BACHELOR OF SCIENCE IN ELECTRONIC AND TELECOMMUNICATION ENGINEERING

A Deep CNN Biomedical Imaging Technique for Detecting Infected Covid Patients

Submitted by

Md Sheikh Ejab (T181048)

Supervised by

Engr. Md. Razu Ahmed

Professor

Department of CCE

International Islamic University Chittagong

Department of Electronic and Telecommunication Engineering
International Islamic University Chittagong
Kumira, Sitakunda, Chattoagram-4318

DEDICATION

This thesis work is dedicated to all of our esteemed teachers and our beloved parents for their prayers and support towards the achievement of our goal.

CERTIFICATE OF APPROVAL

The thesis titled "A Deep CNN Biomedical Imaging Technique for Detecting Infected Covid Patients" Submitted by **Md Sheikh Ejab** bearing Matric ID**: T181048;** of Academic Year 2022, Session: Autumn 2022 has been found as satisfactory and accepted as partial fulfillment of the requirement for the B.Sc. in Electronic and Telecommunication Engineering on International Islamic University Chittagong and approved as to its style and contents for the examination held on 19^{th} November 2022.

Approved by:

Md. Razu Ahmed

Supervisor and Professor

Department of Computer and Communication Engineering

Faculty of Science and Engineering

International Islamic University Chittagong

Kumira, Sitakunda, Chattagram

CANDIDATES DECLAREATION

It is hereby declared that the work presented in this thesis has not been submitted elsewhere for the award of any degree or diploma, does not contain any unlawful statement.

Md Sheikh Ejab

Id: T-181048

ACKNOWLADGEMENT

In the name of Allah, the Most Merciful, the Most Merciful. All praise and glory are to Allah (SWT) for blessing us with abundant opportunities and giving His mercy and guidance to us throughout our lives. May the peace and blessings of Allah be upon the Prophet Muhammad (pbuh) who guides and inspires our lives. We express our sincere gratitude and indebtedness to our thesis supervisor **Md. Razu Ahmed** for his initiative in this field of research, valuable guidance, encouragement throughout the time of research. We express our thankfulness to **Engr. Syed Zahidur Rashid**, Chairman of the Department of Electronic and Telecommunications Engineering, IIUC for providing us with best facilities in the department and his timely suggestions. We are also thankful to **Abdul Gafur**, Convener of our thesis for his dedication and sacrifice. We also express our sincere gratitude to all of our teachers for their best efforts throughout our academic years. And so far, we remember our parents for their support in our lives. And we also express our gratitude to our friends, well-wishers, for their direct or indirect involvement in the completion of this thesis.

ABSTRACT

The newly discovered coronavirus (COVID-19) sickness has caused havoc for people all over the globe and placed the whole planet on high alert. A second wave of the deadly illness has occurred because coronavirus infections have returned. This has been confirmed in most countries. The contagious virus may cause symptoms such as an itchy throat all the way up to pneumonia, and it has been responsible for the deaths of thousands of people while infecting millions more throughout the world. Getting a COVID-19 infection diagnosed as soon as possible is very important because it helps stop the illness from spreading and makes it easier to keep sick people apart and treat them. Recent studies in radiological imaging show that X-rays and CT scans of the lungs can show how COVID-19 infection will progress in people with severe symptoms. The main goal of this experiment is to quickly diagnose COVID-19 progression and other lung diseases by looking at X-rays of people with many symptoms. Using modern applications of artificial intelligence, a cutting-edge Deep CNN model that is both new and very good has been made to predict COVID-19 infections quickly and accurately. The model that has been made (healthy) divides X-ray pictures of the lungs into two groups: COVID and non-COVID. Accuracy, sensitivity, precision, and the f1 score are some of the assessment measures used to determine how well the suggested systems work. In the present study, a dataset consisting of X-ray samples was used. It was found that the CNN model had a high recognition rate of 99.38% and was in line with the current state of the art. The suggested model is very effective and accurate. It could help radiologists, and other medical professionals diagnose COVID-19 infection early in people who are showing signs of the disease. In addition, investigate various transfer learning approaches to contrast them with our model.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
ANN	Artificial Neural Network
K-NN	K- Nearest Neighbors
HOG	Histogram of Oriented Gradients
SIFT	Scale Invariant Feature Transform
FC	Fully Connected
ReLU	Rectified Linear Unit
NLP	Natural Language Processing
TP	True Positive
FN	False Negetive

Chapter 1
Introduction

1.1 Artificial intelligence (A.I.)

A.I. is the ability of a digital computer or computer-controlled robot to do tasks often associated with intelligent beings. Artificial general intelligence refers to the effort to create systems with cognitive abilities similar to those of humans. These abilities can include the ability to reason, find meaning, generalize, or learn from experience (A.G.I.). It has been shown that since the advent of the digital computer in the 1940s, computers can be taught to do tasks requiring human-level knowledge. Finding proofs for mathematical theorems and playing chess are two instances of such work. However, despite constant advancements in computer processing speed and memory capacity, no programs have yet been developed that can match the human level of flexibility across a wider variety of topics or in endeavors that need a considerable degree of common knowledge. However, in some instances, computer programs have surpassed the performance of human specialists and experts. Therefore, A.I. in this narrow meaning, may be found in fields as diverse as medical diagnosis, web search engines, and text-to-speech conversion.

1.2 How can one define intelligence?

When evaluating whether or not insects are intelligent, even the most sophisticated actions they exhibit are never taken into account; on the other hand, all human activities other than the most basic are ascribed to intelligence. Psychologists seldom use the word "intelligent" to refer to a single characteristic of human beings; instead, they emphasize the sum of a person's many varied talents instead of using the phrase "intelligence." Research in the subject of artificial intelligence has focused the bulk of its emphasis on the following components of intelligence: learning, reasoning, problem-solving, perception, and the use of language [1].

1.3 Branches of artificial intelligence

The objective of the field of study known as artificial intelligence (A.I.) is to create robots with the same capacity for logical thinking as humans. Reactive machines, Limited memories, theory of mind, and self-awareness are the four subcategories that fall under the umbrella of artificial intelligence. The levels of production, employment, and competitive behavior are all being significantly impacted by artificial intelligence at this point [2].

- **Reactive Machines:** Reactive machines take in information from the outside world and plan their actions based on that. The machines do specific jobs and only know how to do that one job. When the same thing happens over and over, the machines always act the same way. In the 1990s, I.B.M. made a machine called Deep Blue that could play chess competitively. Deep Blue could predict chess moves by knowing where each piece was on the board.

- **Limited Memories:** Machines with a smaller memory may nonetheless make sound judgments by relying on current data. When analyzing observational data, the machines use the conceptual framework they were designed with. After a certain period of time, the observational data is deleted permanently.
- **Theory of mind:** The idea of mind says that machines can think and make decisions based on how they make people feel. This means that they can interact with other people. Machines are still developing, but many of them can do things that humans can. For example, think about voice assistant apps that can understand simple commands and prompts but can't hold a conversation.

