IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

EARLY LUNGS DISEASE DETECTION USING CONVOLUTIONAL NEURAL NETWORK ARCHITECTURE

[1]Zarka Javaid, [2]Dr.Navneet Kaur Sandhu
[1]M.Tech Student, [2]Assistant Professor,
[1]Computer Science Engineering,
[1]Desh Bhagat University, Mandi Gobindgarh, Fatehgarh Sahib, Punjab- 147301, India

Abstract: The lungs are organs in the human respiratory system that function as an exchange of oxygen with carbon dioxide in the blood. The decline in air quality and COVID-19 in the world, including in India, has an impact on increasing the risk of lung disease. Lung disease that is often encountered is wet lung. Wet lung is a form of acute respiratory infection namely Lungs Diseases that attacks the lungs. The lungs are made up of tiny sacs called alveoli, which fill with water when a healthy person breathes. When a person has Lungs Diseases, the alveoli fill with pus and fluid, which makes breathing painful and limits oxygen intake. Early diagnosis of Lungs Diseases has a major impact on the life of a patient. Diagnosis of Lungs Diseases is generally done clinically (physical symptoms by a doctor). In addition, Lungs Diseases can also be diagnosed through chest radiographs, CT scans, and MRIs. Chest radiograph examination is one of the most frequently used medical imaging examinations because it is more affordable. Reading chest radiographs has drawbacks, namely it is difficult to detect disease, so it takes a long time before medical personnel or doctors diagnose the disease suffered by the patient. One method to overcome this problem is to classify chest radiograph images into certain classes using machine learning. The method used to process the data is multilayer perceptron (MLP). MLP has disadvantages for several types of data, especially for images, but MLP is not well adapted so that it loses spatial information contained in images. Therefore using multiple Convolutional Neural Network Architectures (AlexNet and ResNet) within which the importance of being able to identify the better architecture to obtain the results (diagnose Lungs Diseases) has been essential for the research performed, according to the antecedents of the authors proposed in the methodology, the best architecture analyzed is that of ResNet for having presented high performance in past research. Consequently, The joint classification (Resnet and Alexnet) has proven to be an innovative idea at the time of show a result of a predictive model, as it uses several classifications in turn to determine an end result. Which helped ensure the veracity of the output, which is very important to be able to determine if there is pneumonia in early stage, being this difficult to detect. Designing a Predictive Model for Early Detection of Lungs Diseases using Deep Learning and allowed this research to contribute providing a tool that serves as support for the diagnosis of this disease and reduce the high death rate that exists nationally and internationally.

***Index Terms* - Lungs Diseases Detection, Deep Learning, Artificial Neural Networks, Convolutional Neural Networks.**

I. INTRODUCTION

The lungs are one of the important organs in the respiratory system serves as a place of exchange of oxygen (O_2) with carbon dioxide (CO_2) in blood. In the process of breathing, air will enter through the mouth or The nose then passes through the trachea (throat), bronchi, and bronchioles to get to the alveoli. The exchange of oxygen (O_2) and carbon dioxide (CO_2) takes place in alveoli. Alveoli absorb oxygen from the air and distribute it inside blood to be circulated throughout the body [1-3].

Inflammation of the lungs can cause interference with respiratory. Pneumonia, Pneumothorax, Fibrosis, Infiltration, Nodule, Emphysema, and other lung diseases are the infectious disease that attacks the lungs, causing the air sacs inside the lungs (alveoli) to inflamed and swollen. Infection caused by Pneumonia, Pneumothorax, Fibrosis, Infiltration, Nodule and Emphysema can occur on one side of the lung or both. Pneumonia, Pneumothorax, Fibrosis, Infiltration, Nodule, Emphysema are caused by bacteria, viruses, fungi or parasites where the alveoli are responsible for absorbing oxygen from the air becomes inflamed and filled with water or mucus fluid, so that pneumonia is often called wet lung. These above mentioned Lungs Diseases is the third leading cause of death worldwide, causing 3.23 million deaths in 2019[1-3].

One of the diagnoses of chronic obstructive pulmonary disease is by X-ray. With this examination, an overview of the condition of the lungs with chronic obstructive pulmonary disease are obtained. The diagnosis is then analyzed by a specialist to find out whether there is Lungs Diseases exits or not. X-ray or X-ray examination is a medical imaging technique that uses electromagnetic radiation to take pictures or photos of the inside of the body. The X-ray procedure is done quickly and painlessly. This procedure is part of a supporting examination for the purpose of establishing a more accurate diagnosis. When an x-ray examination is carried out, the machine will send short waves of electromagnetic radiation to the body to scan the condition of the inside of the body. The radiation absorbed by each part of the body will vary. This is what later makes the x-ray photos show color differences from white, gray to black. For example, calcium in the body absorbs the most x-rays, so bones appear white. While fat and other soft tissues absorb less, so they look gray [4,5]. Previous research to detect Lungs Diseases has been carried out through x-ray image results using the CNN (Convolutional Neural Network) algorithm which extracts features from chest X-ray images.

The drawback of the chest x-ray image method is that the findings are still determined manually, so it takes a long time for medical personnel or doctors to diagnose the patient's disease. Therefore, the author tries to solve this challenge in this scheme by categorizing chest x-ray images into several classifications using deep learning methods. The data used in deep learning is in the form of two-dimensional chest x-ray images that can be processed using the Convolutional Neural Network (CNN) approach. Using supervised learning, this method helps in categorizing labeled data. The supervised learning approach operates on a set of labeled data samples. This sample data set is useful for describing the characteristics of the feature size distribution of behavior in each type of application, resulting in the formation of a data behavior model [10]. The CNN method is considered the most effective method for detecting and identifying chest x-ray images [11]. Based on these problems, the authors conducted research using different cases and methods. The author proposes a model that can read chest x-ray images for the diagnosis of various lungs diseases using a Convolutional Neural Network (CNN) with Residual Neural Network Layer 34 (ResNet-34) architecture. From the CNN model, it will later be used to diagnose several dataset labels or classes, namely COVID-19, Normal, Pneumonia, and Tuberculosis (TBC) from chest x-ray images. ResNet18 is used in this study because it is able to overcome the vanishing gradient problem in the CNN model with stable performance.

II. LITERATURE REVIEW

- **Deep learning**

Deep Learning consists of an artificial neural network of multiple layers. It is necessary previously to mention that an artificial neural network in other words is the representation of the system that resembles the neurons of the brain, where each neuronal nucleus is an algorithm that calculates a set of values, these values are called: inputs. These values are processed through a function or mathematical model implemented in these algorithms to eventually produce an output, which is the expected result. Figure 1 shows an example as is the simple representation of an artificial network or neural network [6,7].

