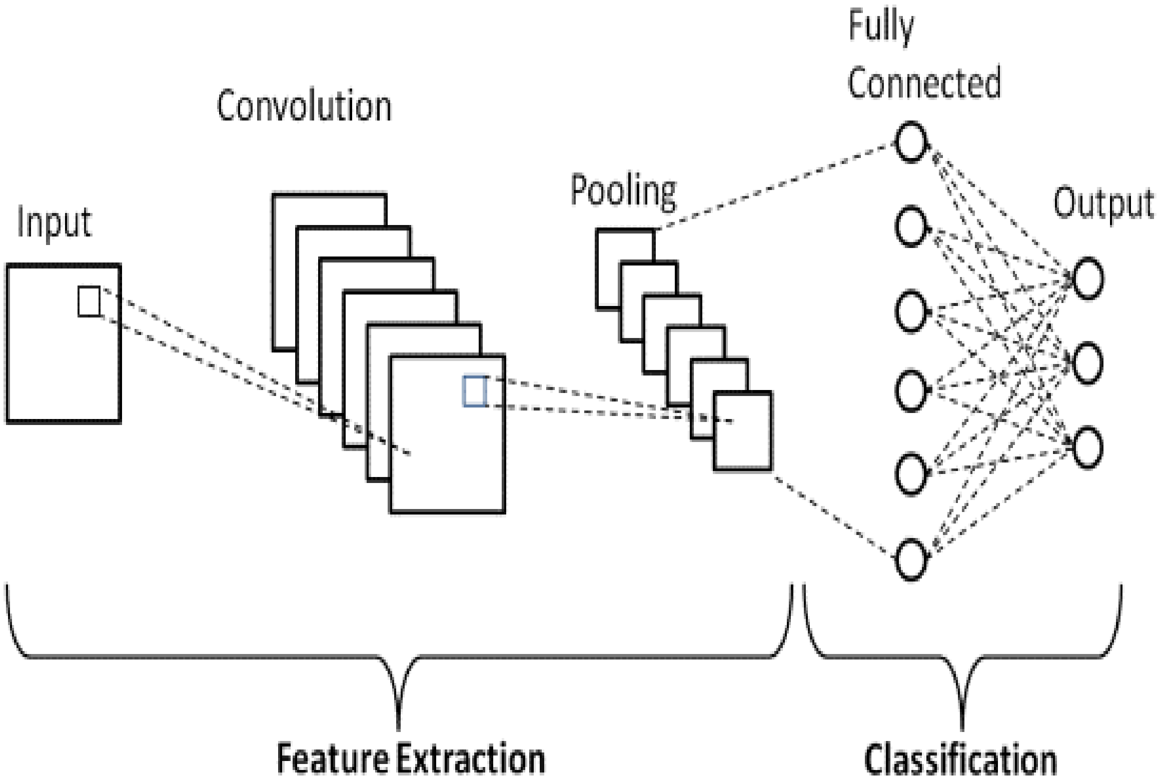


Figure: 1.2 Proposed System

1.4 System Requirements

1.4.1 Hardware Requirements:

• System Type	: Intel Corei5 or above
• Hard – Disk	: 512Gb
• RAM	: 8GB

1.4.2 Software Requirements:

• Operating System	: Windows 10
• Coding Language	: Python
• Python Distribution	: IDLE, Visual Studio Code

2. LITERATURE SURVEY

2.1 Deep Learning

Deep learning is a subfield of machine learning that focuses on the use of artificial neural networks to model and solve complex problems. What distinguishes deep learning from traditional machine learning is the use of deep neural networks, which consist of multiple layers (deep architectures). These layers enable the network to automatically learn hierarchical representations of data, capturing intricate patterns and features at various levels of abstraction.

At the core of deep learning are artificial neural networks, inspired by the structure and function of the human brain. These networks consist of interconnected nodes, or neurons, organized into layers. The input layer receives data, and the output layer produces the desired predictions or classifications. Between the input and output layers are hidden layers, where the network learns to extract features from the input data through iterative training processes. Deep learning has demonstrated remarkable success in a wide range of applications. Convolutional Neural Networks (CNNs) are commonly used for image and video analysis, excelling in tasks like image recognition and object detection. Recurrent Neural Networks (RNNs) are effective for sequential data, making them suitable for natural language processing and time-series prediction. Additionally, architectures like Long Short-Term Memory (LSTM) networks address challenges related to capturing long-term dependencies in sequential data.

Training deep neural networks typically involves a process called backpropagation, where the network adjusts its internal parameters based on the error between predicted and actual outcomes. The availability of large labeled datasets and advances in computational power, especially the use of Graphics Processing Units (GPUs), has facilitated the training of deeper and more complex neural networks. As a result, deep learning has become a dominant approach in various fields, including computer vision, speech recognition, natural language processing, and autonomous systems. Moreover, efforts in model interpretability and explainability aim to make deep learning systems more transparent and understandable for practical applications.

2.2Deep Learning Methods

Deep learning encompasses a variety of methods and architectures designed to model complex patterns and representations in data using deep neural networks. Here are some key deep learning methods:

- **Convolutional Neural Networks (CNNs):**

  Description: CNNs are particularly effective for image and video analysis. They use convolutional layers to automatically learn spatial hierarchies of features in an image. This makes them well-suited for tasks such as image recognition, object detection, and image segmentation.

- **Recurrent Neural Networks (RNNs):**

  Description: RNNs are designed for sequential data and have connections that form directed cycles. They are effective in capturing dependencies over time, making them suitable for tasks like natural language processing (NLP), speech recognition, and time-series analysis.

- **Long Short-Term Memory Networks (LSTMs):**

  Description: LSTMs are a type of RNN designed to address the vanishing gradient problem, which can occur during the training of traditional RNNs. LSTMs use memory cells with gating mechanisms, allowing them to capture long-term dependencies in sequential data. They are commonly used in applications involving sequences, such as language modeling and speech recognition.

- **Autoencoders:**

  Description: Autoencoders are neural networks designed for unsupervised learning and dimensionality reduction. They consist of an encoder and a decoder, with the objective of reconstructing the input data. Autoencoders are used for tasks such as data denoising, feature learning, and anomaly detection.

- **Generative Adversarial Networks (GANs):**

  Description: GANs consist of two neural networks, a generator and a discriminator, which are trained simultaneously through adversarial training. The generator aims to create realistic data, while the discriminator tries to distinguish between real and generated data.

- **Transformers:**

  Description: Transformers have gained prominence in natural language processing tasks. They use self-attention mechanisms to capture contextual relationships between words in a sequence. BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer) are examples of successful transformer-based models.

- **Deep Reinforcement Learning:**

  Description: Deep reinforcement learning combines deep neural networks with reinforcement learning principles. Agents learn to make decisions by interacting with an environment and receiving feedback in the form of rewards. Deep Q Networks (DQN) and Proximal Policy Optimization (PPO) are popular algorithms in this category, used in game playing, robotics, and autonomous systems.

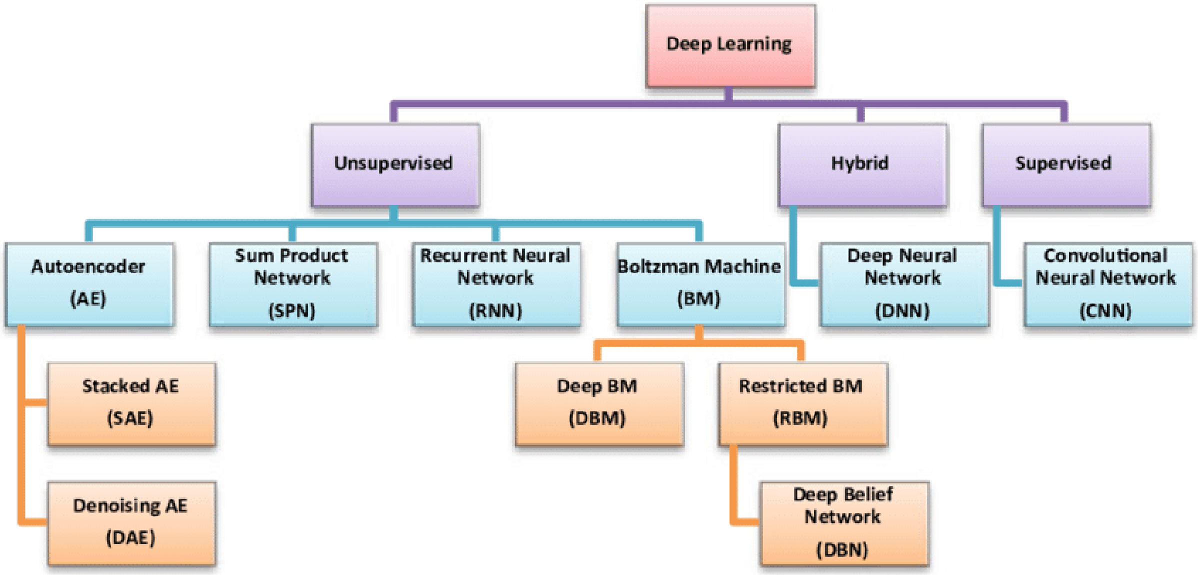


Figure: 2.1 Deep Learning Categories and Algorithms

2.3 DEEP LEARNING APPLICATIONS IN HEALTH CARE

At present, data-driven deep learning (DL) methods have superseded the performance of handcrafted feature extraction mechanisms by self-discovering the attributes from 11 raw pixel data and performing end-to-end feature extraction and classification [18]. In particular, the convolutional neural networks (CNN), a class of DL models, have demonstrated promising results in image classification, recognition, and localization tasks [19]. The promising performance of CNNs is attributed to the availability of huge amounts of annotated data. Under circumstances of limited data availability as in the case of medical images, transfer learning strategies are adopted. Var et al. [20] and [21] proposed methods for computer aided diagnosis based on pre-trained convolutional neural networks as feature extractors to identify malaria parasites. The CNN models are pretrained on large-scale datasetslike ImageNet [22] to transfer the knowledge learned in the form ofgenericimage features to be applied for the target task. The pretrained weights serveas a good initialization and are found to perform better than training the model from scratch with randomly initialized weights. Literature studies reveal the application of conventional ML and data-driven DL methods toward the challenge of malaria parasite detection in thin-blood smear im- ages. Dong et al. [23] compared the performance of kernel-based algorithms such as support vector machine (SVM) and CNN'stoward classifying infected and normal cells.A smallscale collection of segmented red blood cells (RBCs) were randomly split into train/validation/test sets. It was observed that the CNNs achieved a classification accuracy of over 95% and significantly outperformed the SVM classifier that obtained 92% accuracy. The CNNs self-discovered the features from the raw pixel data, thereby requiring minimal human intervention for automated diagnosis. Liang et al. [24] performed cross-validation studies at the cell level to evaluate the performance of custom and pre-trained CNN models toward classifying parasitized and normal cell images. Experimental results demonstrated that the custom CNN outperformed the pretrained AlexNet [19] model with an accuracy of 97.37%. In another study [25], the authors performed randomized splits with peripheral smear images and evaluated the 12 performance of a shallow deep belief network toward detecting the parasites. Experimental results demonstrated that the deep belief network showed promising performance with an F-score of 89.66% as compared to that of SVM based classification that gave an F-score of 78.44%. A CNN model was customized to analyze focal stack of slide images for the presence of parasites [26]. In the process, they observed that the custom CNN model achieved a Matthews Correlation Coefficient (MCC) score of 98.77% and considerably outperformed the SVM classifier that achieved 91.81% MCC. These studies were evaluated at the cell level, with randomized splits and/or small-scale datasets. The reported outcomes are promising; however, patient-level cross-validation studies with large-scale clinical datasets are required to substantiate their robustness and generalization to real-world applications. Rajaraman [21] used a large-scale, annotated clinical image dataset, extracted the features from the optimal layers of pre-trained CNNs and, statistically validated their performance at both cell and patient-level to- ward discriminating parasitized and uninfected cells.