Self-Awareness: Conceptualization, the creation of wants, and a comprehension of their internal states are all ways in which self-aware computers display intelligent behavior. Alan Turing designed the Turing Test in 1950 to determine whether or not a machine might act indistinguishably like a human being.

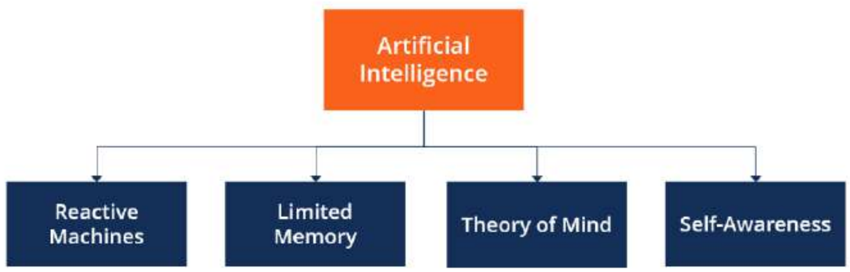


Figure 1.1: Types of Artificial Intelligence

1.4 How was Machine Learning (ML) achieved?

Comparing how A.I. makes a computer intelligent to how people acquire knowledge is the most straightforward approach to grasping how a machine may become clever. Take, as an example, a young kid who is learning how to ride a bicycle. The youngster hops on the bike, grabs the handlebars, and crosses their fingers so that they can maintain their balance and keep their bearings. Instead of learning how to ride a bike via knowledge of the mechanics of bicycles, a youngster learns how to ride a bike by trial and error.

Artificial intelligence is created in the same way that a youngster learns the unwritten rules of riding a bicycle via practice. This is done by repeatedly simulating different scenarios.

1.5 Machine Learning

The term "machine learning" refers to a technique that may be used to teach a machine to carry out a variety of tasks, such as making predictions, suggestions, estimates, and so on, by making use of historical data or by drawing on the experiences they have had in the past. Machine learning is a method of teaching computers to mimic human behavior by using historical data and data based on predictions made from that data [3].

The concept of machine learning may be broken down into three main categories, which are as follows:

- The primary issue that fascinates us is referred to as the "task," and it is this issue that we will focus on in this article. For example, this activity or issue could be connected to forecasts, suggestions, or estimates.
- Learning from historical or previous facts, as is the case with experience, and using that knowledge to estimate and solve future problems are two definitions of experience.
- In machine learning, performance is defined as the degree to which a given machine can resolve a given problem or task and provide optimal results. What we mean by this is that each given machine has the potential to tackle any issue or job associated with machine learning. However, the performance varies greatly depending on the specific machine-learning problems being addressed.

1.5.1 Techniques of Machine Learning

The following four categories are the primary divisions for machine learning techniques:

- **Supervised Learning:** Learning through supervision is achievable anytime a machine has access to sample data, that is, data with suitable labels for its inputs and outputs. With proper labels, the model's precision may be tested in several ways. Using our prior knowledge and carefully labeled samples, we can accurately predict the future using the supervised learning approach. It begins by analyzing the preexisting training dataset. And then, it gives an inferred function that can predict the values of the outputs.
- **Unsupervised Learning:** Learning is machine learning in which the output is not known in advance, and the computer is trained using just some input samples or labels. Unlike supervised learning, the computer may not always provide the correct output if the training material is not labeled or classified. While supervised learning is more often utilized in real-world commercial situations, unsupervised learning may be useful for discovering insights inside datasets and making conclusions about previously unrevealed patterns. Unsupervised learning is valuable, but it's not very common.
- **Reinforcement Learning:** The machine learning approach of reinforcement learning is based on the idea of feedback. Agents, which are computer programs, must investigate their surroundings, complete tasks, and get feedback in the form of incentives. Computer programs are responsible for this form of education. They are rewarded favorably for good behavior and punished severely for bad behavior. A reinforcement learning agent aims to amass the most significant potential sum of rewards. The lack of labeled data forces the agent to rely on its own experiences for training.
- **Semi-supervised Learning:** This term describes a technique that combines aspects of both supervised and unsupervised learning. It performs operations on labeled and unlabeled data, albeit it excels at the latter. However, in many instances, the data lack labels. This has the added advantage of reducing the cost of the machine learning model; since labels are costly, albeit for practical reasons,

it may only incorporate a subset of labels. More than that, it boosts the efficiency and precision of the machine learning model.

1.5.2 Classification algorithms in machine learning

Deep learning has a classification challenge, which has traditionally been addressed using machine learning methods. Among the many classification-related career opportunities, sentiment analysis is the most common. Every algorithm is tailored to solve a specific problem; hence it's not uncommon for different jobs to need different algorithms. Classification is a significant area of research in statistics, and additional datasets call for different approaches to classification. The five most popular machine learning algorithms are listed below.

1.5.2.1 Logistic Regression

As the name implies, logistic regression is used to predict a true/false result. It's possible to display this data as a simple yes/no, pass/fail, alive/dead, etc. Independent variables are examined, and the results are classified into one of two binaries. The dependent variable is always categorical, although the independent variables might be either numerical or categorized. In the following form:

$$P(Y=1|X) \text{ or } P(Y=0|X) \quad (1)$$

Specifically, it determines the probability of Y given X. If you want to know if a certain word is likely to have a positive or negative connotation, this might help (0, 1, or on a scale between). Or it may be used to give a probability between 0 and 1 to each object in an image to determine what it is (a tree, a flower, grass, etc.).

1.5.2.2 Naive Bayes

The likelihood that a given data point belongs to or does not belong to a specific category may be calculated using the Naive Bayes method. In the process of text, the analysis may be used to assess whether or not individual words or phrases correspond to a specific "tag" (category).

1.5.2.3 K-nearest neighbors

K-nearest neighbors, often known as k-NN, is a pattern recognition technique that uses training datasets to determine the k people who are the most closely related to future samples. Data is given to the category of its nearest neighbor based on a computation when k-NN is used for classification. This assignment occurs automatically. If k is equal to 1, it is assigned to the class that is most similar to 1. K's categorization is determined by considering the number of its adjacent nodes.

1.5.2.4 Decision Tree

Because of its ability to properly arrange categories, the supervised learning method known as a decision tree is beneficial for jobs requiring categorization. It operates in the same manner as a flowchart, grouping data points into two categories that are similar at a time, moving from "tree stem" to "branches" to "leaves," where the categories become more connected to one another. This leads to sorts inside types, enabling organic categorization to occur with minimal human monitoring.

1.5.2.5 Random Forest

The random forest method is an extension of the decision tree technique. In the decision tree technique, you first create a large number of decision trees using training data, and then you fit your new data into one of the trees as a "random forest." The random forest method is an example of an extension of the decision tree technique.

1.5.2.6 Support Vector Machines

Beyond simple X/Y prediction, a support vector machine (SVM) may be trained to classify data within a range of polarities. Once a novel feature is implemented and shown to be helpful, it creates a new area of study that has been investigated for decades. Feature extraction is generally performed thoroughly automated across all DL algorithms. This motivates academics to find ways to extract discriminative features with as little manual labor and domain expertise as possible [4].