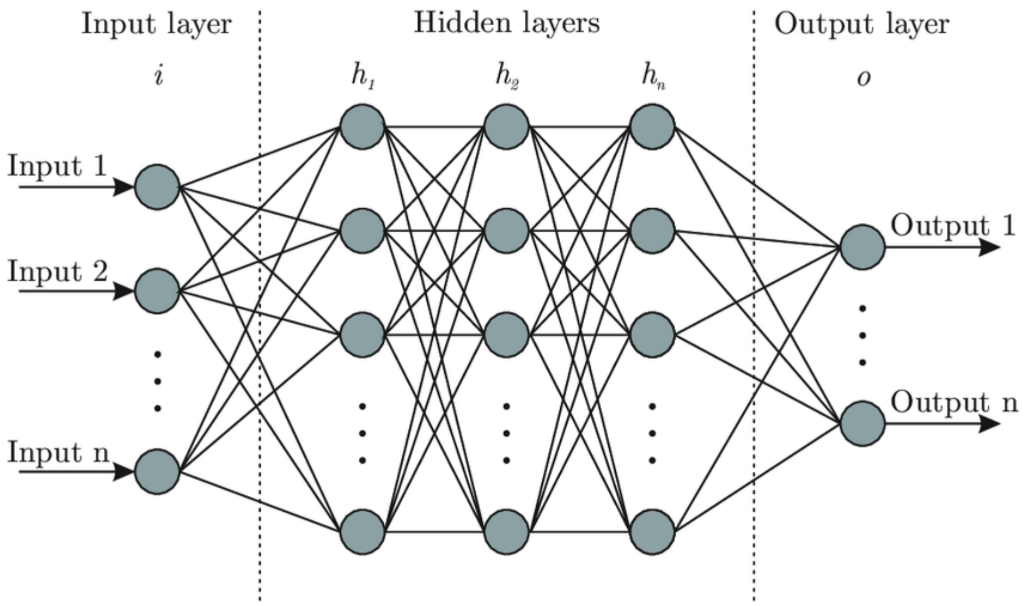

Figure 1: Artificial Neural Network

With this mentioned, in a deep learning neural network, there are sets of " hidden neurons ”or hidden layers intermediate between the input and output layers, whose work is to do all the processing and complex calculation for the interpretation of the object's input and deliver more accurate results. The terminology of "deep" comes from The more hidden (or deep) layers the neural network has, the more complex the processing of the information to be learned by that network. It is explained in an investigation that this type of Neural network is complex and highly computationally expensive [8, 9].

Another of the characteristics of deep learning neural networks is their concept of pre-training, which consists of readjusting the weights of each input value to values balanced in such a way as to avoid using small weights so that processing does not become extremely demanding, or avoid using large weights so as not to generate results deceptively optimal. As the authors explain, each hidden layer of this network of neurons performs an increasingly complex task. For example, if you wanted to recognize the image of some animal, each layer of neurons will have to learn certain characteristics of the object, from the simplest, such as a small segment of an edge, to the most complex like the entire silhouette, so that it recognizes the information it processes in such a way efficient as human vision. Figure 2 represents in a simple way what has been explained, where each layer of the architecture is responsible for recognizing more complex details of the image in the analysis [10, 11].

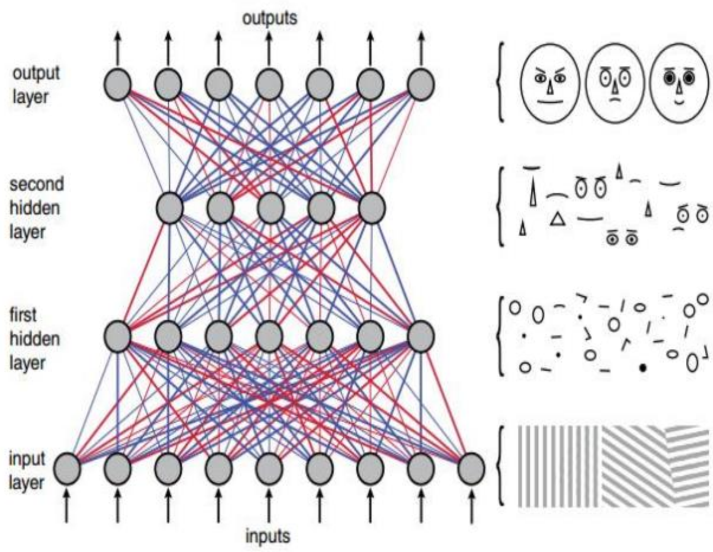

Figure 2: Deep Learning Neural Network

- **Deep Learning Models**

Multilayer Network Perceptron (MLP): Perceptron multilayer networks are currently considered the most basic form in which a neural network is appreciated. MLPs are represented by one or more layers of neurons. The structure of the MLP follows the basic characteristics of the neural networks of deep learning: they have a data entry layer, followed by hidden layers that, in addition to hidden layers, more abstraction of the information, and finally followed by the output layer where predictions are obtained, so this type of architecture is also known as feed-forward. Figure 3 depicts a multilayer neural network [12, 13].

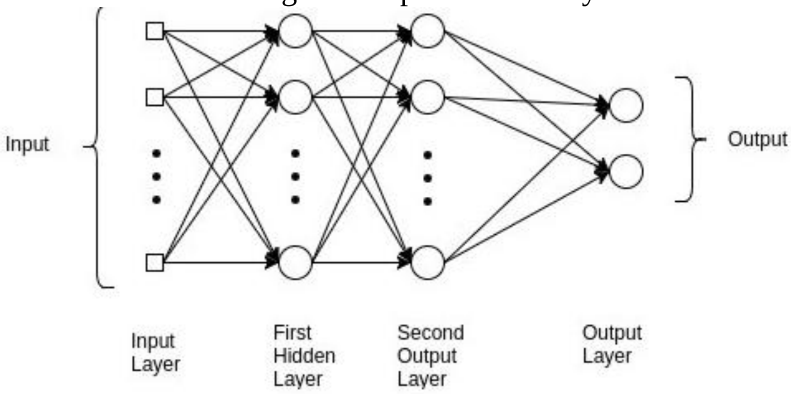

Figure 3: Perceptron multilayer neural network.

The authors point out that the so-called hidden layers of the model are not directly connected to the network environment, so the input layer is independent and its function main is to transmit the input values to the upper layers without there being any type of processing at the entrance. Finally, it expresses that the power of the perceptron model multilayer lies in the non-linear activation functions that it uses, among the most used, the unipolar (or logistic) sigmoid. Next, the equations that compose it: the function of sigmoid activation shown in Equation 2.2, and the weighted sum of the inputs of the neural network which is shown in Equation 1.

$$f(x) = \left(\frac{1}{(1+exp^{-x})}\right) \text{ (eq.1)}$$

$$z_k = \sum w_{ik} s_i + b_k \ (eq. 2)$$

In equation 1, exp^{-x} is an exponential constant equal to 2.71828, x is the sum weighted input, represented as $\sum \cdot_i$. In weighting x like z_k whereas in equation 3, b_k is the bias that verifies how predisposed the input is C to be 0 or 1. The acronym represents the weights multiplied by the input vector w_{ik}. According to what is explained by authors, MLPs are suitable in predictions whose inputs are assigned a class or label. They are also suitable for problems where a true quantity must be predicted given a set of inputs. These data provided are obtained in a tabular format, for what this type of neural network is used tabular data set, prediction problems of classification and regression, for example, if you need to analyze an image, each pixel in this it can be in a cell and together form a long row of data that forms a pattern prediction. Although there are other neural network models that perform this better work, such as convolutional neural networks, below.