2.3Characteristics of Deep Learning

Supervised, Semi-Supervised or Unsupervised When the category labels are present while you train the data then it is Supervised learning. Algorithms like Linear regression. Logistic regression, decision trees use Supervised Learning. When category labels are not known while you train data then it is unsupervised learning. Algorithms like Cluster Analysis, K means clustering, Anomaly detection uses Unsupervised Learning. The data set consists of both labeled and unlabeled data then we call it is Semi Supervised learning. Graph-based models, Generative models, cluster assumption, continuity assumption use Semi-Supervised learning.

- **Huge Number of Resources**

  It needs advanced Graphical Processing Units for processing heavy workloads. A huge amount of data needs to be processed like big data in the form of structured or unstructured data. Sometimes more time also required to process the data, it depends on the amount of data fed in.
- **Large Number of Layers in Model**

  A huge number of layers like input, activation, the output will be required, sometimes the output of one layer can be input to another layer by making few small findings and then these findings are summed up finally in the SoftMax layer to find out a broader classification for final output.
- **Optimizing Hyper-parameter** Hyperparameters like no of epochs, Batch size, No of layers, Learning rate, needs to be tuned well for successful Model accuracy because it creates a link between layer predictions to final output prediction. Over-fitting and under-fitting can be well handled with hyper-parameters.
- **Cost Function**

  It says how well the model performance in prediction and accuracy. For each iteration in Deep Learning Model, the goal is to minimize the cost when compared to previous iterations. Mean absolute error, Mean Squared Error, Hinge loss, Cross entropy are different types according to different algorithms used.

2.4 Occurrence of Pneumonia

Pneumonia is a common respiratory infection that affects a wide range of people worldwide and poses a serious threat to public health. Pneumonia can be caused by a wide range of infectious agents, such as fungus, viruses, bacteria, and other microbes. The wide range of pneumonia cases is caused by common bacterial infections like Streptococcus pneumoniae and viral agents like influenza. There is a higher danger for vulnerable groups, such as small children, the elderly, and people with weakened immune systems. Susceptibility is increased by elements including cramped living quarters, starvation, and restricted access to healthcare, especially in some areas.

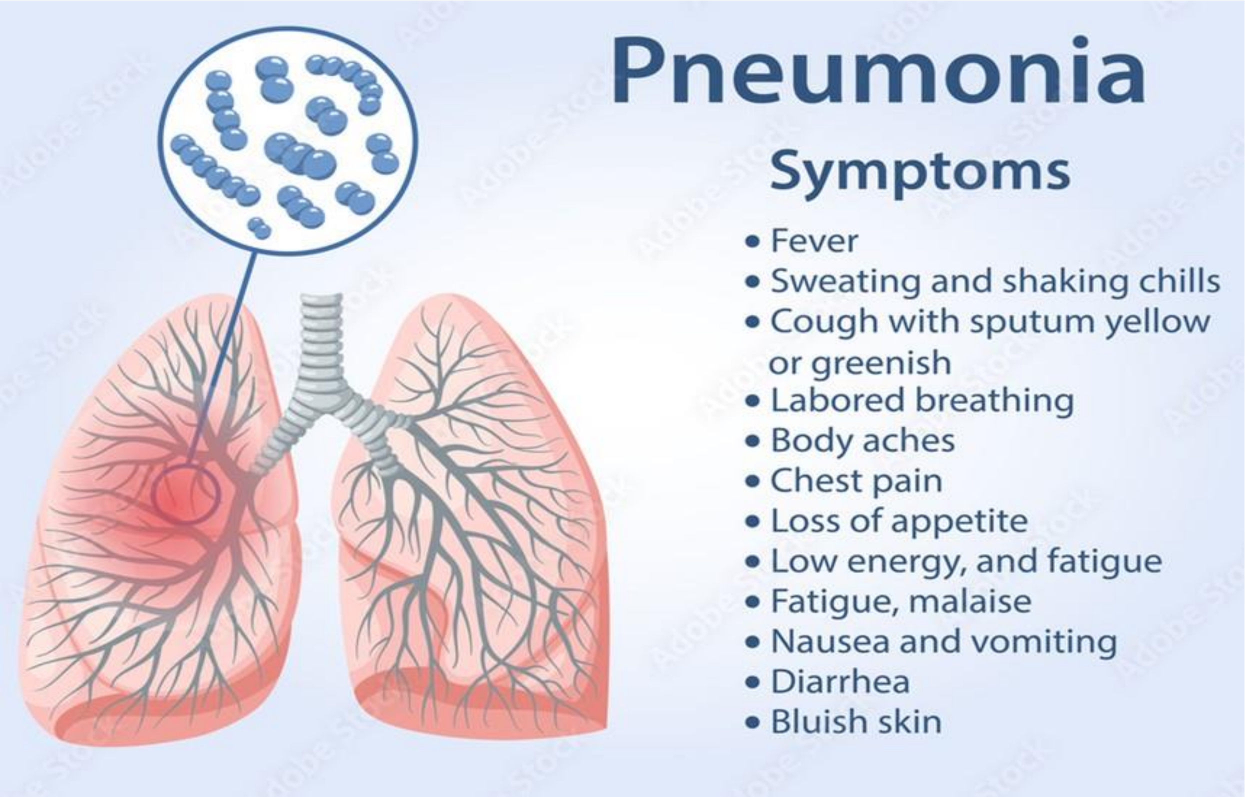
Seasons play a role in the dynamic occurrence of pneumonia; colder months tend to see higher instances. Wintertime is usually the peak season for viral pneumonia, which is frequently linked to influenza and respiratory syncytial virus. The risk of respiratory infections is increased by indoor biomass fuels, tobacco smoke, and air pollution exposure, among other environmental factors. Low temperatures and high humidity are examples of climate factors that can affect the propagation of infectious pathogens.

Depending on where the infection occurred, pneumonia is further divided into community-acquired and hospital-acquired categories. Whereas hospital-acquired pneumonia is linked to inpatient stays or events that happen after discharge, community-acquired pneumonia usually happens outside of medical facilities. Vulnerability is exacerbated by underlying medical issues, such as cardiovascular and chronic respiratory diseases. Important preventive actions include addressing hunger, encouraging hygienic habits, and reducing environmental risk factors.

Pneumonia has a significant influence on public health globally, with lower burdens in low-income nations. One important preventive measure that stands out is vaccination against common infections, which highlights the significance of effective immunization programs. Pneumonia is a common respiratory infection that highlights the need for extensive public health initiatives to lower its prevalence and impact. These efforts should include education, enhanced healthcare infrastructure, and socioeconomic support.

There are some main types of pneumonia.

- Community-Acquired Pneumonia (CAP)
- Hospital-Acquired Pneumonia (HAP)
- Ventilator-Associated Pneumonia (VAP)
- Aspiration Pneumonia
- Viral Pneumonia
- Bacterial Pneumonia
- Mycoplasma Pneumonia
- Fungal Pneumonia
- Atypical Pneumonia
- Childhood Pneumonia
- Walking Pneumonia



Pneumonia

Symptoms

- Fever
- Sweating and shaking chills
- Cough with sputum yellow or greenish
- Labored breathing
- Body aches
- Chest pain
- Loss of appetite
- Low energy, and fatigue
- Fatigue, malaise
- Nausea and vomiting
- Diarrhea
- Bluish skin

2.5 Increasing of Pneumonia Diseases

Pneumonia is becoming more and more common worldwide, which is concerning because of a number of interrelated causes. The rise of bacterial strains resistant to antibiotics poses a serious threat, undermining the effectiveness of traditional therapies and facilitating the persistence and spread of pneumonia. The burden and spread of respiratory diseases are made worse by the world's population expansion, which is especially noticeable in areas with high population density and poor access to healthcare. Older people are more vulnerable to pneumonia because of a higher frequency of comorbidities and compromised immune systems brought on by the aging demographic in many societies. The risk is further increased by underlying medical conditions, such as cardiovascular and chronic respiratory diseases.

Environmental variables affect respiratory health by affecting the prevalence and distribution of infections related to pneumonia, such as air pollution and the effects of climate change. The surge in pneumonia cases is also attributed to gaps in vaccine coverage, immune-compromising factors such as hunger, and overcrowded living circumstances. The complexity is increased by influenza outbreaks, which can result in viral pneumonia. A comprehensive strategy that includes improved healthcare infrastructure, extensive immunization programs, antibiotic stewardship, environmental laws, and public health education is required to address this expanding problem. Effective prevention, early detection, and management of the impact of pneumonia on public health necessitate collaborative efforts at the local, national, and international levels.