These strategies use a layered data representation architecture, with the lower-level characteristics being extracted by the first layers and the higher-level ones by the last. The first layer is responsible for pulling the basic features, while the previous layer is responsible for removing the more complex ones. Always remember that A.I. was the original motivation for this kind of design, which attempts to simulate the activity of the brain's vital sensory centers. The human brain can automatically derive data representation utilizing several settings. Precisely speaking, the acquired scene information is the "input," and the detected things are the "output" of this process. This method is reminiscent of the way the human brain works. Therefore, this emphasizes DL's key benefit.

1.6 Difference between A.I. vs. ML vs. DL

In the year 2020, humans use artificial intelligence daily, using it to power apps such as music recommender systems, Google maps, Uber, and many more applications. Despite this, there is still a great deal of ambiguity around the phrases ai, ml, and dl. Artificial intelligence is a scientific discipline much like mathematics or biology. It investigates methods to construct intelligent programs and robots capable of creatively finding solutions to problems, which has traditionally been regarded as a human prerogative. At a symposium on computer science held in Dartmouth in 1956, the phrase "artificial intelligence" was initially used for the first time. The term "artificial intelligence" (A.I.) refers to the endeavor to replicate how the human brain operates and, based on this understanding, to develop more complex computers. The researchers thought it shouldn't take too much time to figure out how the human mind operates and then digitalize that information [5].

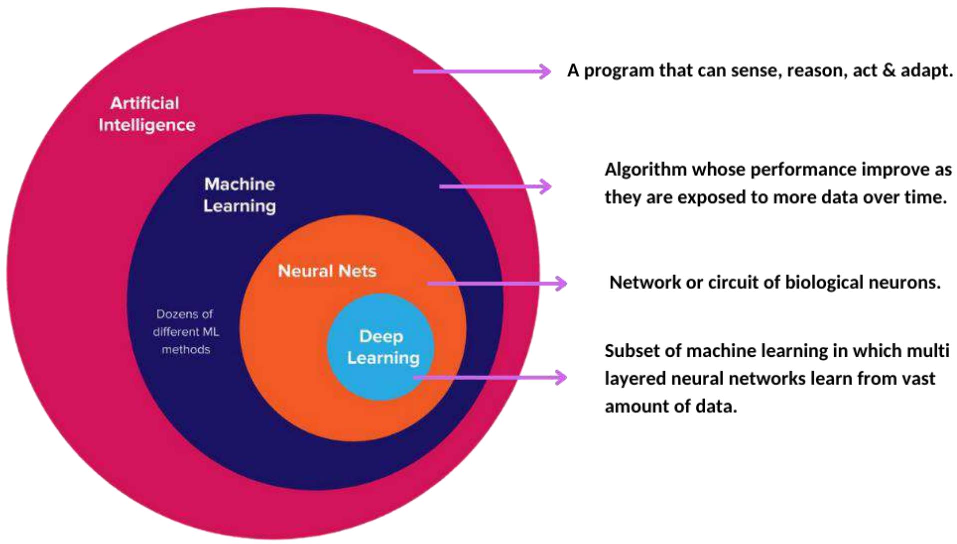


Figure 1.2: A.I. vs. ML vs. DL

One may think of DL as a subset of ML. An artificial neural network (ANN) is a piece of computer code or hardware that mimics how human brain cells communicate with one another to complete a job. Synthetic neural networks include layers of linked neurons that are fed data as inputs and are then given weights. A series of activations, analogous to the synapses between real neurons, is then output after some mathematical processing. To function, Deep Learning doesn't rely on predetermined rules created by humans, but instead, it utilizes a massive quantity of data to "train" itself to map the input to the desired labels correctly. The algorithms (artificial neural networks, or ANNs) used to create DL include several layers. Each layer offers a unique interpretation of the data that has been input into the system [6, 7].

1.7 Deep Learning and Traditional Machine Learning

Using typical machine learning methods, one needs to execute several sequential operations to complete the classification task. Pre-processing, Feature Extraction, Discretionary Feature Selection, Learning, and Classification are the several steps involved in this process. A machine learning method's effectiveness is also heavily dependent on the characteristics used to train the model. Incorrect categorization of the different groups is possible if traits are picked in a biased manner. However, DL can automatically learn feature sets for several tasks, but conventional ML methods cannot. [6,8].

With DL, it's possible to do the two learning and classification tasks simultaneously. Since the advent of big data and its rapid development in recent years, DL ML algorithms have gained much attention [9, 10].

It has streamlined the enhancement of several learning domains [11, 12], including picture super-resolution [13], object identification [14], and image recognition [14, 16], and is currently in ongoing development in regards to innovative performance for numerous ML tasks [17-18].

DL performance has recently surpassed human performance on jobs like picture categorization (Fig. 4). This technique has affected almost every scientific discipline. The usage of DL has already disrupted and revolutionized most sectors and enterprises. The world's top technological and economy-focused firms are competing to advance DL. Forecasting the time required to make automobile deliveries, certifying loan requests, and predicting movie reviews are just a few examples of the numerous domains in which DL's performance exceeds that of humans [19].

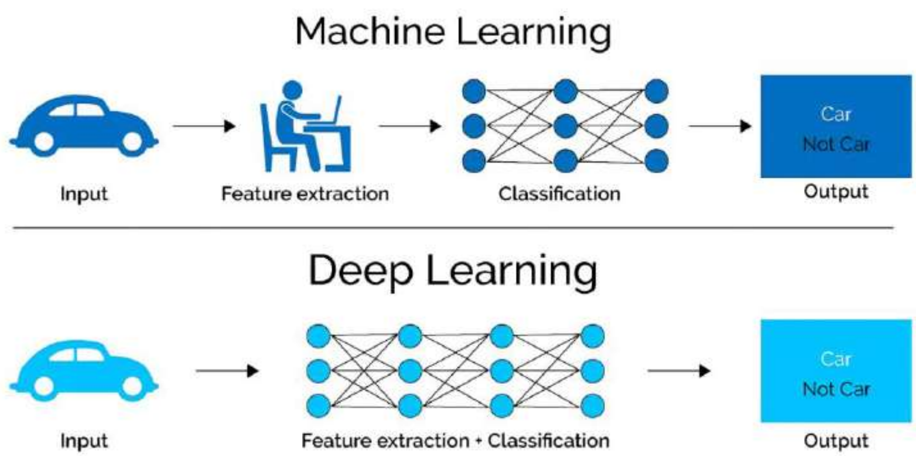


Figure 1.3: Core difference between ML and DL

1.8 Introduction and Overview of Deep Learning

Machine learning (ML) has gained significant traction in recent years, both in the academic community and in industry, where it is being used in a wide range of tasks, including text mining, spam detection, video recommendation, image categorization, and multimedia idea retrieval. A wide variety of research initiatives have used ML [20]. The ML method of deep learning (DL) is often employed in these contexts [21-23].