- **Convolutional Neural Networks (CNN)**

One of the most recognized networks for image processing are networks neuronal convolutional (CNN) which helps in the recognition of characteristics or patterns. The layers of these networks are divided into 3 types [14-18]: convolutional, grouped (pooling) and fully connected. The representation of these layers can be represent in Figure 4.

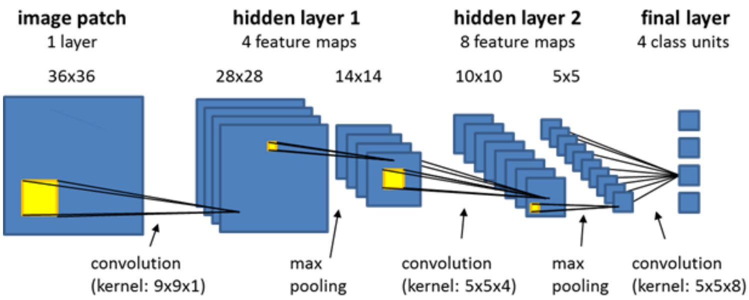

Figure 4: Convolutional Neural Network.

As explained in a previous concept, the goal of convolutional layers is the to extract characteristics from the input (such as images), from a low level, such as edges, color or gradient of an object, down to high-level characteristics that together are obtains a comprehensive understanding of the analyzed object. The grouped or pooling layers are order the reduction of the dimensionality of the object with the intention of reducing the power computational for data processing. The fully-connected layer is one of the last layers of the network in charge of driving the final ranking decision. Each layer of this network of neurons have 2 stages, the feed-forward stage where the first layer of the red takes the input values and feeds the rest of the layers. The second scenario is that of back-propagationstage whose method is used to train the network neuronal where, if errors are detected, it starts searching from the last layers of the network and goes backtracking until you find the neuron that causes the error. The author claims that CNN include complex and varied application domains such as in the monitoring of wind turbines, motion estimation and correction of medical images, prediction of crop yield, advanced image processing, flow prediction aerodynamic, etc. Among the most common architectures of CNN and in which This research will be based on AlexNet and ResNet among others. This work of research is based on deep learning networks, so more will be investigated in their features [14-20].

- **Structure of the CNN**

Next, the theoretical bases of the main characteristics of what CNN is composed:

1. Convolution

The term convolution refers to a mathematical operation that uses two functions f and y, g which produce a third function h, which becomes an integral that expresses the amount of overlap of a function through its displacement on another function, whose objective will be to find an impulse response (T) of a dynamic system in the face of different input types (t). It is denoted from the following equation [14-20] :

$$h = f * g \int_{\infty}^{\infty} f(t)g(t-T)dT \ (eq.3)$$

2. Input layer

It is the first layer within the schema of the architecture of a CNN, whose information To begin with, for this field of research, it is an image. Logically, the layer of input is represented as a three-dimensional grid, denoted by the following equation, where I represents the input layer size, K as image width, L height image and M as number of image channels [14-20]:

J = K * L * M (eq.4)

3. Convolutional layer

These layers are responsible for identifying the characteristics of the images of entry. As convolutional network actuation is described in deep learning, there are more layers of "deeper" convolutions that are responsible for the identification of characteristics more complex to finally classify the object to be recognized. Logically, this layer is represents as weight matrices with connections to those of previous layers, complying with a image filtering operation. The input image rendered as $m * m * r$ have as a filter $n * n * q$ so that finally a denoted matrix of the following is obtained equation [14-20]:

$$(m-n+1) * (m-n+1) * \ p \ (eq.5)$$

4. Layer of grouping or pooling

In this layer the matrix of the convolutional layer is sampled to reduce its dimensions, with the intention of reducing the computational load (reducing the number of features unnecessary) and collaborating with the obtaining of important features in the characterization of the analyzed images. Logically, the pooling layer selects a sub-matrix of size $p * p$ from the upper left side of the convolutional activation matrix. The two stocks, Max pooling y and Average pooling, based on the following [14-20]:

- If the layer is average pooling, the elements of the sub-matrix are taken and the mean, the result of which is stored in the first position of the output array.
- If the max pooling layer, the element with the highest value of the sub-matrix is taken and stores in the first position of the output array.

In this way, the following sub-matrix in is taken $p + 1$ position on the right and performs the same operation. When the trade reaches the right edge, the position moves $p + 1$ down, and the operation starts again from left to right. When the matrix is size of the convolutional filter, it means $n * n * q$ the output of this layer is denote from the following equation:

$$\frac{n}{p} * \frac{n}{p} * q \ (eq.6)$$

5. Classification layer or fully connected

The feature data will be converted into a feature vector, which will be the input data of this layer, which is basically in charge of classifying these vectors in labels or classes for which the neural network has been trained, which will ultimately be the result. Logically, the dimensions of this output are represented in the vector $[1 * 1 * N]$ where N is the number of output labels or classes. A type of classification layer used in different CNN architectures, it is the Softmax layer that, unlike Logistic Regression, allows the classification of a problem in k different classes depending on the network under investigation. Logically, from an input x where the hypothesis $h_\theta(x)$ estimates a probability $P(y = k|x)$ that a sample belongs to the class $k = 1,2,\dots\ k$ the following equation is denoted that allows to normalize the results, where the sum of all the elements of the vector must be 1 [14-20].

$$P(y = k|x) = \frac{1}{\sum_{j=1}^{k} \exp(\theta^{(j)T} x)} \ (eq.7)$$

6. Joint classification

Joint classification is a logical operation where each classifier of a model predictive P_x gives a “vote” that represents your qualifying output S_x of the classes that has analyzed. These predictions will be stored in a prediction vector2) = $X_i = (P_1 = S_1, P_1 = S_2, \dots, P_x = S_x$ Which is denoted in the following joint classification equation [14-20].

$$\hat{y} = argmax\left(\sum_{t=1}^{T} \widehat{W_t} f_t\ (x)\right)(eq.8)$$

- **CNN Training Method**

In these theoretical bases, the Backpropagation method that will be used as method of training the neural network with the aim of adjusting its parameters to find the weights of the neurons that allow the correct identification of the images by analyze. This training is made possible according to the error that will allow to measure the performance of the weights and readjust them based on the research objectives [14-20].