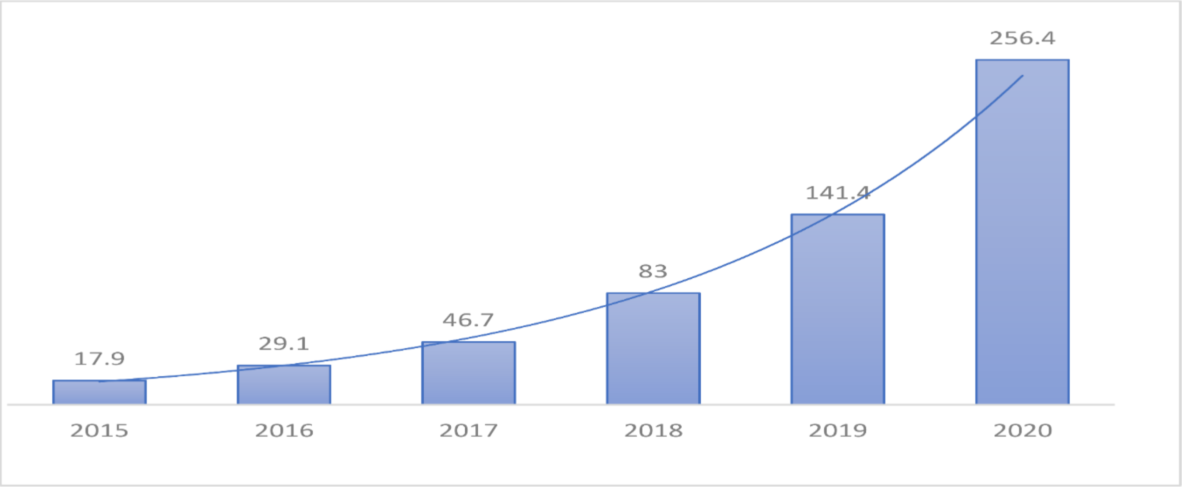


Figure: 2.4 Gradual Growth of Diseases in Pneumonia

3. SYSTEM ANALYSIS

3.1 Scope of the project

The purpose of this study is to compare and contrast the effectiveness of deep learning models, specifically Convolutional Neural Networks, in categorizing human blood smear images as Pneumonia vs. Healthy. This research also focuses on uncovering the effects of preprocessing techniques on model accuracy. Many researches have been conducted on automating the pneumonia detection using deep learning [27], [17] and [10]. Deep Learning has shown great results in the health care industry like the google's Deep Learning algorithm to help identify cancerous tumors on mammograms [28]. Research on diagnosing diabetic retinopathy in retinal images is one of the most successful discovery in the IT and Health sector [29]. Many researchers have conducted their study on the pneumonia Detection and have achieved great results, but with the advancement in the technology and availability of much more data, new field of Artificial Intelligence is gaining popularity that is Deep Learning. Deep learning is a sub field of machine learning that imitates the workings of the human brain in processing data and creating patterns for use in decision making. In this research, we compared the results gained during the research with the preexisting results using machine learning techniques in earlier studies Research on CNNs were not focused on preprocessing techniques, rather they were 15 more relied on getting comparable results using CNNs and transfer learning. Most researches were limited to designing CNN from scratch. We have conducted extensive experiments using the NIH malaria dataset [30] on three different settings, namely, custom network from scratch, fine tuning on pre-trained model and Ensemble mode.

3.2 DATA SET

The dataset for this study comes from one of the many deep learning contests held on Kaggle.1. The collection includes lung medical professionals validated X-ray pictures acquired from the Guangzhou Women's and Children's Medical Center. Every chest X-ray was taken as part of the patient's regular care. medical treatment. Of the 5856 annotated photos in the dataset, 4273 displayed pneumonia, while the remaining 1583 were negatives. A generative adversarial network was utilized to produce additional images for the minority class (which were used primarily during training) because of the imbalance in the dataset. There was no generated image used in the algorithm's evaluation. The single-channel intensity pictures used for all scans ranged in size from 1346 1044 to 2090 1858 pixels. Every image was converted to the 224 224 3 formats in order to meet the requirements of the majority of CNN network topologies. displays a few tests multiple pictures along with their ground truth.

3.3 HARDWARE

A virtual machine with 16 central processing units (CPUs), Google Cloud Platform: Google Cloud Platform is a set of Computing, Networking, Storage, Big Data, Machine Learning and Management services provided by Google that runs on the cloud infrastructure that Google uses internally for its end-user product.

3.4 TECHNIQUES

- Deep Learning
- Keras
- Tensorflow

We used Python script for this project. As a framework, we used Keras 2.3, which is a high-level neural network API written in Python. But Keras can not work by itself; it needs a backend for low-level operations. Thus, we installed a dedicated software library — Google's TensorFlow 2.2. CNN Models: In Deep Learning, Convolutional Neural Network (CNN) is a type of an Artificial Neural Network.CNN or ConvNet is a class of deep, feed-forward artificial neural systems, most normally connected to examining visual representations. CNN is widely used for image recognition, images classifications, objects detections, recognition faces, etc. [33]. In this research, we used two different CNN models: ResNet-50 and, VGG-19.

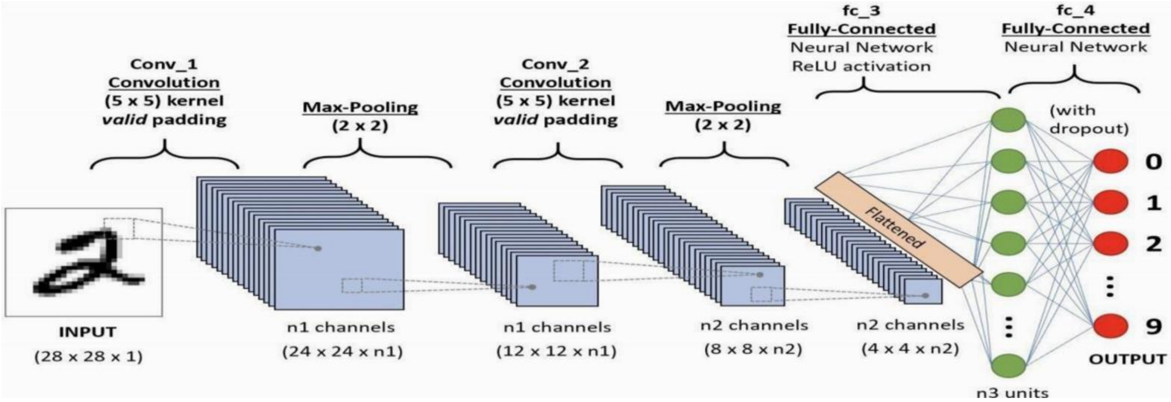


Figure 3.1: CNN Workflow [1]

3.5 PREPROCESSING

In order to extract pertinent information and improve the quality of medical imaging data for precise pneumonia identification using deep learning models, data preprocessing and feature extraction are essential. An outline of the essential procedures for feature extraction and data preprocessing is provided below:

- **Data Cleaning:**

  Remove or handle missing values: Missing data can negatively impact model training. You can either remove instances with missing values or impute them using techniques like mean, median, or interpolation. Outlier detection and handling: Identify and handle outliers to prevent them from disproportionately influencing the model.

- **Data Normalization/Standardization:**

  Scale numerical features to a similar range. Normalization involves scaling the values between 0 and 1, while standardization involves scaling to have a mean of 0 and a standard deviation of 1. This helps the neural network converge faster and can improve its performance.

- **Categorical Encoding:**

  Convert categorical variables into numerical representations. One-hot encoding is a common technique where each category is represented by a binary value in a separate column.

- **Handling Imbalanced Data**

  If your dataset has imbalanced classes, consider techniques such as oversampling the minority class, under sampling the majority class, or using synthetic data generation methods.

- **Feature Engineering:**

  Create new features that might provide additional information to the model. Feature engineering can involve mathematical transformations, interaction terms, or domain-specific knowledge.

- **Handling Time Series Data:**

  If your data involves time series, consider the temporal aspects such as lag features, rolling statistics, and handling seasonality.

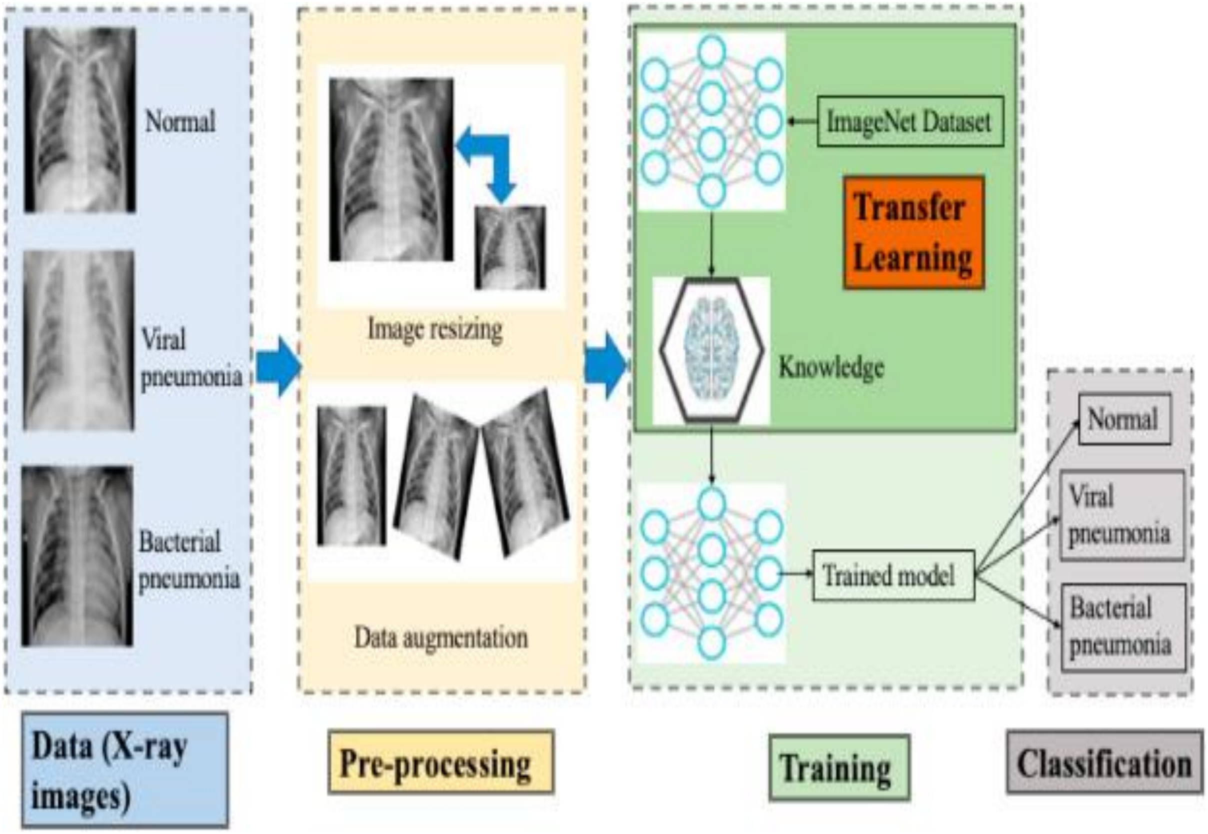


Figure: 3.2 Steps for data preprocessing and feature extraction.

training dataset consists of 63% of the dataset with 8707 healthy images and 8653 pneumonia images, the validation dataset consists of 7% with 1001 healthy and 928 Pneumonia images, and the testing dataset is 30% of the whole dataset having 4144 and 4124 healthy and pneumonia images, respectively. After analyzing the dataset, images in the dataset have different dimensions varying from [46, 58] to [385, 394] over the dataset. Scaling images up gives an advantage in performance but also requires a lot of computation time and memory space. It is betterto maintain this trade-off of accuracy and computation. In this research, we decidedto resize all the images to [125,125]. This resulted in a better classification score withgood processing speed. Since neural networks receive inputs of the same size, all images need to be resized to a fixed size before inputting them to the CNN [23]. The larger the fixed size, 22 the less shrinking required. Less shrinking means less deformation of features and patterns inside the image. This will mitigate the classification accuracy degradation due to deformations. However, large images not only occupy more space in the memory but also result in a larger neural network. Thus, increasing both the space and time complexity. It is obvious now that choosing this fixed size for images is a matter of tradeoff between computational efficiency and accuracy.