Another name for DL is Representation Learning (R.L.). Unpredictable increases in data availability and remarkable advances in hardware technologies like High-Performance Computing both have a role in the ongoing development of new research in deep and distributed learning areas. Because of both, novel research may be undertaken (HPC) [24]. Although DL evolved from the classic neural network, it is much superior to its forerunners. In addition, deep learning simultaneously employs transformations and graph technologies to build multi-layer learning models. Modern deep learning (DL) techniques have been shown to be effective in various settings. Natural language processing (N.L.P.), image processing, and audio/voice data processing are only a few examples [25-29].

The success of a machine learning system often hinges on how accurately the data is represented. Using the correct data representation has been proven to provide better results than relying on a less-than-ideal one. For some years, feature engineering has been a hot topic in machine learning studies, and as a consequence, it has inspired a wide range of studies. The raw data is the starting point for this technique, which aims to extract valuable properties from them. Also, it requires a lot of human labor and is very sector-specific. Examples of many types of features were compared and contrasted in the field of computer vision, including a histogram of oriented gradients (H.O.G.) [30], a scale-invariant feature transform (SIFT) [31], and a bag of words (BoW) [32]. As seen in Figure 1 below, the human brain is nature's most impressive technological achievement.

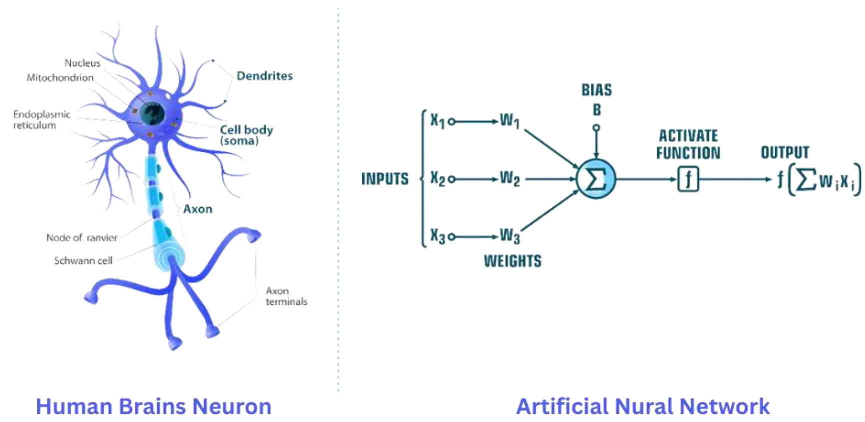


Figure 1.4: Human brain neuron to artificial neuron network concept

1.8.1 How Deep Learning Functions works

Since neural network topologies are used in the bulk of deep learning approaches, "deep learning" models are also frequently referred to as "deep neural networks." In most contexts, the term "deep" refers to the number of hidden layers inside a neural network. Deep neural networks may contain as many as 150 hidden layers, while regular neural networks typically only have two or three layers. To teach deep learning models, vast data sets and specific topologies of neural networks are used. These networks learn features directly from the data, eliminating the need for humans to extract features manually.

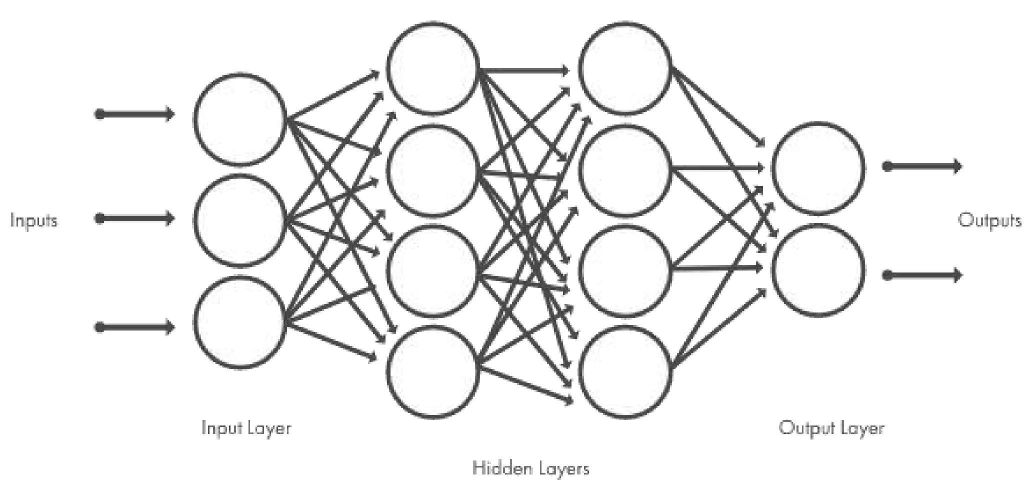


Figure 1.5: Layers of linked nodes make up neural networks. Hundreds of hidden layers may be found in networks.

1.8.2 Convolutional Neural Network

There are many kinds of deep neural networks, but one of the most prevalent is the convolutional neural network (CNN or ConvNet). This architecture uses convolutional layers and combines previously learned features with the incoming data, making it an excellent choice for processing 2D data such as photos.

Because CNN eliminates the need for hand-crafted feature extraction, there is no longer a need to choose picture categorization characteristics manually. CNN can function via the process of directly extracting attributes from photographs. The relevant factors are not pretrained; instead, the network learns them as it is being trained on an image collection. The precision of deep learning models used in computer vision applications such as object classification is improved by the autonomous extraction of features used in these models.

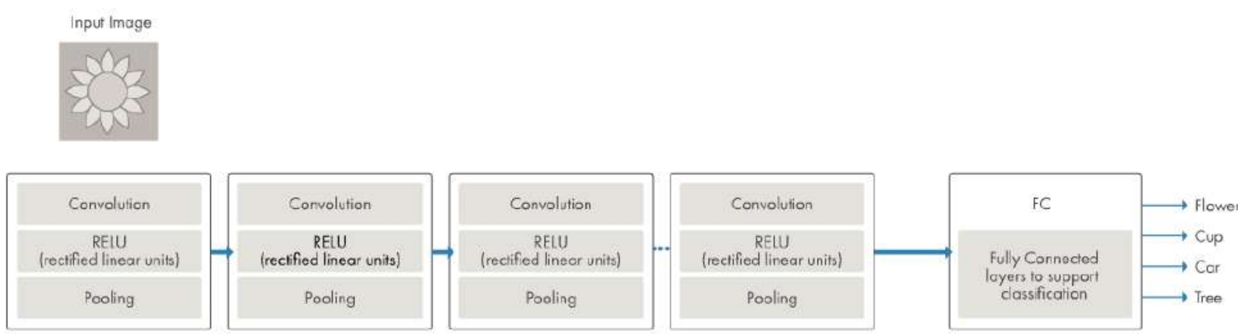


Figure 1.6: Illustration of a network with several convolutional layers

CNN can train to distinguish various visual qualities by using tens or even hundreds of hidden layers. The intricacy of the collected visual attributes is increased with each uncovered hidden layer. For example, the first hidden layer may learn how to recognize edges, while the last hidden layer may learn how to identify more sophisticated shapes customized to the geometry of the object we are trying to remember [33].