- **Backpropagation**

It is a neural network training algorithm which starts with an input pattern that propagates through all layers of CNN until it gets its own output. Thereafter, it is compared with the expected classification output and a calculation of the error of all neurons in the output layer and propagates backward from the output to the layers intermediate (or hidden) of the network. These neurons have an error value proportional to their contribution on the total error value of the network. With this information, you can adjust the weights of neurons. The cycle of training and adjustment is called epoch, which is it will repeat until the weights are stable and network performance is optimal. Logically, denotes from the following equation [14-20]:

$$E\left(x; W^{(1)}, W^{(2)}\right) = \frac{1}{2}\sum_{i=1}^{m}(t_i - o_i^{(2)})^2 \text{ (eq.9)}$$

Where $o_i^{(2)}$ is the output layer, of the set of x y values that approximates the value wantedt_i. $W^{(1)}$ represent the weights between the input layer connections y and the layers intermediate and y $W^{(2)}$ are the weights between the connections of the intermediate layers and the layer of exit [14-20].

- **CNN optimization algorithm**

This class of algorithms allows the reduction of loss or error in training of the networks, consequently, an efficient update of the weights and adjustment of the model hyperparameters for better network learning rate. Then it explain the algorithm used in the research methodology: descending gradient stochastic.

- **Stochastic descending gradient (SGD)**

This algorithm allows a calculation of the decrease of the loss function based on a set of weights. After this, the weights are updated for each example of the training, in the direction indicated by the gradient and descends the error curve until reaching a local minimum. Each training example is selected in a way randomization before starting training, which makes this algorithm more efficient with relative to others that perform the same task (such as the standard descending gradient). And denotes from the following equation, where w2 is the set of weights, Q) is the contribution of training example and η∇ the siphon cup threshold [14-20].

$$w_{t+1} = w_t - \eta \nabla Q_i(w_t) \ (eq.10)$$

- **CNN Architectures**

As mentioned in a previous point, pre-trained CNNs are models that already they have been trained by a dataset with certain types of classes or labels. There are many architectures today, or are also formed by the evolution or combination of others, generating new versions of them or totally new designs. In this case, and under the research works on which this research work is based, the architectures that will be studied and developed in the proposed design, will be AlexNet and ResNet [14-20].

- **AlexNet Architecture**

AlexNet is a convolutional neural network 8 layers deep, 5 convolutional and 3 fully-connected. This architecture uses the function of activation known as rectified linear unit (ReLU), represented in the following equation:-

$$f(x) = \max(0, x) \begin{cases} x_i & if\ x_i \geq 0 \\ o, & if\ x_i < 0 \end{cases} \text{(eq.11)}$$

Unlike the sigmoid function, the ReLU function (equation 2.12) transforms the x input values 0, it means that the negatives are x_i in if $x_i < 0$ and the neuron does not know will activate, leaving the positives x_i on if $x_i \geq 0$ as information for the following layers, eliminating the problem of gradient fading or loss. AlexNet architecture now it has been trained by a data set called ImageNet. Thanks to this, this network is capable to classify images into 1000 different categories, including everyday objects and animals, so you already have rich knowledge to effectively detect images that contain these elements. For this, the images processed by this architecture must be resized and cropped to a scale of 227x227 pixels before being converted to a value of entry [14-20]. The graphic representation of this architecture can be seen in Figure 5.

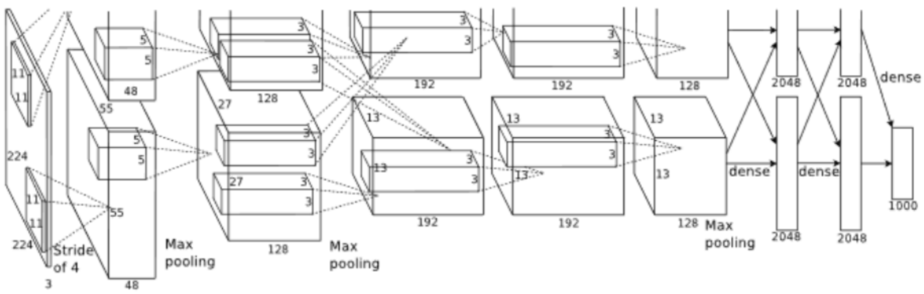

Figure 5: CNN AlexNet Architecture.

- **ResNet Architecture**

The ResNet architecture, also called Residual Networks, has several versions depending on the number of layers in which it is formed: 18, 34, 50, 101, 152 and 1202. Focusing on the ResNet18 architecture, which is used in many investigations for its high performance, It consists of 18 million parameters, accepts 224 x 224 pixel input images and is It consists of 5 convolutional layers, 1 pooling layer, 1 fullconnected layer and 1 softmax layer, its operation consists of increasing the number of network layers through a connection residual, which is represented by the residual of the difference of the mapping of the underlying layers. Logically it is understood that if the residual of this mapping $F(x) = H(x) - x$, then the network learn to $F(x) + x$ Using the ReLU activation function, this will improve the process of learning without getting into a gradient cancellation problem. Figure 7 shows the ResNet architecture [14-20].

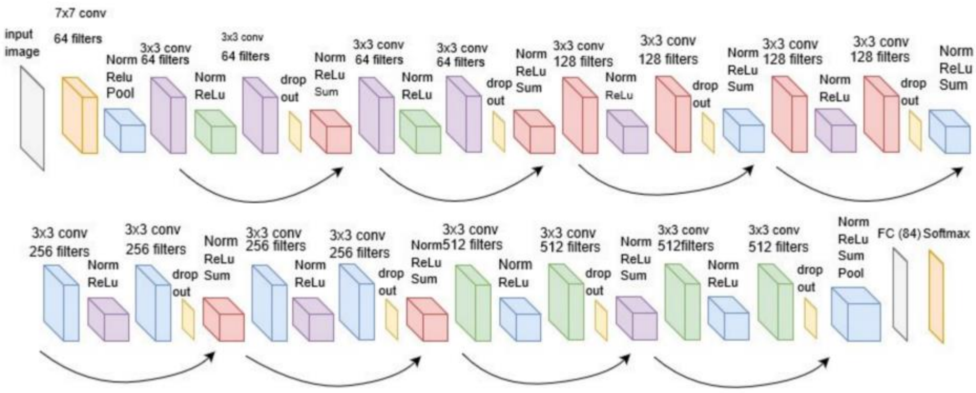

Figure 7: ResNet Architecture

III. PROPOSED METHODOLOGY

Based on the research chosen, a methodology consisting of 5 phases. As shown in Figure 8 below, the phases would be: Selection of the set of data (1), Pre-processing of data set images (2), Data augmentation (3), transfer learning (4) and joint classification (5) using 2 architectures together, for best results.