3.6 LABLE ENCODING

Label Encoding is a technique of converting categorical variables to a numeric value. In figure 5.2 each value in the column is converted to a number. In this research, we used label encoding to encode the target dataset. The labels Pneumonia and healthy are encoded as 1 and 0, respectively. The label encoder is implemented only on the training and validation datasets.

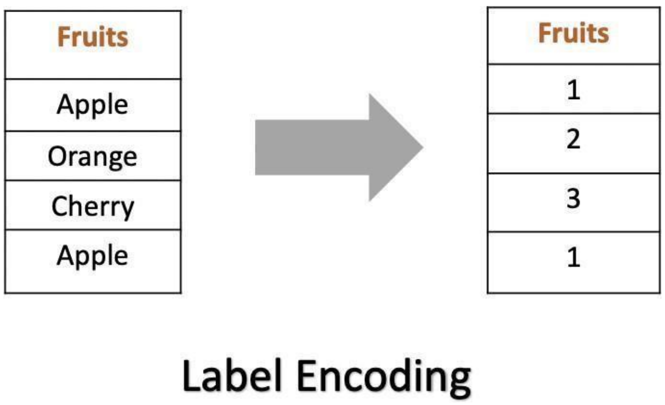


Figure 3.3 : Label Encoding

3.7 NORMALIZATION

Normalization is an important preprocessing task which reduces the color and intensity variations present in stained images from different laboratories. According to past 23 research, stain normalization has proven to significantly increase the accuracy of the unseen dataset by approximately eight percent [34]. In this research, the images are collected from the preexisting dataset of human blood cells which is prepared from in laboratory examination. The smear slides are prepared in the laboratory using various chemical stains which results in color variation due to the use of different chemicals and staining procedures. This staining results in the model to learn and deal with more complex models with a diverse set of images leads to maximizing error rate. A solution to standardize this is normalization. Stain Normalization is a common preprocessing technique that attempts to reduce color variability and improve the generalization of algorithms by transforming the input data to a common space. In stain normalized digital pathology samples, regions of digital tissue specimens are mapped to similar color characteristics regardless of the scanning device, stain vendor, and preparation protocols.

Because of the reduced variability in color characteristics of tissues, Stain Normalization has demonstrated improvement in computer-assisted diagnostic tools [35, 36, 37]. In their article, Cimopi et al. [31] have shown how methods like histopathology [38] and stain normalization can improve the classification of colorectal tissues in colorectal cancer [31]. In this research, we have implemented stain normalization on the training and validation dataset while leaving the test dataset untouched.

3.8 MODEL ARCHITECTURE

A deep learning project's success depends on creating a strong model architecture, particularly when it comes to pneumonia detection. The neural network's design controls how it interprets input data and generates predictions. An overview of the essential components for creating a model architecture for pneumonia detection is provided below**:**

Input Layer:

- Specify the input layer to accommodate the size and format of medical images (e.g., chest X-rays).
- Consider appropriate normalization techniques for pixel values.

Convolutional Layers (CNN):

- Utilize convolutional layers for automatic feature extraction from images.
- Stack multiple convolutional layers to capture hierarchical features.
- Experiment with different filter sizes and strides.

Activation Functions:

- Apply activation functions (e.g., ReLU) after convolutional layers to introduce non-linearity.
- Enhance the network's ability to learn complex patterns.

Pooling Layers:

- Integrate pooling layers (e.g., MaxPooling2D) to downsample feature maps.
- Reduce spatial dimensions while retaining essential information.

Batch Normalization:

- Include batch normalization layers to improve convergence and stability during training.
- Normalize activations within each mini-batch.

Dropout:

- Add dropout layers to prevent overfitting.
- Randomly drop a fraction of connections during training.

Flatten Layer:

- Flatten the output from convolutional layers into a one-dimensional vector.
- Prepare the data for fully connected layers.

Fully Connected (Dense) Layers:

- Design fully connected layers for learning global patterns.
- Experiment with different activation functions in these layers.

Output Layer:

- Configure the output layer with a single neuron for binary classification (pneumonia-positive or pneumonia-negative).
- Use the sigmoid activation function for binary classification**.**

3.9 TRANSFER LEARNING

Transfer Learning is a feature that enables users to transfer the knowledge of pre-trained models and use it in their own problem set. Instead of creating a model from scratch in this research, we used the models that are trained on large datasets such as ImageNet with 100,000 datapoints and explored the power of transfer learning which is proven to be significant in many image classification types of research [42].

In this research, we recognized that CNNs trained on large-scale datasets could serve as feature extractors

3.8 Residual Networks

With the express purpose of addressing the difficulties associated with training very deep networks, Residual Networks (ResNets) constitute a significant advancement in deep neural network topologies. ResNets changed the field when they were introduced in 2016 because they included residual blocks, which each had a shortcut link that omitted one or more layers. Because of this breakthrough, ResNets can train networks with hundreds or thousands of layers, which helps to mitigate the issue of disappearing gradients that arises with standard deep networks. The network's learning duty is made simpler by the skip connections, which allow for direct information flow and aid in the preservation of gradients during backpropagation. Global average pooling and batch normalization are two typical techniques used by ResNets, which enhance their parameter efficiency and deep model training efficacy.

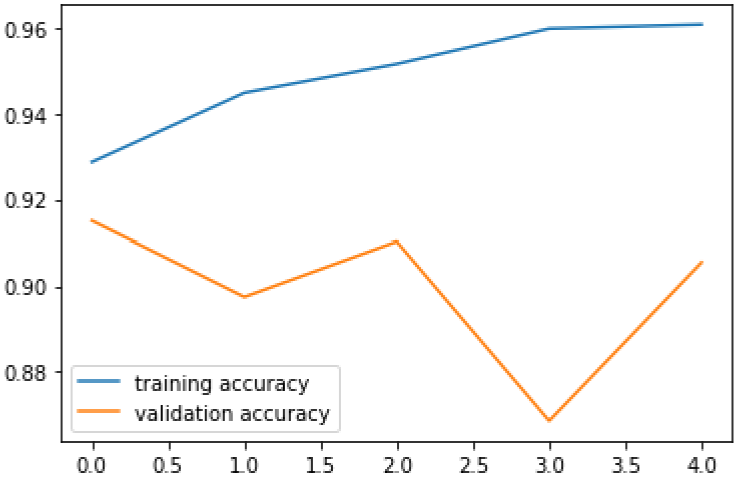


Figure 3.4 Training and Validation Accuracy

Residual Networks (ResNets) are becoming widely used in deep learning research and applications due to their unique advantages that go beyond their innovative architectural design. ResNets have several advantages over other architectures with comparable capacity, one of which is their parameter efficiency. ResNets may frequently achieve competitive performance with fewer parameters. Because of its efficiency, ResNets can be used in contexts with limited resources because they require less compute during training and inference.

Furthermore, even in situations when the ideal transformation is near to the identity mapping, the usage of residual blocks in ResNets enables efficient training. Because of this trait, learning is made easier, allowing the network to concentrate on improving the residual mapping. Remaining blocks' identity mapping ability guarantees that the network can decide to forego pointless transformations, thus increasing efficiency.

3.9 Residual Block

In order to solve the problem of the vanishing/exploding gradient, this architecture introduced the concept called Residual Network. In this network we use a technique called skip connections. The skip connection skips training from a few layers and connects directly to the output. The approach behind this network is instead of layers learn the underlying mapping, we allow network fit the residual mapping. So, instead of say H(x), initial mapping, let the network fit, F(x): = H(x) – x which gives H(x) : = F(x) + x.

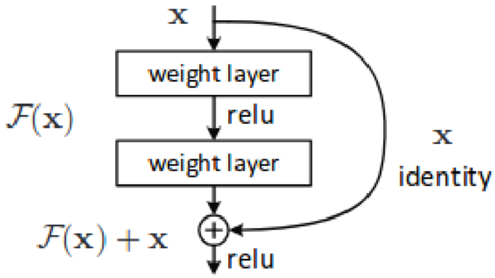

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Figure: 3.5 Residual Block with ReLu

3.10 PERFORMANCE PARAMETER

Model performance is evaluated on four different parameters: accuracy, precision, recall, and F1-score.