1.8.3 Multi-Layer Perceptron

One common variation of CNN has several convolution layers like the multi-layer perceptron (M.L.P.), subsampling (Pooling) levels, and F.C. layers. A typical CNN design for labeling images is shown in Figure 7. In a CNN model, the input x of each layer is organized in three dimensions: height, width, and depth, or m, m, r, where the width (m) matches the size (r) and where the depth (r) is greater than the width (m) (m). In addition, the channel number is another name for the depth. The depth, denoted by the letter r, is always equal to three in an RGB image, for instance. In each convolutional layer, there are several kernels, also known as filters, that can be accessed. The letter k denotes these kernels, each having three dimensions, represented by the notation n * n * q. These dimensions are analogous to the picture being fed into the network. In addition, the kernels serve as the basis for the local connections, which have similar qualities (bias bk and weight Wk) when producing k feature maps. hk that have a size of (m minus n minus 1) and are convolved with input, as was previously explained. Like that of the N.L.P., the convolution layer produces a dot product between its input and the weights following Equation 1, but the size of its inputs is less than the size of the original image.

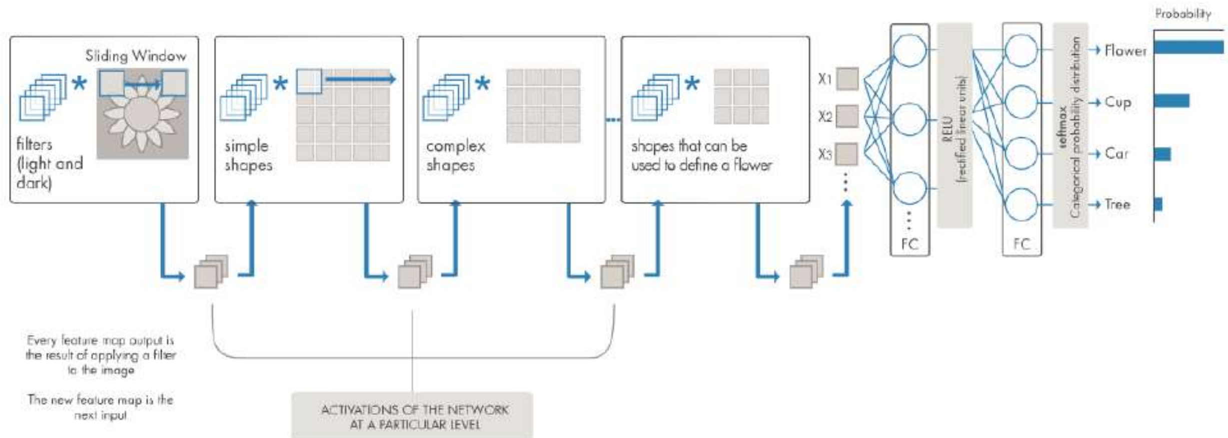


Figure 1.7: Different resolutions of filters are applied to each training picture, and the result of each convolved image serves as the input to the next layer.

The following is what we get when we apply nonlinearity to the output of the convolution layer or an activation function:

$$hk=f\ (Wk * x + bk) \qquad (2)$$

The final step is to conduct a down sample on the feature map of each subsampling layer. This causes a reduction in the parameters of the network, which speeds up the training process and enables the overfitting issue to be handled. For every feature map, the pooling function (such as maximum or average) is implemented on a surrounding area with the dimensions p x p, where p represents the size of the kernel. The F.C. layers are the ones that take in the input at the intermediate and low levels. They then proceed to produce the high-level abstraction, which is analogous to the layers that make up the final stage of a typical neural network. The scores for classification are generated by the last layer, which may be comprised of support vector machines (SVMs), softmax, or both. Each score represents how likely it is that a particular category will occur for a given occurrence.

1.8.4 Benefits of Convolutional Neural Network

When it comes to computer vision applications, using CNNs rather than traditional neural networks has several benefits, including the following:

- The ability to simultaneously learn the feature extraction and classification layers result in a model output that is highly ordered and very dependent on the derived features.
- CNN makes the development of large-scale networks simpler than other types of neural networks.

1.8.5 CNN Layers

The structure of CNN is multi-leveled (or multi-building blocks). The function of each layer of the CNN design is described below in more detail. Convolutional layers, pooling layers, and fully connected (F.C.) layers are the three main layers that make up CNN.

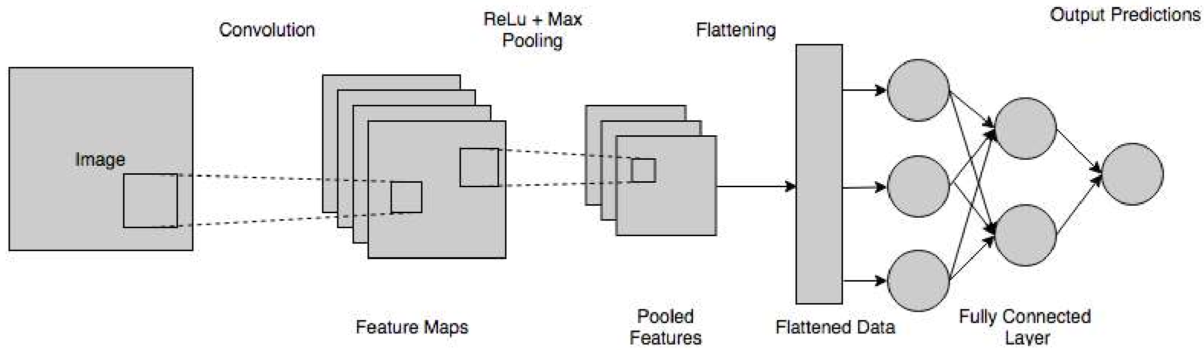


Figure 1.8: Layers of CNN

These layers will be stacked to form a CNN architecture. Two more essential elements, the dropout layer and the activation function, are detailed in greater depth below along with these three layers.

1.8.5.1 Convolutional Layer

This layer is the starting point for extracting specific features of an image. Convolution between the input image and a filter of size MxM is carried out by this layer. The dot product is determined between the filter and the parts of the input image that are in proportion to the filter's size when the user drags it over the image (MxM) [34].