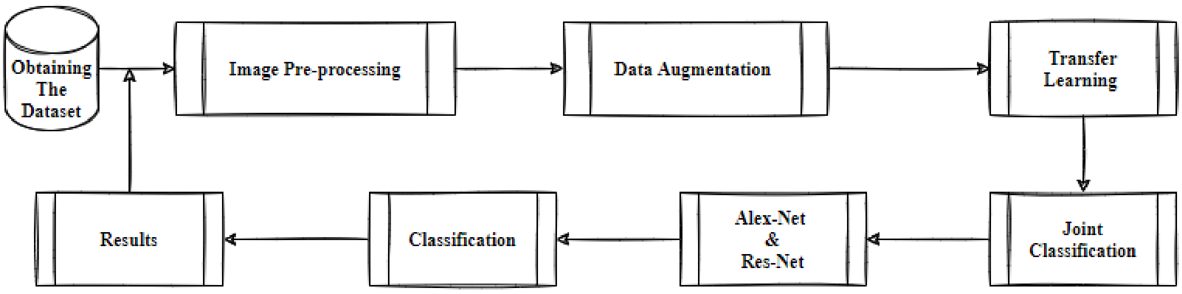

Figure 8: Proposed Model

- **Obtaining the Data Set**

In this phase of the proposed methodology, it is essential to obtain a set of data that contain the radiographic images and the necessary attributes to be able to be analyzed and processed for the identification of pneumonia at an early stage, within which the images that will be needed will have to be of two kinds: Patients with early pneumonia and healthy patients. The consolidation attribute is the attribute that meets the objective of identifying chest x-rays showing pneumonia in the first phase known as congestion or consolidation. With this explained, it was decided to obtain the public dataset called Chest X-ray Dataset of 14 Common Lungs Disease Categories, belonging to the NIH Clinical Center. This database has a weight of 2.09GB, with images in PNG format with a resolution of 1024 x 1024 pixels and a total of 5600 approx X-ray images, which contains the following attributes described in Table 1:

ATTRIBUTES
IMAGE INDEX
FINDING LABEL S
FOLLOW UP #
PATIENT ID
PATIENT AGE
PATIENT GENDER
VIEW POSITION
ORIGINAL IMAGE WIDTH
ORIGINAL IMAGE HEIGHT
ORIGINAL IMAGE PIXEL SPACING X
ORIGINAL IMAGE PIXEL SPACING Y

Table 1: Chest X-**raydataset** main attributes

To obtain the distribution of images and to be able to obtain the data set of training, validation and tests for the design of the predictive model, the present research builds on research for which the quantities of images according to Table 2, which contains images of the 14 classes in the consolidation phase and images without any type of disease the segregation is as under.

	TRAINING	VALIDATION	TESTS
NORMAL	3000	5000	5000
WITH DISEASE	2000	303	303
TOTAL	5000	5303	5303

Table 2 Training, validation and testing distribution

As the first phase of the proposed methodology, obtaining the data set, They will carry out 4 activities: the collection of information, the loading of the dataset, the division of the data sets (training, validation and tests) and finally the balancing of classes, having as requirements what is mentioned as under:-

- Information Gathering (Phase 1)

  The dataset that has as an attribute will be selected the presence of pneumonia at an early stage (consolidation) and that it was used in the methodologies of scientific sources related.

  Obtained the data set, whereas be selected classes of images representing the state normal of a patient and with the diseases (mentioned in the previous point).

- Load the dataset
  - Define the route where the images are located radiographs and attach them as a set of total data.
- Division into training sets
  - The total size of the data set will be calculated.
  - The proportions of the data set will be defined training and validation.
  - A temporary set will be defined that will calculate the size total minus training and validation, from which the test set will be obtained.
  - All three sets will have a ratio based on the proposed by the author in the methodology,
- Class balancing
  - Positive (with disease) and negative (without disease) class will be balanced using the multinomial distribution.

- Image pre-processing (Phase 2)

Followed by obtaining the training, validation and testing data set, the details the activities of the second phase of the proposed methodology: the pre-processing of images, which means preparing the information for model training predictive. The activities to be carried out will be: Use the resizing technique of images and visual normalization of images, which should be applied to all sets of the data set. The activities are detailed below:-

- Use the technique of resizing images
  - The images will be normalized by setting the data increase with the resizing of the size of dataset images in for each model 224 x 224 x 3 pixels, using the bilinear interpolation technique.
- Data augmentation (Phase 3)

  - The training, validation and test sets have been configured for their normalization in the phase 2, The activities of the phase 3 operation: Increase of Data for the training set of the proposed model.
  - Use Random Resized technique Crop :- Configure data augmentation with clipping randomized images from the dataset training.
  - Use Random Horizontal technique Flip :- Configure data augmentation with the horizontal transformation of the training data set,
  - Compilation of augmented data:- All data augmentation techniques will be entered as parameters for setting the preset processing and data augmentation of the 3 sets. The data augmentation is based on the model shown in Figure 8.

- Transfer Learning
  - After obtaining the training set with increased data, the validation set and of tests, we will proceed with phase 4: Transfer of Learning, where the pre-trained architectures, the knowledge of basic extraction of characteristics of these, and a new model will be formed with the capacity to classify radiographic images.
  - Configuration of the pre-models trained:- The training sets, validation and test with the settings established in the phase 2 and 3.Pre-trained models will be loaded.
  - Take initial layers of the pre-model trained:- Only the convolutional part will be loaded (leaving the prediction layer) and its set of weights of the pre-trained model with database ImageNet. They will be frozen (the parameters will not be modified) the first layers of the pre-trained network. Therefore, declaring the last prediction layer as trainable. The pre-trained model will retain its function of ReLU activation.
  - The final layers of the pre-model will be trained to start the new training. The binary cross loss function will be used thereafter the entropy using the Adam optimization.
  - Train the new model:- The amount of training periods will be defined with the Deep Learning (CNN) methodology. The learning rate will be defined and the learning reduction rate according to the CNN methodology. The training of the newly model will be compiled trained. The training will be repeated for each architecture of the proposed model.

- Joint classification using AlexNet and ResNet18 trained models (Phase 5).
  - Joint classification:- Finally, in phase 5: Joint classification, all the results of classification of each architecture proposed by the methodology, in order to obtain a result final. In the following section 4, we will proceed to explain the activities to be carried out in the process joint classification of radiographic images for the detection of Lungs Diseases in stage early.
  - Get prediction of each Model :- The classifier value of each model will be obtained trained classification of vector-trained models prediction The classification result of each architecture in a prediction vector denoted by the following equation 8 to obtain the final classification by majority vote.
  - Departure :- The classification model adhered to a system doctor will be able to classify radiographic images as "With Lungs Diseases" or "Without any Diseases (Normal)".

IV. SIMULATION AND RESULTS

- **Development of The Proposed Predictive Model**

For the development of the design of a predictive model for the early detection of Pneumonia using Deep Learning and Computational Vision, according to the previous chapter where The phases of the proposed methodology are described, the development of the same.