Accuracy = *TP* + *TN* / TP + TN + FP + FN

Precision = TP / TP + FP

Recall = TP / TP + FN

F1-score = 2 × PRECISION × RECALL / PRECISION + RECALL

3.11 Network Architecture

VGG-16 Model

VGG16 is a convolutional neural network architecture from the Visual Geometry Group that is notable for its ease of use and efficiency in picture classification. VGG16, which was first presented in 2014 as part of the ImageNet Large Scale Visual Recognition Challenge, is a 16-layer architecture that consists of 13 convolutional layers and 3 fully linked layers. Its architecture, which consists of repeating blocks of tiny 3x3 filters and max-pooling layers, is what makes it unique. This allows the model to gradually capture complex hierarchical properties. Nonlinearity is introduced via ReLU activation functions, and a softmax layer for classification is the result of the last layers. Using pre-trained weights on big datasets like

ImageNet, VGG16 is well suited for transfer learning, which is one of its noteworthy advantages

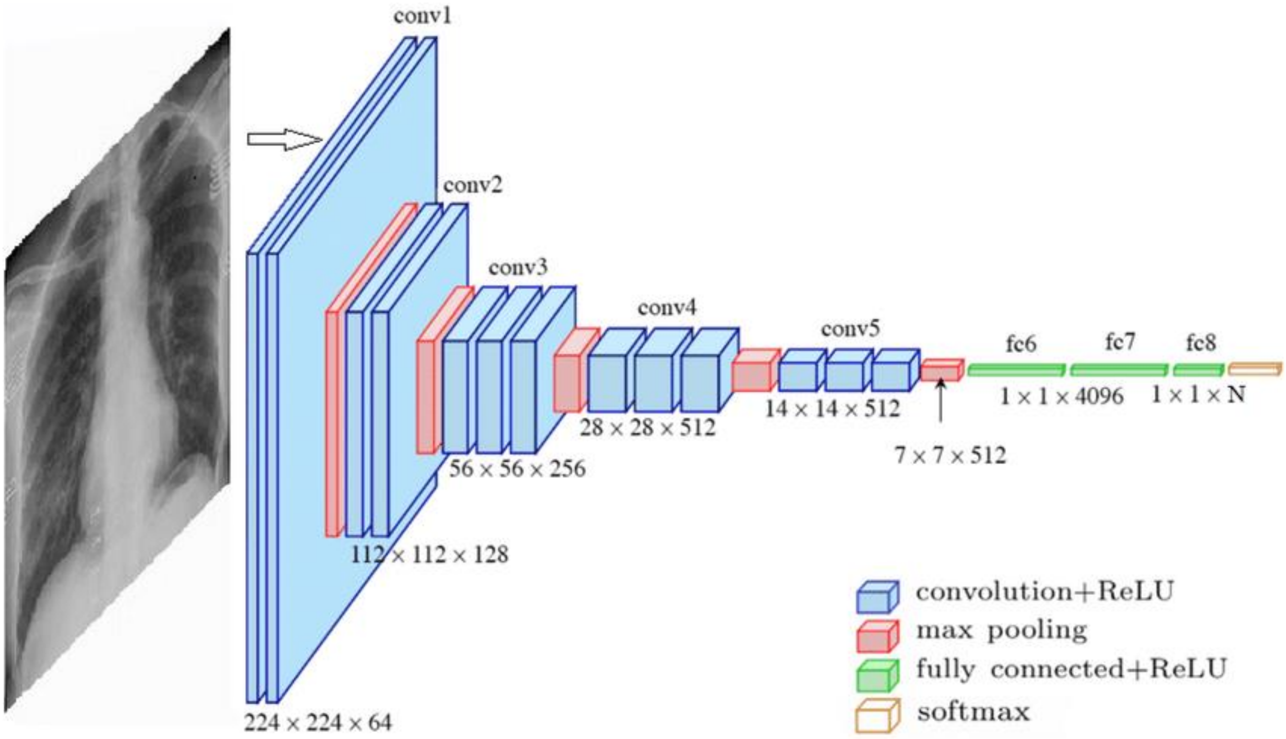


Figure: 3.6 VGG-16 Architecture

VGG16, a convolutional neural network model proposed by K. Simonyan and A. Zisserman from the University of Oxford in their paper "Very Deep Convolutional Networks for Large-Scale Image Recognition," has garnered recognition for achieving a top-5 test accuracy of 92.7% on the ImageNet dataset. This dataset comprises over 14 million images distributed among 1000 classes and served as a benchmark for models submitted to ILSVRC-2014. VGG16's notable improvement over AlexNet involves the substitution of large kernel-sized filters (11 and 5 in the first and second convolutional layers, respectively) with multiple consecutive 3×3 kernel-sized filters. The training of VGG16 spanned several weeks, utilizing NVIDIA Titan Black GPUs for computational support.

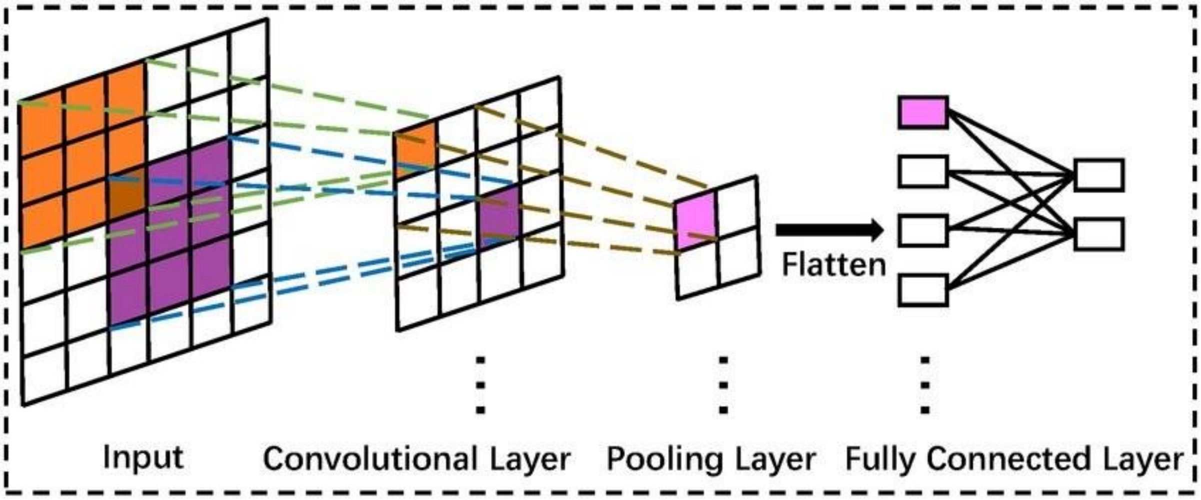


Figure: 3.7 Mathematics of Conv and Pooling Layer

A false positive (FP) occurs when the model incorrectly predicts the positive class, while a false negative (FN) refers to an outcome where the model incorrectly predicts the negative class.

- **Sensitivity or recall or hit rate or true positive rate (TPR)**

  It is the proportion of individuals who actually have the disease were identified as having the disease.

  - TPR= tp / (tp + fn)

- **Precision or positive predictive value (PPV)**

  If the test result is positive what is the probability that the patient actually has the disease.

  - PPV= tp / (tp + fp)

- **Negative predictive value (NPV)**

  If the test result is negative what is the probability that the patient does not have disease.

  - NPV= tn/ (tn+fn)

- **Fall-out or false positive rate (FPR)**

  It is the proportion of all the people who do not have the disease who will be identified as having the disease.

  - FPR=fp/ (fp+tn)

- Accuracy

  The accuracy reflects the total proportion of individuals that are correctly classified.

  - ACC=(tp+tn) / (tp+tn+fp+fn)

4. DESIGN

4.1 The design of the system

There are 12 distinct steps in the proposed design, representing the layers and phases of a Convolutional Neural Network (CNN) for image processing.

the above image

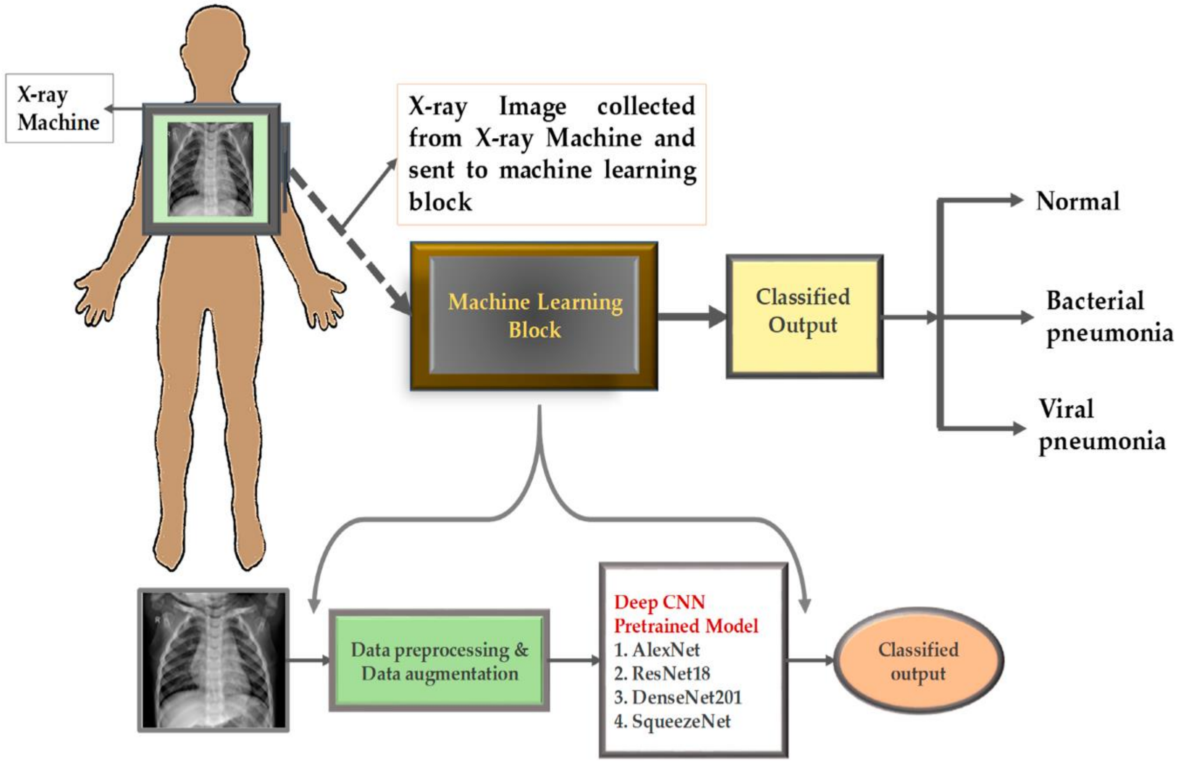


Figure: 4.1 Design

1. Input Data

- Collect a dataset of labeled images for training and testing.
- Preprocess the images to ensure uniform dimensions (e.g., resize to 224x224 pixels) and adjust the number of channels (commonly 3 for RGB).

2. Data Augmentation

Apply data augmentation techniques to artificially increase the diversity of the training dataset. This may include random rotations, flips, zooms, and shifts.

3. Normalization

- ➢ Normalize pixel values to a standard range (e.g., [0, 1] or [-1, 1]).
- ➢ Normalize based on the mean and standard deviation of the ImageNet dataset if using pre-trained weights.