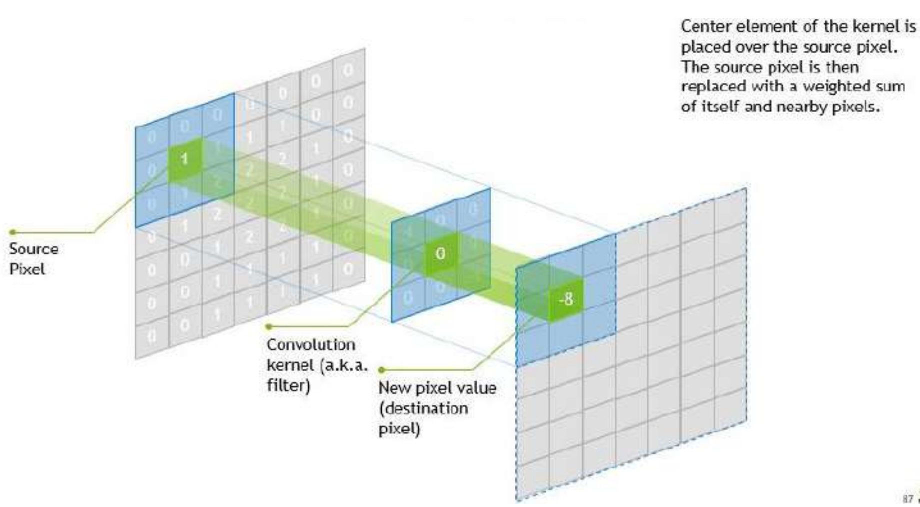


Figure 1.9: Convolutional layer [35]

The result is the Feature map, which provides information about the picture, including its corners and edges. Later, this feature map is supplied to other layers so they may learn more picture features.

After performing the convolution operation on the input, CNN's convolution layer transfers the output to the next layer. Convolutional layers in CNN are very advantageous since they preserve the spatial link between pixels.

1.8.5.2 Pooling Layer

It is common practice to use a Pooling Layer after a Convolutional Layer. The primary goal of this layer is to minimize the size of the convolved feature map to save on computational costs. This is accomplished by isolating each feature map for processing in isolation, hence decreasing the number of interlayer connections. It's important to note that Pooling procedures might vary widely depending on the chosen methodology. Fundamentally, it condenses the features that a convolution layer generates.

The feature map is the primary source for the most significant part of Max Pooling. Average Pooling takes a segment of a specific size from an image and calculates its intermediate component within that segment. Sum Pooling adds together all the individual parts in the specified area. Commonly, the Pooling Layer is used to connect the Convolutional Layer with the F.C. Layer. This CNN paradigm allows networks to independently identify the features by generalizing the properties returned by the convolution layer. It helps reduce network computations as a bonus.

1.8.6 Activation Function

Activation's Purpose (nonlinearity) All activation functions in all N.N. types serve the essential purpose of mapping input to output. The input value is calculated by summing the neuron's input and bias, with the latter factoring in more heavily (if present). It follows that the activation function is responsible for producing the related output, which signals whether or not a neuron fires in response to a certain input. In CNN architecture, non-linear activation layers are applied after all weighted layers (so-called learnable layers, such as F.C. and convolutional layers).

In addition to allowing the CNN to learn complicated new information, the activation layers' non-linear performance also indicates that the mapping from input to output will be non-linear. For error back-propagation to be used in training the network, te activation function must also be able to distinguish. When it comes to CNN's and other types of deep neural networks, the following activation functions are often used:

As a result of the activation layers' non-linear behavior, the mapping from input to output will likewise be non-linear, and these layers provide the CNN's learning ability its complexity. The activation function must also discriminate, an essential feature that permits the utilization of error back-propagation in the training of the network. The most common activation functions in CNNs and other types of deep neural networks are as follows:

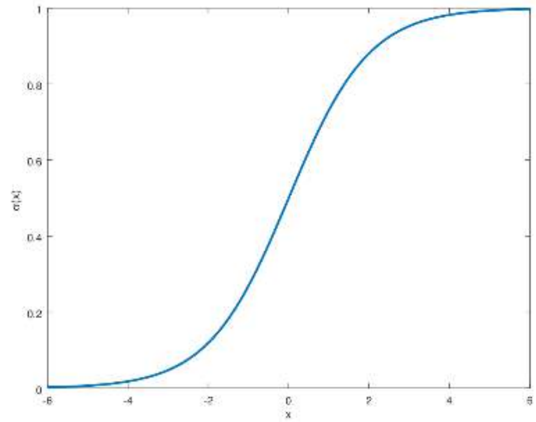


Figure 1.10: Sigmoid Function

The sigmoid function curve is S-shaped and is formally described by the equation.

Tanh: Like the sigmoid function, it takes absolute values as input but can only return a value between 1 and 1.

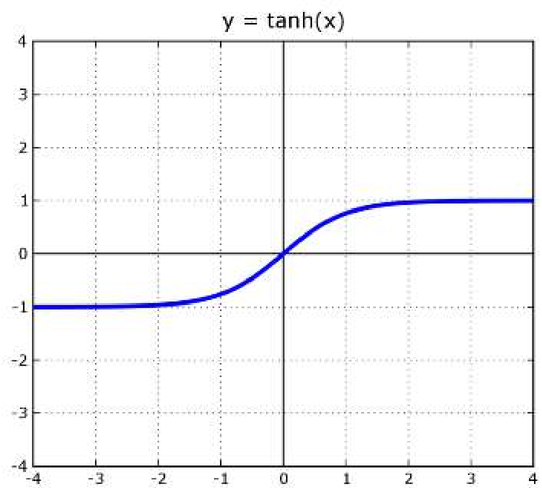


Figure 1.11: Tanh Function

If you're talking about a CNN, the most popular function is ReLU. It modifies the integer values in the input such that they are all positive. The computational cost is much decreased by using ReLU compared to other techniques. Its mathematical expression is given in Equation 4.

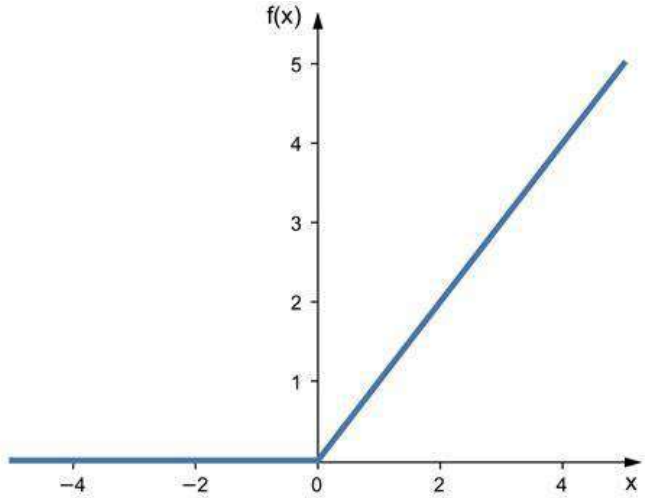


Figure 1.12: ReLU Function

There are times when utilizing ReLU may be pretty challenging. Take, for instance, a back-propagation of errors technique where the gradient is increased. If we feed this gradient into the ReLU function, we may change the weights so that the neuron is no longer active. "Dying ReLU" describes this issue well. There are more methods besides ReLU that can solve these problems. Some examples of these are discussed here.

Instead of downscaling negative inputs as ReLU does, this activation function ensures they're always considered.