- **Loading The Dataset**

To start with the design of the proposed model, the libraries must be imported necessary to carry out each activity and then the data set of the radiographic images from the directory where it is located. In case it is necessary to check which are the labels of the dataset, we will proceed to read the csv file that you have attached to the dataset and that describes its attributes. Additionally, the training of the model will begin with the new fullyconnected. This process will be carried out in the computer's GPU by the configuration model.to (Device).

```
import os
import numpy as np
import pandas as pd
from tqdm import tqdm
from operator import itemgetter
from collections import OrderedDict

from PIL import Image
import seaborn as sns
import matplotlib.pyplot as plt
```

```
import torch
from torch import optim,nn
import torch.nn.functional as F
from torchvision import transforms as T,models
from torch.utils.data import Dataset, DataLoader, random_split
from torchvision.utils import make_grid
import pandas as pd
device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
print(device)
```

```
data = pd.read_csv('/content/drive/MyDrive/DataSet/sample_labels.csv')
data.head()
```

	Image Index	Finding Labels	Follow-up #	Patient ID	Patient Age	Patient Gender	View Position	OriginalImageWidtl
0	00000013_005.png	Emphysema|Infiltration|Pleural_Thickening|Pneu...	5	13	060Y	M	AP	305
1	00000013_026.png	Cardiomegaly|Emphysema	26	13	057Y	M	AP	250
2	00000017_001.png	No Finding	1	17	077Y	M	AP	250
3	00000030_001.png	Atelectasis	1	30	079Y	M	PA	299
4	00000032_001.png	Cardiomegaly|Edema|Effusion	1	32	055Y	F	AP	250

Figure 9: Data Set Loading.

- **Predictive Model Classes**

To obtain the data for the design of the proposed predictive model, according to the investigations proposed in the present work, where several researchers decided use public domain images, it was decided to use a data set called Chest XRay Dataset of 14 Lungs Diseases, belonging to the NIH Clinical Center. The source where it is obtained, the main characteristics of this dataset, and that will be used to the present investigation is detailed as under:-

```
pathology_list = ['Cardiomegaly','Emphysema','Effusion','Hernia','Nodule','Pneumothorax','Atelectasis',
                  'Pleural_Thickening','Mass','Edema','Consolidation','Infiltration','Fibrosis','Pneumonia']

for pathology in pathology_list :
    data[pathology] = data['Finding Labels'].apply(lambda x: 1 if pathology in x else 0)

data['No Findings'] = data['Finding Labels'].apply(lambda x: 1 if 'No Finding' in x else 0)
```

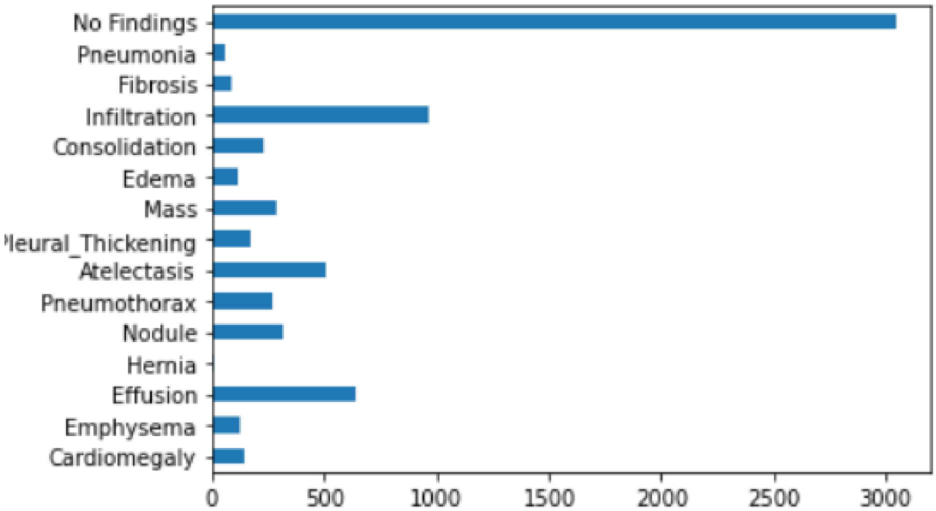

Figure 10: Predictive Model Classes

According to the previous chapter, once the data set is obtained, we proceed to use images with pneumonia with other lungs diseases in the consolidation stage (example are shown in Figure 11) and images without any type of disease (Figure 48 shows two examples), for which the must extract a percentage of the total.

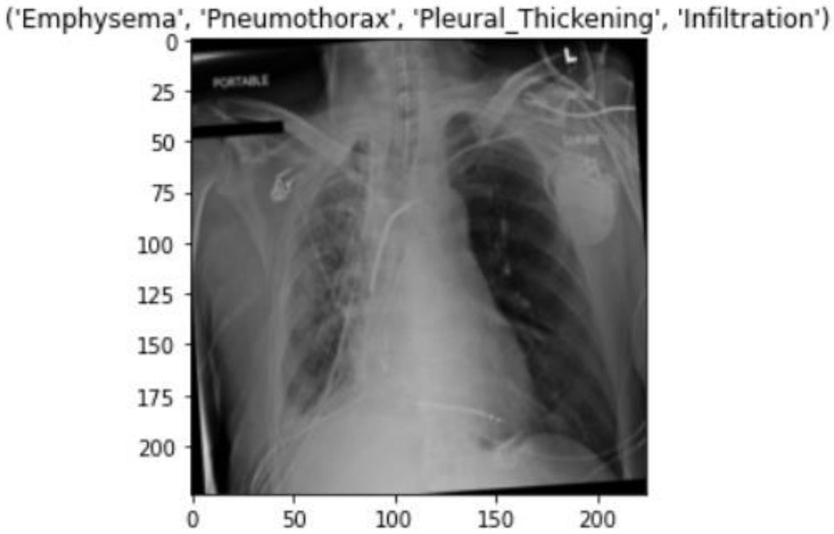

Figure 11: Consolidation Stage

- **Division into Training, Validation and Testing sets**

Once the set of images has been obtained with which to proceed to work, these must be divided into three folders: training, validation and tests (train, validation, test). In the research selected as the basis of the proposed methodology, it is recommended that the training set must be 80% of the total set of the data set, the remaining 20% will be divided between the validation and tests set, each subset would be 10% of the total. The distribution is seen in the following Figure 12.

```
trainset, validset, testset = random_split(trainds, [5000,303,303])
print("Length of trainset : {}".format(len(trainset)))
print("Length of testset : {}".format(len(testset)))
print("Length of validset : {}".format(len(validset)))
```

```
Length of trainset : 5000
Length of testset : 303
Length of validset : 303
```

Figure 12: Dataset division.

- **Data balancing**

The multinomial distribution will be used to mathematically obtain samples the negative class (Not Finding) and the positive class (Consolidation) and calculate how many times the surplus class appears in the sample. Once the majority class is obtained, must construct new probabilities of the minority class until in the sample both are equal. This would be equivalent to the following code:-

```
pos_weights = freq_neg
neg_weights = freq_pos
pos_contribution = freq_pos * pos_weights
neg_contribution = freq_neg * neg_weights
```

```
df = pd.DataFrame({"Class": pathology_list, "Label":
                   "Positive", "Value": pos_contribution})
df = df.append([{"Class": pathology_list[l], "Label": "Negative", "Value": v}
                for l,v in enumerate(neg_contribution)], ignore_index=True)
```

```
trainloader = DataLoader(trainset,
                         batch_size = 32,
                         shuffle = True)

validloader = DataLoader(validset,
                         batch_size = 32,
                         shuffle = False)

testloader = DataLoader(testset,
                        batch_size = 32,
                        shuffle = True)
```

Figure 13: Balancing of Classes.