4. Model Selection

Choose the VGG16 architecture as the base model due to its effectiveness in image classification tasks.

5. Transfer Learning

- ➢ Load pre-trained weights from ImageNet to benefit from learned features.
- ➢ Decide whether to freeze all pre-trained layers or fine-tune specific layers based on the size of your dataset.

6. Customization

- ➢ Add custom fully connected layers on top of the VGG16 base to match the number of classes in your specific classification task.
- ➢ Consider using dropout layers for regularization to prevent overfitting.

7. Model Compilation

Compile the model using an appropriate optimizer (e.g., Adam), categorical crossentropy loss (for multi-class classification), and accuracy as the evaluation metric.

8. Training

- ➢ Split the dataset into training and validation sets.
- ➢ Train the model on the training set, validating on the validation set.
- ➢ Monitor training metrics, and consider early stopping to prevent overfitting.

9. Evaluation

- ➢ Assess the model's performance on a separate test set using relevant metrics.
- ➢ Analyze confusion matrices, precision-recall curves, and ROC curves for a comprehensive evaluation.

10. Deployment

- ➢ Integrate the trained model into your application or system for real-world use.

- Ensure compatibility with the target platform and optimize the model for inference.

12. Monitoring and Maintenance

- Establish mechanisms for continuous monitoring of model performance.
- Periodically retrain the model with new data to adapt to evolving patterns.

13. Output

The entire design process, including dataset details, model architecture, hyperparameters, and training outcomes.

5. MPLEMEMTATION

5.1 CODE FOR EXTRACTING THE DATA

The python source used in the study is as presented in this section. This source code was used to extract data from the Ethereum network. Various pythoin libraries were used. They include NumPy, SciPy, Pandas, Matplotlib, network, pickle, SYS and OS.

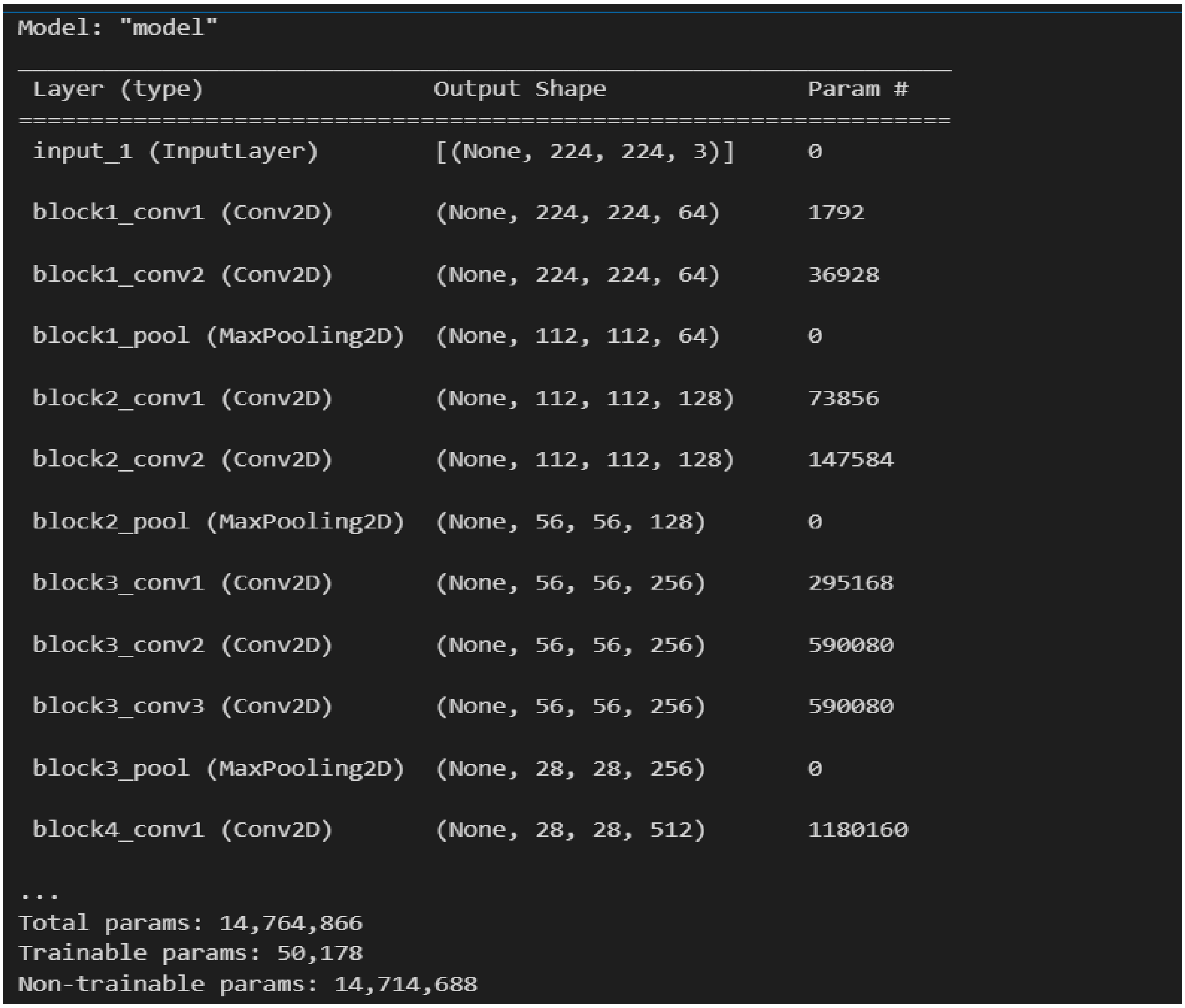
```
from keras.models import Model
from keras.layers import Flatten,Dense
from keras.applications.vgg16 import VGG16
import matplotlib.pyplot as plot
from glob import glob
IMAGESHAPE = [224, 224, 3
vgg_model = VGG16(input_shape=IMAGESHAPE, weights='imagenet', include_top=False)
training_data = 'chest_xray/train'
testing_data = 'chest_xray/test' #Give our training and testing path

for each_layer in vgg_model.layers:
    each_layer.trainable = False #Set the trainable as False, So that all the layers would not be trained.
classes = glob('chest_xray/train/*') #Finding how many classes present in our train dataset.
flatten_layer = Flatten()(vgg_model.output)
prediction = Dense(len(classes), activation='softmax')(flatten_layer)
```

```
final_model = Model(inputs=vgg_model.input, outputs=prediction) #Combine the VGG
output and prediction , this all together will create a model.
final_model.summary() #Displaying the summary
final_model.compile( #Compiling our model using adam optimizer and optimization metric
as accuracy.
loss='categorical_crossentropy',
optimizer='adam',
metrics=['accuracy']
)
from keras.preprocessing.image import ImageDataGenerator
train_datagen = ImageDataGenerator(rescale = 1./255, ImageDataGenerator in keras.
shear_range = 0.2,
zoom_range = 0.2,
horizontal_flip = True)
testing_datagen = ImageDataGenerator(rescale =1. / 255)
training_set = train_datagen.flow_from_directory('chest_xray/train', #inserting the images.
target_size = (224, 224),
batch_size = 4,
class_mode = 'categorical')
test_set = testing_datagen.flow_from_directory('chest_xray/test',
target_size = (224, 224),
batch_size = 4,
class_mode = 'categorical')
fitted_model = final_model.fit_generator( #Fitting the model.
training_set,
validation_data=test_set,
epochs=5,
steps_per_epoch=len(training_set),
validation_steps=len(test_set)
)
plot.plot(fitted_model.history['loss'], label='training loss') #Plotting the accuracies
plot.plot(fitted_model.history['val_loss'], label='validation loss')
plot.legend()
plot.show()
plot.savefig('LossVal_loss')
plot.plot(fitted_model.history['accuracy'], label='training accuracy')
plot.plot(fitted_model.history['val_accuracy'], label='validation accuracy')
plot.legend()
plot.show()
plot.savefig('AccVal_acc')
final_model.save('our_model.h5') #Saving the model file.
final_model.save('our_model.h5')
from keras_preprocessing import image
from keras.models import load_model
from keras.applications.vgg16 import preprocess_input
import numpy as np
model=load_model('our_model.h5') #Loading our model
img=image.load_img(r'E:\FINAL
PROJECT\Achieve\archive\chest_xray\chest_xray\val\NORMAL\NORMAL2-IM-1436-
0001.jpeg', target_size=(224,224))
imagee=image.img_to_array(img) #Converting the X-Ray into pixels
imagee=np.expand_dims(imagee, axis=0)
img_data=preprocess_input(imagee)
prediction=model.predict(img_data)
if prediction[0][0]>prediction[0][1]: #Printing the prediction of model.
    print('Person is safe.')
else:
    print('Person is affected with Pneumonia.')
print(f'Predictions: {prediction}')
```

6. RESULT ANALYSIS

RESULT

The customized model i.e a combination of CNN based feature-extraction and supervised classifier algorithm resulted in optimal solution for classifying abnormal (Pneumonia labeled) and normal Chest X-Ray images primarily due to the substantive features provided by DenseNets followed by optimal hyper-parameter values of SVM classifier. Literature studies reveal the contribution of transfer learning methods including feature-extractions toward visual recognition tasks. For this reason, we extracted features from various variants of pre-trained CNN models available such as VGGNets, Xception , ResNet-50 and DenseNets. Studies from the literature also reveal the use of classifiers in combination with CNN-based feature extraction majorly in medical image analysis to meliorate the performance of models. Following the mentioned past approaches, we evaluated each of the pre-trained models with distinct classifiers to determine the ideal model for the purpose. We observed from the comparative experimental results presented in Table 1 and Table 2 that ResNet50 outperformed the results of all the other pre-trained CNN models when employed with default parameter values of SVM classifier. In addition, DenseNets were also observed to achieve results close to ResNet50. Literature studies reveal that DenseNets outperformed all the pre-trained CNNs in the ImageNet dataset. For this reason, we chose ResNet50, DenseNet-121 and DenseNet169 as the optimal CNN models for the feature-extraction stage and SVM as the optimal classifier for the classification stage for further experiments in the study. The selection of SVM classifier with rbf kernel based on the statistical results presented in Figure 3 further led to hunt of optimal hyperparameter values (C and gamma). In the process of tuning hyper-parameters, we performed close to 350 combinations of C and gamma, the crucial combinations among these are presented in Table 3,4 and 5. We observed in this process that DenseNet-169 outperformed all the other customized models and hence chosen as the best feature-extractor for the final customized model followed by optimal hyper-parameter values of SVM rbf kernel. The best results achieved with DenseNet169 architecture as feature extractors can be explained due to its capability of accessing feature-maps from all of its preceding layers. Literature studies of DenseNets mentions the information flow from the beginning layer to the end layers and removal of redundant features by transition layers as the primary reasons for the high-features representations. To our knowledge, no literature was found to perform the studies on the combination of CNN based feature extractions and supervised classifier algorithms for the underlying task. In regard, we have proposed a model architecture for detecting Pneumonia from frontal chest X-ray images with the utilization of Densenet as feature-extractors and SVM as the process of meliorating the model performance, we found that our customized model outperforms the results documented in the recently released work of Benjamin Antin et al. [18] for the same problem of pneumonia detection.



```
Model: "model"
_________________________________________________________________
 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

...
Total params: 14,764,866
Trainable params: 50,178
Non-trainable params: 14,714,688
```