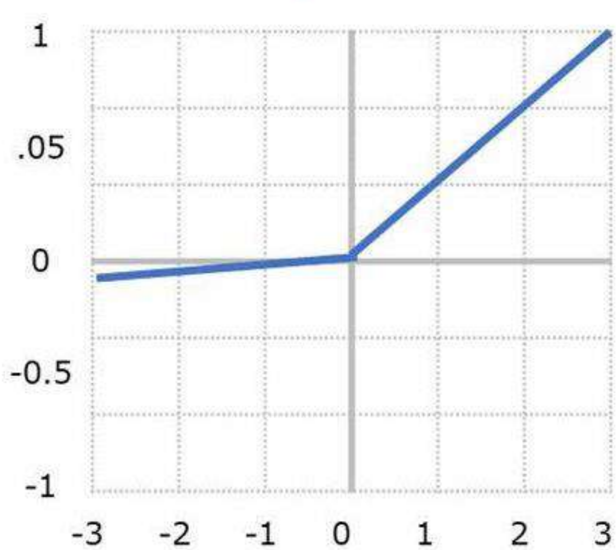


Figure 1.13: Leaky ReLU Function

To deal with the dying ReLU problem, this is implemented. Leaky ReLU is defined explicitly by Equation 5. The leak factor is denoted by the symbol m. It is often set to a minimal value, like 0.001.

1.8.7 Fully Connected Layer

Neurons from two different layers are connected using the Fully Connected (F.C.) layer, which also contains weights and biases. These layers are often the final ones in a CNN Architecture, just before the output layer.

The F.C. layer receives the input image from the preceding layer after it has been flattened in this stage. As the process continues, the vector goes through many more F.C. levels, which is typical for mathematical function operations. The dividing up of data into valuable categories starts here. Two layers are coupled because their combined performance is more outstanding than a single connected layer. The requirement for human oversight is reduced because of these CNN levels.

Loss Function: Various CNN layer-types were shown before. In addition, classification is completed by the CNN's output layer, the very last layer in the architecture. The expected error based on the training data is computed by applying several loss functions to the CNN model's output layer. This slip-up highlights the discrepancy between actual and predicted production. The next step is to fine-tune it using CNN's educing process. There are two factors in the loss function that influence the error. The first input is the predicted or anticipated CNN output. In this case, the output label is the second parameter. Different problem categories call for other loss function implementations. The many forms of loss functions are briefly discussed here.

CNN performance is often measured using the Cross-Entropy or Softmax Loss Function. Another name for it is the log loss function. You'll get back the likelihood that p is between 0 and 1. In addition, the square error loss function is often replaced by it in multiclass classification problems. Softmax activations are utilized to generate probabilistically distributed output in the final "output" layer. The probability of a given output class is expressed mathematically in Equation 8.

N represents the number of neurons in the output layer, and eai represents the non-normalized output from the layer preceding it. Cross-entropy loss is expressed mathematically in Eq. 3.

$$H(p,y) = -\sum_{i} y_i \log(p_i) \quad where \quad i \in [1,N] \qquad (3)$$

In regression, the Euclidean Loss Function is often used. Mean square error is also provided. The calculated Euclidean loss is represented by Equation 4.

$$H(p,y) = \frac{1}{2N}\sum_{i=1}^{N}(p_i - y_i)^2 \qquad (4)$$

The Hinge Loss Function is often employed in problems requiring a yes/no answer. In the context of maximum-margin-based classification, this problem is significant for support vector machines (SVMs) that use the hinge loss function, in which the optimizer attempts to maximize the margin around dual goal classes. The mathematical formula is the number 5.

$$H(p,y) = \sum_{i=1}^{N} max(0, m - (2y_i - 1)p_i) \qquad (5)$$

Commonly, margin m is set to 1 In addition, the expected output is represented by pi, while the intended outcome is represented by yi.

1.8.8 Dropout

The training dataset may be overfitted if all features are connected to the F.C. layer. When a model does very well on the training data but suffers when used on new data, this is known as overfitting.

This problem may be solved by using a dropout layer, which prunes the size of the neural network during training by eliminating some connections between the network's nodes. A dropout threshold of 0.3 causes the random elimination of 30% of neural network nodes. Dropout improves a machine learning model's performance because it reduces the likelihood of the model being overfitting by making the network more simplistic. In order to enhance, neural networks must shed neurons while being trained [36].

1.8.9 Regularization

Regarding CNN models, the key to good generalization behavior is over-fitting. We will go into more depth about what it means for a model to be over-fit in the next section, but suffice it to say that it is over-fit when it does well on training data but poorly on test data (new data). When a model doesn't learn enough from its training data, it's said to be under-fitted.

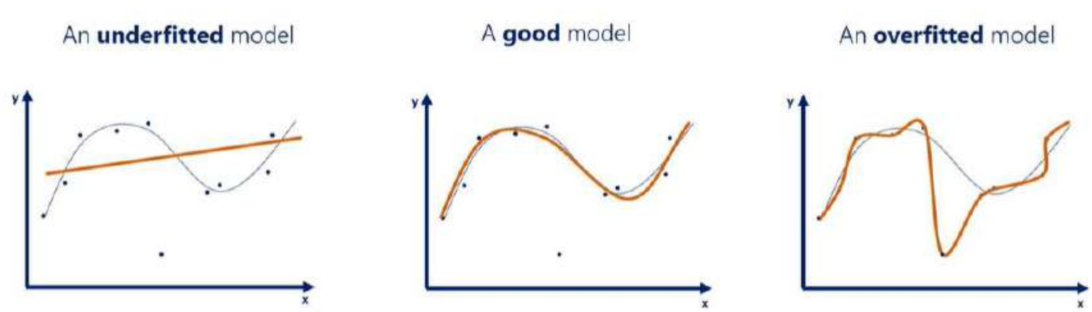


Figure 1.14: Understanding model via fit curve [37].

A model is said to be "just-fitted" if it produces satisfactory results on both the training and testing data. See Figure 11 for an illustration of these three types. The regularization is helped along by several intuitive ideas that help it avoid over-fitting; both over-fitting and under-fitting are detailed in further detail in the following sections.

Dropout: This method of generalization is often used. Neurons are lost at random throughout each training cycle. By doing so, the model is coerced into learning a wide variety of feature types that are not interdependent on one another. During training, the removed neuron will not participate in either back- or forward-propagation. Predictions are generated using the full-scale network throughout the testing process.

Drop-Weights: To a large extent, this method mimics dropping out. The only difference between drop-weights and neuronal dropout is that the weights (connections between neurons) are removed in each training session rather than the neurons themselves.

Data Augmentation: Training the model on a large amount of data is the most straightforward way to avoid over-fitting. As a result, data augmentation is used. The training dataset is artificially inflated utilizing several different methods. The last section offers a more in-depth breakdown of the various data augmentation methods [38].

Batch Normalization: This output has the shape of a normal distribution with a mean of one. Normalization is achieved by removing the mean and dividing by the standard deviation from the output at each layer. It is possible to separate this network from others and to integrate it with others; thus, it may be seen as a preprocessing activity at each network layer. It also helps reduce the activation layers' "internal covariance shift." Depending on how the activation matrix is modified, a given layer's internal covariance will alter. Frequently updating weights during training, which may occur if training data samples are acquired from a wide range of sources, magnifies this change to a significant degree (for example, day and night images). Therefore, training time will increase since the model needs more convergence time. With this issue in mind, the CNN architecture incorporates a layer that stands in for the batch normalization method.