- **Image Pre-processing**

- Use the image resizing technique

In the activity of resizing images to the required number of pixels, the The main function of the bilinear interpolation function will be to eliminate pixels within the array of pixels representing the image to be processed. For example, in the case of X-rays obtained, each one has an original size TO 1024x1024 pixels and will be resized to AlexNet network input size Ñ 227x227. Which each dimension will be divided into P;/ P< and 6;/ 6< to obtain the new matrix or grid that will be the new image and where color scale values will be entered. How bilinear interpolation will take all 4 pixels 7 (0.0) 7 (0.1) 7 (1.0) 7 (1.1) neighbors to reduce them to 1 you will need to calculate the coordinates of these, which would be (P, 6), (P, 6 + 1), (P + 1, 6) and (P + 1, 6 + 1) why finally stay with (P, 6) and so on with every 4 neighboring pixels until the required size is reached. The same i know would apply for each network if ResNet18 Ñ is 224 x 224. This would be equivalent to following code in Python language, calling the following Tensorflow code structure:-

```
[ ] data_transform = T.Compose([
        T.RandomRotation((-20,+20)),
        T.Resize((224,224)),
        T.ToTensor(),
        T.Normalize(mean=[0.485, 0.456, 0.406],
                    std=[0.229, 0.224, 0.225])
    ])
```

Figure 14: Resizing Image

- **Data Normalization**

Since the information in the dataset is images, each channel must be normalized RGB color of each input image, so you would have to experiment with values within the gray scales to determine a correct normalization. However, as the transfer of learning from a pre-trained neural network is going to be used in a set of larger data (ImageNet), you can use mean and standard deviation values that are networks already have their knowledge integrated. The values used for the mean u in each channel They are: 0.485, 0.456 and 0.406. On the other hand, the values used for standard deviation or in each channel are: 0.229, 0.224 and 0.225. This would be equivalent to the following code in the Python language, calling the following algorithm transforms.Normalize () for the case of ResNet18 as depicted in Figure 4.5.

- **Transfer Learning**

Once the data augmentation settings are known in the proposed model, we will proceed to design the learning transfer, where the pre-trained models must be loaded (AlexNet and ResNet18) as they will be used as feature extractors, it means that the weights of the first layers of the model will be kept except the last layers, such as fully connected, that will need to be initialized (that is, train them from scratch).

```
for param in model.parameters():
    param.requires_grad = False

model.fc = nn.Sequential(
    nn.Linear(512, 14),
    nn.Sigmoid()
)

model.to(device)
```

Figure 15: Loading Existing Model

- **Initial configuration of pre-trained models**

First, the training validation and test datasets must be loaded (integrating the data pre-processing, balancing and augmentation tasks). To do this, the function Sequential. For experiments you can alter the linear function values (the amount of samples displayed by epoch), (number of threads for each block of generated samples) and shuffle if you want the trained information to be random. The parameter values 512, 14 will indicate if the pre-processing and data augmentation of images are will process on the computer's GPU. The following figure shows the loading of the sets of training, validation and testing to be used for transfer of learning.

```
for param in model.parameters():
    param.requires_grad = False

model.fc = nn.Sequential(
    nn.Linear(512, 14),
    nn.Sigmoid()
)

model.to(device)
```

```
      (bn2): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
      (downsample): Sequential(
        (0): Conv2d(64, 128, kernel_size=(1, 1), stride=(2, 2), bias=False)
        (1): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
      )
    )
```

Figure 15: Loading Pre-Trained Model

- **Train The New Model**

The training rate value has been defined as a parameter in the algorithm Adam optimizer, from the previous section, under the parameter lr = 0.001, as mentioned previously. With this, the training reduction is defined with the ReduceLROnPlateau() function, whose parameters will be the full configuration of the optimizer algorithm, the numbers of iterations for the reduction and the reduction factor.

```
optimizer = optim.Adam(model.parameters(),
                       lr = 0.0001)
schedular = optim.lr_scheduler.ReduceLROnPlateau(optimizer,
                                                 factor = 0.1,
                                                 patience = 4)
epochs = 15
valid_loss_min = np.Inf
```

Figure 16: Adam Optimizer Inculcation

With the models prepared and the training configurations defined, we will proceed with the training phase of each model, defining the number of epochs. The numbers of epochs, training rate and learning rate reduction per epoch in each architecture will be defined in accordance with what is mentioned by the author of the selected methodology. These may vary based on the results of the experiments, taking into account that the author trained his model to classify all the diseases (14 classes).

```
model.train()
for images,labels in tqdm(trainloader):
    images = images.to(device)
    labels = labels.to(device)

    ps = model(images)
    loss = weighted_loss(pos_weights,neg_weights,ps,labels)

    optimizer.zero_grad()
    loss.backward()
    optimizer.step()

    train_loss += loss.item()
avg_train_loss = train_loss / len(trainloader)
```

Figure 17: Evaluating Model using Epochs

- **Joint Classification**

An encapsulated functionality will be created in charge of classifying the images X-rays and determine its output value. The parameters will be a new input image and the trained mode By using this functionality with a new image, the model will be able to display the value of classification of radiography. In the following figure, the functionality is used, but for the ResNet18 trained model:

```
model = models.resnet18()
model.load_state_dict(torch.load('/content/drive/MyDrive/model/resnet18-5c106cde.pth'))
```

Figure 18: RestNet Architecture Integration

The main idea of using the joint classification in the proposed model is to obtain an influential final result from all the predictions that have been obtained from the previous task.

- **Result output**

The importance of the predictive model proposed is to allow the identification by means of chest x-rays the presence of Lungs Diseases at an early stage as under therefore ot would be a great help to medical personnel in prescribing helping to fight this disease at an early stage.

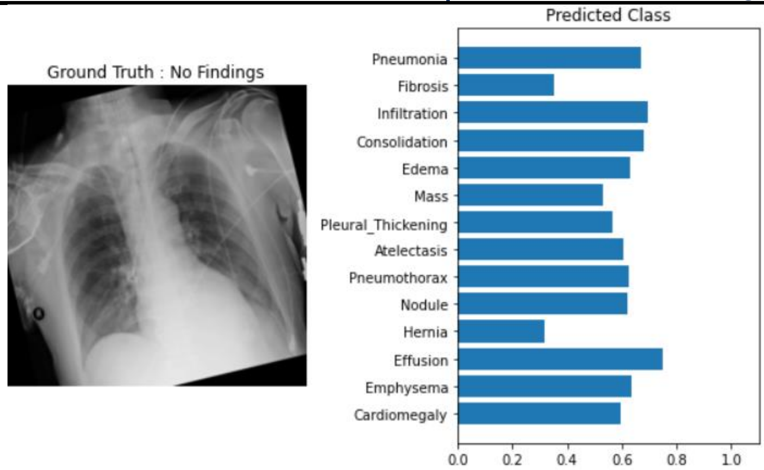

Figure 19: Result Output.