Figure 5.1 basic CNN model

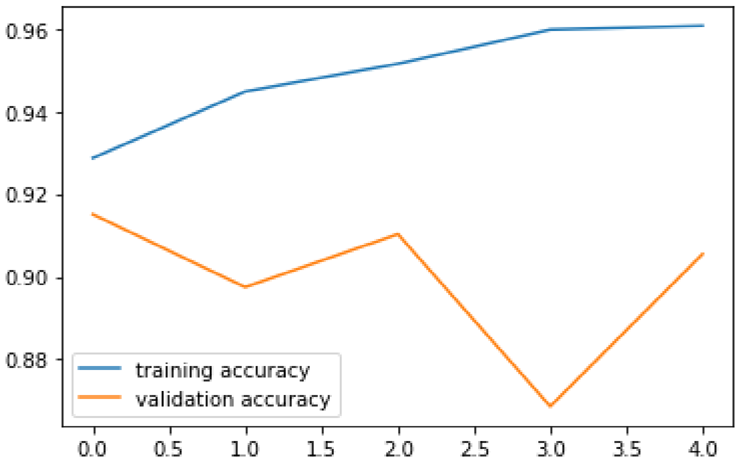
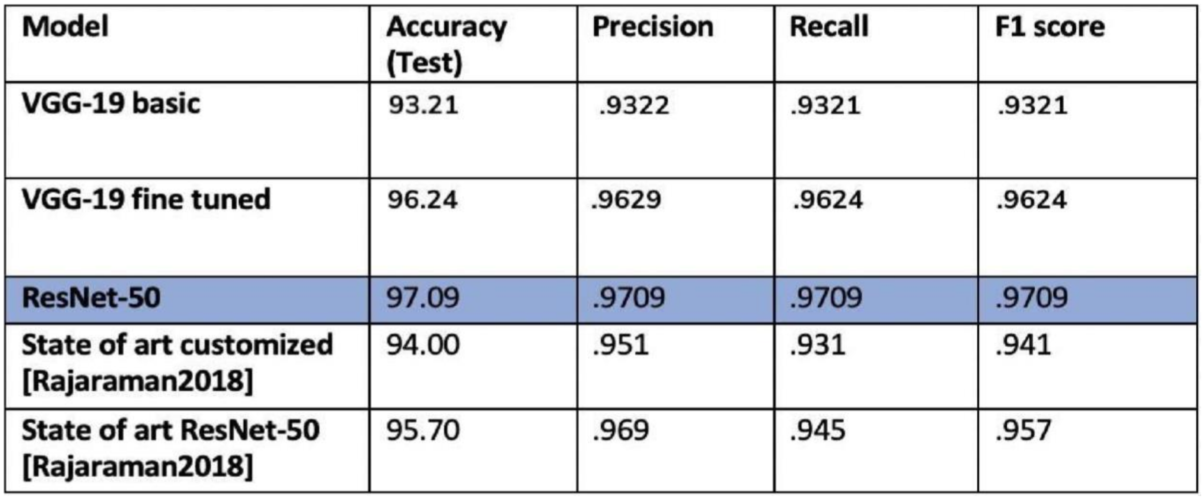


Figure 5.2: Training and Validation



Model	Accuracy (Test)	Precision	Recall	F1 score
VGG-19 basic	93.21	.9322	.9321	.9321
VGG-19 fine tuned	96.24	.9629	.9624	.9624
ResNet-50	97.09	.9709	.9709	.9709
State of art customized [Rajaraman2018]	94.00	.951	.931	.941
State of art ResNet-50 [Rajaraman2018]	95.70	.969	.945	.957

Figure 5.3: Model performance

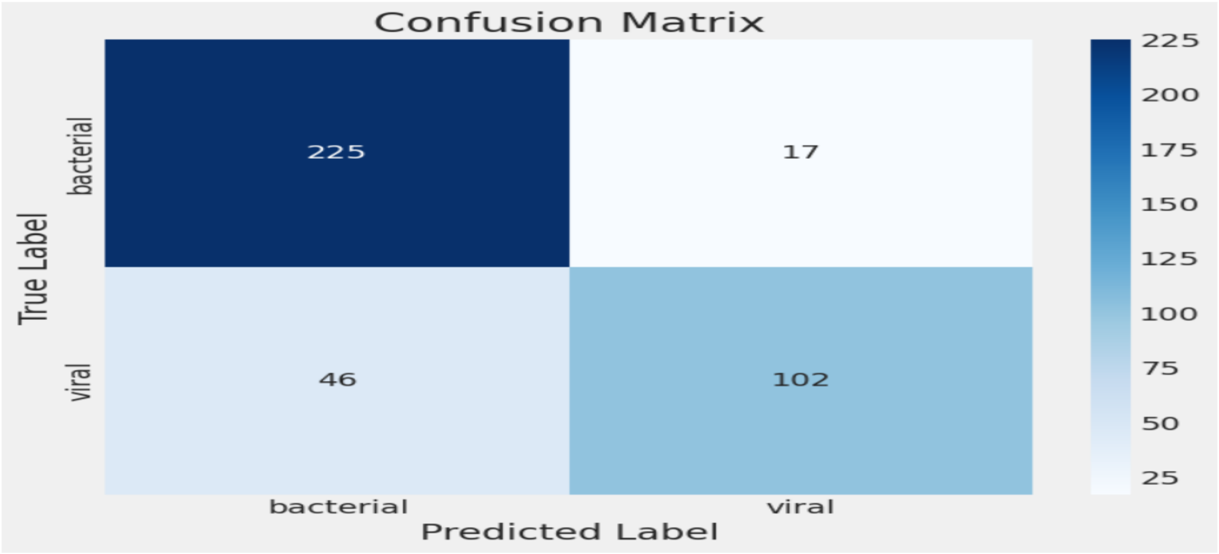
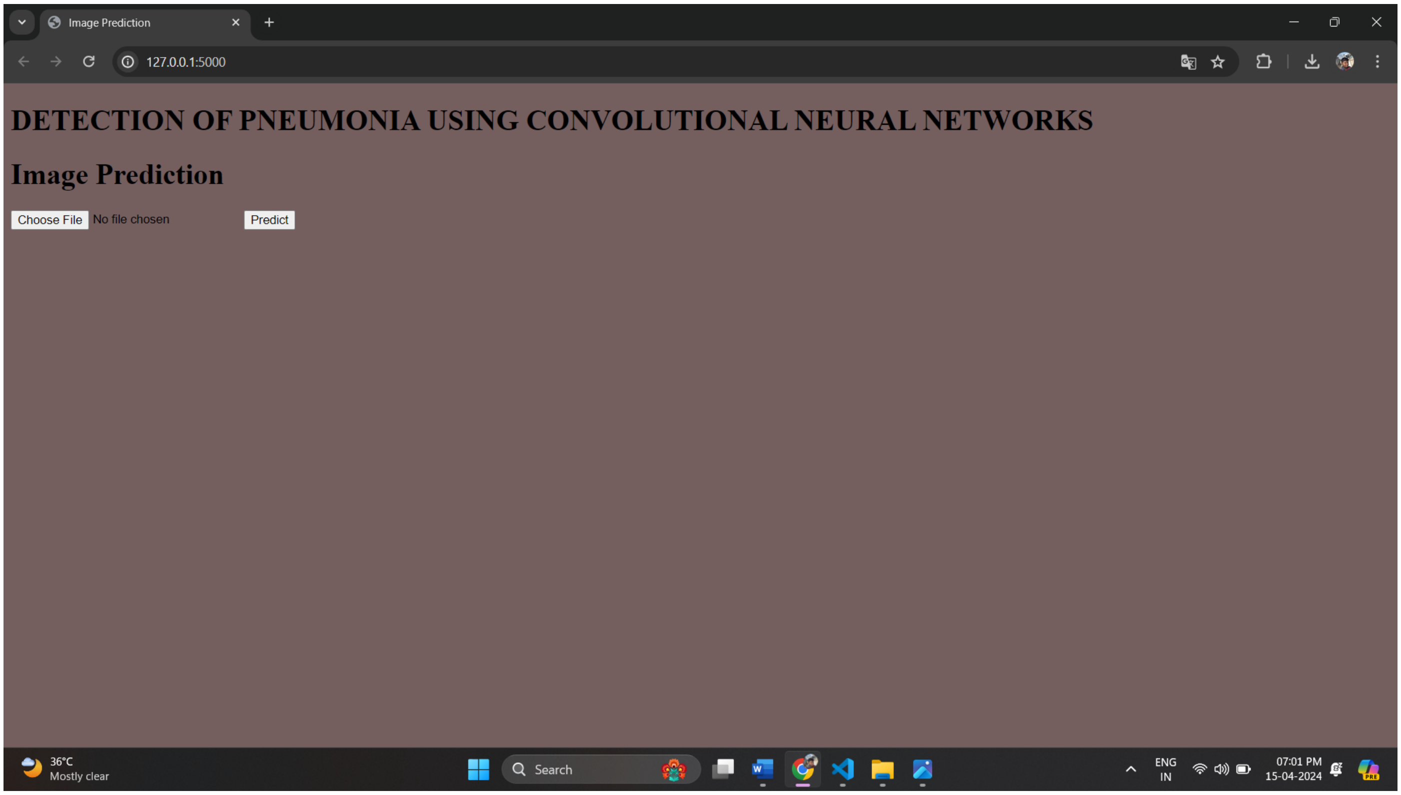


Fig. 5.4 Confusion Matrix

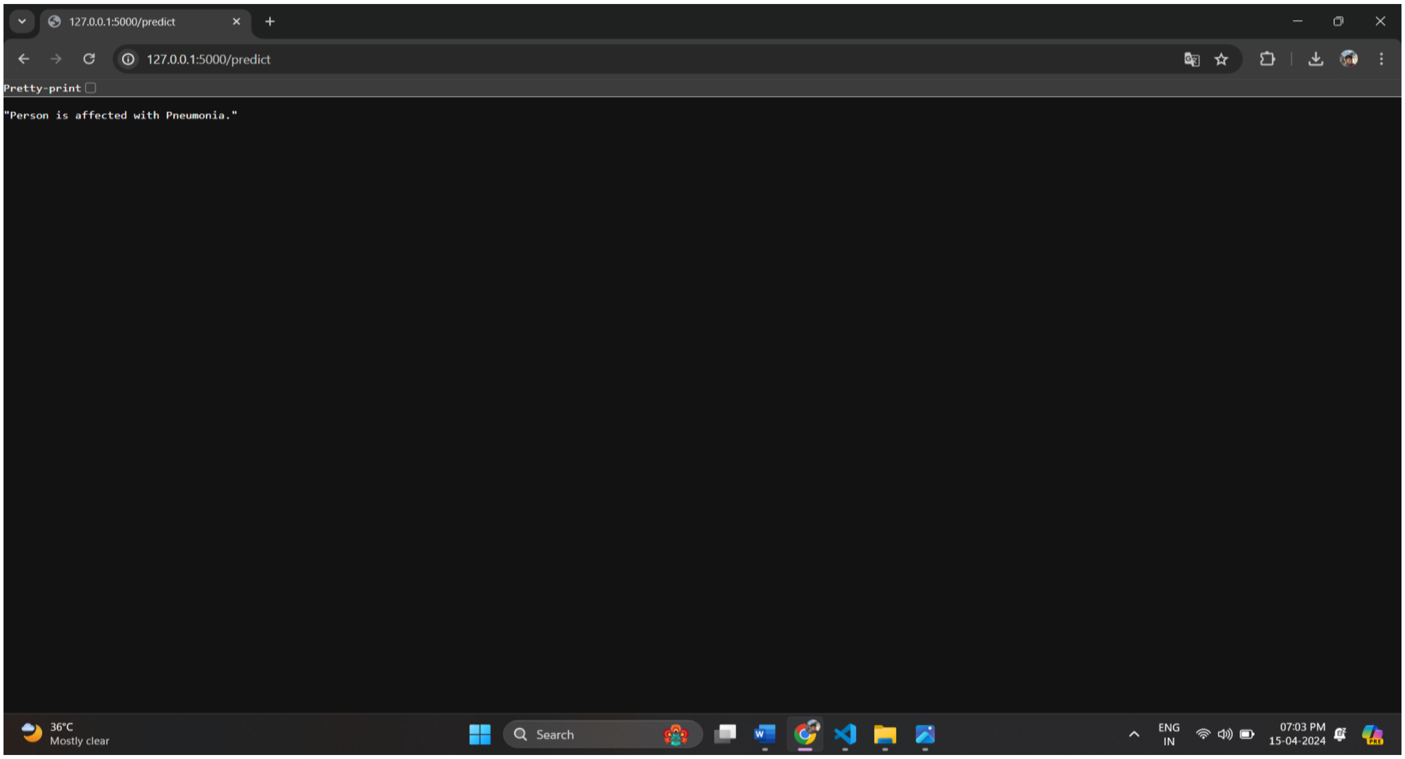
7. SCREEN SHOTS



DETECTION OF PNEUMONIA USING CONVOLUTIONAL NEURAL NETWORKS

Image Prediction

Choose File No file chosen Predict



Pretty-print □

```
"Person is affected with Pneumonia."
```

8. CONCLUSION

This study describes a CNN-based model aiming to diagnose pneumonia on a chest X-ray image set. The contributions in this paper are listed as follows. We designed a CNN model to extract the features from original images or previous feature maps, which contained only six layers combining ReLU activation function, drop operation, and max-pooling layers. The results of the obtained accuracy rate of 96.07% and precision rate of 94.41%, shows that our proposed model performs well in comparison to state-of-the-art CNN model architectures. To illustrate the performance of our proposed model, several comparisons of different input shapes and loss functions were provided.

In the future, we will continue the research to explore more accurate classification architectures to diagnose two types of pneumonia, viruses, and bacteria. According to the description discussed above, the CNN-based model is a promising method to diagnose the disease through X-rays.

9. FUTURE SCOPE

Deep learning-based pneumonia detection and classification is expected to make major strides in the future. We expect more improvements in accuracy, real-time diagnosis capabilities, and interaction with clinical workflows through continued research and development. The implementation of multi-modal approaches and uncertainty estimation techniques is expected to enhance the dependability and comprehensibility of deep learning models, hence promoting confidence among medical practitioners. Widespread acceptance and regulatory approval will depend on a smooth integration into the current clinical systems and extensive validation studies. In the end, these advancements have the potential to transform the diagnosis of pneumonia, enhance patient outcomes, and lessen the strain on international healthcare systems.

Future breakthroughs in pneumonia detection and classification using deep learning are expected to prioritize addressing practical difficulties including scalability, interoperability, and accessibility, in addition to technological advancements. In order to facilitate a smooth integration into clinical practice and provide healthcare professionals with effective diagnostic tools, efforts are being made to improve models for deployment on edge devices and interaction with electronic health records. Furthermore, continuing partnerships between scientists, medical professionals, and business partners will spur innovation and make it easier to move AI-powered therapies from the bench to the patient's bedside. Deep learning in pneumonia diagnosis has the potential to transform healthcare delivery and enhance patient outcomes globally by utilizing these multidisciplinary methodologies.

10. BIBILOGRAPHY

1. Lami F., Rashak H., Khaleel H. A., et al. Iraq experience in handling the COVID-19 pandemic: implications of public health challenges and lessons learned for future epidemic preparedness planning. *Journal of Public Health .* 2021;**43**(Supplement_3):iii19–iii28. doi: 10.1093/pubmed/fdab369. [PMC free article] [PubMed] [CrossRef] [Google Scholar]

2. Thiem U., Heppner H., Pientka L. Elderly patients with community-acquired pneumonia. *Drugs & Aging .* 2011;**28**(7):519–537. doi: 10.2165/11591980-000000000-00000. [PubMed] [CrossRef] [Google Scholar]

3. Gupta Y., Agrawal S. A study of lung disease using image processing in big data environment. *IOP Conference Series: Materials Science and Engineering .* 2021;**1022** doi: 10.1088/1757-899X/1022/1/012030.012030 [CrossRef] [Google Scholar]

4. Sirish Kaushik V., Nayyar A., Kataria G., Jain R. Pneumonia detection using convolutional neural networks (CNNs) *Lecture Notes in Networks and Systems .* 2020:471–483. doi: 10.1007/978-981-15-3369-3_36. [CrossRef] [Google Scholar]

5. Alazzam M. B., Alassery F. The dynamic movement of disaster management systems based on vehicle networks and applied on the healthcare system. *Applied Bionics and Biomechanics .* 2021;**2021**:8. doi: 10.1155/2021/5710294.5710294 [PMC free article] [PubMed] [CrossRef] [Google Scholar] Retracted

6. Cardoso I., Almeida E., Allende-Cid H., et al. Analysis of machine learning algorithms for diagnosis of diffuse lung diseases. *Methods of Information in Medicine .* 2018;**57**(05/06):272–279. doi: 10.1055/s-0039-1681086. [PubMed] [CrossRef] [Google Scholar]

7. Sait U., Shivakumar S., lal kv G, et al. A mobile application for early diagnosis of pneumonia in the rural context. Proceedings of the 2019 IEEE Global Humanitarian Technology Conference (GHTC)); October 2019; Seattle, WA, USA. pp. 1–5. [CrossRef] [Google Scholar]

8. Al-kuraishy H., Al-Maiahy T., Musa R. A., Ali Z. H. COVID-19 pneumonia in an Iraqi pregnant woman with preterm delivery. *Asian Pacific Journal of Reproduction .* 2020;**9**(3):p. 156. doi: 10.4103/2305-0500.282984. [CrossRef] [Google Scholar]

9. Al Mamlook R. E., Chen S., Bzizi H. F. Investigation of the performance of machine learning classifiers for pneumonia detection in chest X-ray images. Proceedings of the 2020 IEEE International Conference on Electro Information Technology (EIT); August 2020; Chicago, IL, USA. [CrossRef] [Google Scholar]

10. Shi F., Wang J., Shi J., et al. Review of artificial intelligence techniques in imaging data acquisition, segmentation, and diagnosis for COVID-19. *IEEE reviews in biomedical engineering .* 2020;**14**:4–15. [PubMed] [Google Scholar]

11. Abdullah Hamad A., Thivagar M. L., Bader Alazzam M., Alassery F., Mahmood Khalil M., Ramesh V. Zelalem meraf, vishal kumar, “dynamic systems enhanced by electronic circuits on 7D. *Advances in Materials Science and Engineering* . 2021;**2021**:11. doi: 10.1155/2021/8148772.8148772 [CrossRef] [Google Scholar]

12. Abdullah Hamad A., Thivagar M. L., Bader Alazzam M., Alassery F., Hajjej F., Shihab A. A. Applying dynamic systems to social media by using controlling stability. *Computational Intelligence and Neuroscience* . 2022;**2022**:7. doi: 10.1155/2022/4569879.4569879 [PMC free article] [PubMed] [CrossRef] [Google Scholar] Retracted

13. Enbeyle W., Enbeyle W., Al-Obeidi A. S, et al. Trend analysis and prediction on water consumption in southwestern Ethiopia. *Journal of Nanomaterials* . 2022;**2022** doi: 10.1155/2022/3294954.3294954 [CrossRef] [Google Scholar]

14. Al-Obeidi A. S., Azzawi S. F. A novel six-dimensional hyperchaotic system with self-excited attractors and its chaos synchronisation. *International Journal of Computing Science and Mathematics* . 2022;**15**(1):72–84. doi: 10.1504/ijcsm.2022.122146. [CrossRef] [Google Scholar]

15. Swetha K. R., Niranjanamurthy M., Amulya M. P., Manu Y. M. Prediction of pneumonia using big data, deep learning and machine learning techniques. Proceedings of the 2021 6th International Conference on Communication and Electronics Systems (ICCES); July 2021; Coimbatre, India. [CrossRef] [Google Scholar]