In the present research work, as authors carried out the analyzes of each phase of the proposed methodology based on deep learning determined that to provide a better diagnosis requires a more in-depth training of radiographic images, in this way provide a better response as a result. A major training of the neural networks of the proposed predictive model, will provide a better result in the accuracy of the results. Within the limitations that have been found to be able to have more real data at the reality, is not being able to have local radiographic images, as a whole the test was carried out based on a set of international data, unfortunately in the country the Public health entities do not provide the facilities to carry out network training neuronal, and in the presence of a global pandemic such as Covid-19 the situation is more difficult, having a database of local images of Lungs Diseases in different stages would be of great help for the present work and future research. The more training of the neural networks the results would be more optimal, the proposed model only considers three architectures such as AlexNet and ResNet18, if the training will be carried out with a greater number of architectures, it could be obtained better results, with greater precision and little margin for error, continuing to train the Neural networks would help during the implementation stage of the proposed design.

V. CONCLUSIONS

The present research work allowed to know the reality of the country with respect to the few investigations carried out in the medical sector with reference to the use of intelligence artificial for the detection of various diseases, which served as a stimulus to deepen more on the subject and that it serves for future research in favor of people's health, reaching the objectives proposed at the beginning of the research work.

1. The importance of a robust data set has been fundamental for this type research where it is required to analyze Lungs Diseases an early stage, since it has allowed the optimal training of the model and the objective is reached. Pre-processing techniques, normalization and data augmentation, served as the fundamental basis for the model to be capable of to learn to recognize new cases and in turn enrich your knowledge, all due to that early-stage Lungs Diseases can be very difficult to detect, so the more variety of training, the higher degree of detection the model will have.

2. Computer vision has played an important role in this methodology, since which has made it possible to obtain the key characteristics of X-ray images thanks to the layers convolutional values of pre-trained architectures. The use of techniques such as Transfer Learning, has made it possible to take full advantage of the power of computational vision of architectures experienced, save resources and training effort.

3. There are multiple convolutional neural network architectures, as well as the evolution of new architectures, within which the importance of being able to identify the better architecture to obtain the results, has been essential for the research performed, according to the antecedents of the authors proposed in the methodology, the best architecture analyzed is that of ResNet for having presented high performance in past research.

4. The joint classification of alexnet and resnet has proven to be an innovative idea at the time of show a result of a predictive model, as it uses several classifications in turn to determine an end result. Which helped ensure the veracity of the output, which is very important to be able to determine if there is Lungs Diseases in early stage, being this difficult to detect. OG: Designing a Predictive Model for Early Detection of Lungs Diseases using Deep Learning has allowed this research to contribute providing a tool that serves as support for the diagnosis of this disease and reduce the high death rate that exists nationally and internationally.

VI. FUTURE SCOPE

Being able to count at the national level with a data set of radiographic images of various diseases and radiographs of healthy people of various ages, and publicly accessible for research work would facilitate future projects with training and results with information of real radiography of local patients, which could be implemented by the application in various medical establishments with support from the public and private sectors, providing a great support in the diagnosis of various diseases at an early stage, especially in sectors of our country where they do not have many qualified medical personnel. The study of neural networks is important in the development of new techniques and base of many future investigations, especially directed to the medical field where lacks qualified personnel in a decentralized way, for which it would be necessary to teach of this from the first cycles in engineering careers, updating the curricular mesh of universities, in addition to providing logistical support for development and implementation. Detecting diseases at an early stage is important to avoid their propagation and difficulties in the patient, for which the implementation of the predictive model for the early detection of Lungs Diseases at an early stage would help considerably in lower the mortality of this disease, especially in cities where the cold is strong and does not they have financial resources for their detection, treatment and prevention.

REFERENCES

[1] Andreae S. Lexikon der Krankheiten und Untersuchungen. Stuttgart: Thieme; 2008.

[2] Menche N. (Hg.) Biologie Anatomie Physiologie. München: Urban & Fischer/ Elsevier; 2012.

[3] Pschyrembel W. Klinisches Wörterbuch. Berlin: De Gruyter; 2014.

[4] Ekiz, Ahmet & Kaplan, Kaplan & Ayvaz, Erdal. (2022). Chest X-Ray Images Classification with CNN.

[5] Rahman, Md.Tanvir. (2021). Pneumonia Detection Using X-Ray Image Processing Using CNN. 10.6084/m9.figshare.14370806.

[6] Şen, Zekâi. (2023). Deep Learning. 10.1007/978-3-031-29555-3_9.

[7] Qamar, Usman & Raza, Muhmmad Summair. (2023). Deep Learning. 10.1007/978-3-031-17442-1_8.

[8] Schonlau, Matthias. (2023). Neural Networks. 10.1007/978-3-031-33390-3_14.

[9] Chandrasekaran, Arvind. (2023). Neural Networks. 95-99. 10.5121/csit.2023.131208.

[10] Luqman, Saqib. (2023). Deep Learning: Unraveling the Secrets of Neural Networks.

[11] Dinov, Ivo. (2023). Deep Learning, Neural Networks. 10.1007/978-3-031-17483-4_14.

[12] Verma, Akanksha & Kumar, Dr. (2020). Multilayer Perceptron Artificial Neural Network: A Review.

[13] Silva, Ivan & Spatti, Danilo & Flauzino, R.A. & Bartocci Liboni, Luisa & Reis Alves, Silas. (2017). Multilayer Perceptron Networks. 10.1007/978-3-319-43162-8_5.]

[14] Hvitfeldt, Emil & Silge, Julia. (2021). Convolutional neural networks. 10.1201/9781003093459-15.

[15] Ullo, Silvia & Sebastianelli, Alessandro & Rosso, Maria & Spiller, Dario & Puglisi, Erika & Nowakowski, Artur & Bernardi, Mario & Cimitile, Marta. (2021). Convolutional neural networks. 10.1049/PBTE098E_ch5.

[16] Ketkar, Nikhil & Moolayil, Jojo. (2021). Convolutional Neural Networks. 10.1007/978-1-4842-5364-9_6.

[17] Hatuwal, Bijaya & Thapa, Himal. (2020). Lung Cancer Detection Using Convolutional Neural Network on Histopathological Images. International Journal of Computer Trends and Technology. 68. 21-24. 10.14445/22312803/IJCTT-V68I10P104.

[18] Alhudhaif A, Polat K, Karaman O. Determination of COVID-19 pneumonia based on generalized convolutional neural network model from chest X-ray images. Expert Syst Appl. 2021 Oct 15;180:115141. doi: 10.1016/j.eswa.2021.115141. Epub 2021 May 4. PMID: 33967405; PMCID: PMC8093008.

[19] Ullah, A., Elahi, H., Sun, Z. et al. Comparative Analysis of AlexNet, ResNet18 and SqueezeNet with Diverse Modification and Arduous Implementation. Arab J Sci Eng 47, 2397–2417 (2022). https://doi.org/10.1007/s13369-021-06182-6

[20] B. Donnelly and T. Munasinghe, "Binary Classification Edge Cases in Cyclones Using AlexNet and ResNet Neural Networks," 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 2021, pp. 4153-4155, doi: 10.1109/BigData52589.2021.9671337.