The following are the benefits of adopting batch normalization:

- It avoids the occurrence of the issue of disappearing gradient.
- It can effectively regulate the inadequate beginning of weight.
- It significantly decreases the time necessary for network convergence (which is highly important for large-scale datasets).
- It has difficulty reducing training dependence across hyper-parameters.
- Due to its little effect on regularization, the likelihood of over-fitting is minimized.

1.8.10 Optimizer selection

This section describes CNN's learning process. The learning process includes two important issues: the selection of the learning algorithm (optimizer) and the usage of several upgrades (such as AdaDelta, Adagrad, and momentum) in conjunction with the learning method to improve the output.

Loss functions, which are based on various learnable parameters (e.g., biases, weights, etc.), or error minimization (difference between actual and anticipated output) are the fundamental goal of all supervised learning algorithms. The selection of gradient-based learning strategies for a CNN network is the norm. To reduce error, the network parameters should constantly be updated during all training epochs, and the network should also search for the locally optimum solution throughout all training epochs.

The learning rate is defined as the parameter update step size. The training epoch comprises a single iteration of the parameter update using the whole training dataset. Note that, although a hyper-parameter, it must choose the learning rate with care so that it does not inaccurately influence the learning process.

Gradient Descent or Gradient-based learning: This approach repeatedly modifies the network parameters throughout each training session to reduce the training error. To precisely update the parameters, the objective function gradient (slope) must be computed using the first-order derivative concerning the network parameters. To decrease inaccuracy, the parameter is then modified in the opposite direction of the gradient. Using network back-propagation, the slope at each neuron is back-propagated to all neurons in the previous layer during the parameter update process. Equation 6 is the mathematical expression of this procedure.

$$w_{ij^t} = w_{ij^{t-1}} - \Delta w_{ij^t}, \quad \Delta w_{ij^t} = \eta * \frac{\partial E}{\partial w_{ij}} \qquad (6)$$

The final weight in the current training period is marked by wijt, whereas the weight in the training epoch before the current one is denoted by wijt1. The learning rate is, and the prediction error is E. Options to the gradient-based learning algorithm are available and often used; these alternatives include:

Batch Gradient Descent: During the execution of this method [39], the network parameters are only modified once after considering all training datasets used by the network. It determines the gradient of the whole training set and then employs this gradient to update the parameters. Using B.G.D., the CNN model converges quicker and generates a more stable gradient for small datasets. As parameters are altered just once every training period, a large number of resources are required. In contrast, a big training dataset requires longer time for convergence and may converge to a local maximum (for nonconvex instances).

Stochastic Gradient Descent: In this approach, the parameters are changed for each training sample [40]. Before training, it is desirable to randomly sample the training samples in each epoch. Compared to B.G.D., this approach is more memory-efficient and much quicker for large training datasets. However, since it is often updated, it makes very noisy steps in the direction of the solution, resulting in a volatile convergence behavior.

Mini-batch Gradient Descent: In this method, the training data is split up into a series of smaller batches, or "mini-batches," that may each be thought of as a unique, non-overlapping set of samples [41, 42]

Parameter updates are applied to each mini-batch when gradient calculation is complete. As a result of incorporating features of both B.G.D. and S.G.D. techniques, this approach is practical. Tus has enhanced memory efficiency, steady convergence, and high computational efficiency. Here we detail various methods for improving CNN training using gradient-based learning algorithms (often S.G.D.).

Momentum: A neural network's aim function makes use of this tactic. It enhances the Accuracy and the training speed by adding the estimated gradient at the previous training step, weighted by an amount (known as the momentum factor). This causes it to get stuck at a local minimum rather than a global one. For the most part, this is the case with gradient-based learning methods. These often arise when there is no apparent convex surface to the issue (or solution space).

Adaptive Moment Estimation (Adam): It's a standard method for improving optimization results or training new skills. It's a traditional method for improving optimization results or introducing new skills. It's a standard method for improving optimization results or preparing new skills. It's a standard method for enhancing optimization results or training new skills. It's a common method for enhancing optimization results or introducing new skills for Adam [43] represents cutting-edge of deep learning optimization techniques. This is symbolized by the Hessian matrix, which uses a second-order derivative. Adam is a custom-built learning strategy for educating D.N.N.s. Adam's memory efficiency is higher, but his processing speed is lower. Adam's method involves calculating adaptive L.R. for all of the model's variables. It's like getting the best of Momentum and RMSprop in one convenient package. It is similar to momentum in that it takes the moving average of the gradient to determine the scaling factor for the learning rate (RMSprop). Specifically, Eq. 7 is the representation of Adam's equation.

$$w_{ij^t} = w_{ij^{t-1}} - \frac{\eta}{\sqrt{\widehat{E[\delta^2]^t}} + \epsilon} * \widehat{E[\delta^2]^t} \quad (7)$$

1.8.11 Convolutional Neural Network Architecture

The last ten years have introduced many CNN architectures [44, 49]. Improving the efficiency of many programs requires careful consideration of their model design. Many changes have been made to CNN's structure since its inception in 1989. Examples of these adjustments include parameter optimizations, regularization, and structural reformulation. However, it's important to note that the significant gain in CNN performance was mainly due to the reconfiguration of processing units and adding additional blocks. The use of network depth has been at the forefront of cutting-edge developments in CNN layouts. Starting with the AlexNet model and ending with the High-resolution (H.R.) model, this section looks at the most popular CNN designs from 2012 to 2020. The key to aiding researchers in picking the ideal design for their particular study is to examine a wide range of architectural parameters (such as input size, depth, and robustness). The second table summarizes several CNN designs.

Table 1.1: A summary of CNN architectures [50]

Model Main	Finding	Depth	Dataset	Error rate	Input size
AlexNet	Includes Dropout and ReLU	8	ImageNet	16.4	227 × 227 × 3
NIN	The latest layer, called "mlpconv," uses the G.A.P.	3	CIFAR-10, CIFAR100, MNIST	10.41, 35.68, 0.45	32 × 32 × 3
ZfNet	Conceptualization of intermediate levels	8	ImageNet	11.7	224 × 224 × 3
V.G.G.	Increased depth, small filter size	16, 19	ImageNet	7.3	224 × 224 × 3
GoogLeNet	uses a tiny filter size and has improved feature representation	22	ImageNet	6.7	224 × 224 × 3
Inception-V3	uses a refined filter size for a more accurate depiction of features	48	ImageNet	3.5	229 × 229 × 3
Highway	introduced the idea of several paths	19, 32	CIFAR-10	7.76	32 × 32 × 3
Inception-V4	Divided transform and integration concepts	70	ImageNet	3.08	229 × 229 × 3
ResNet	Overfitting is prevented via symmetry mapping-based skip linkages, making them robust.	152	ImageNet	3.57	229 × 229 × 3
Inception-ResNetv2	Residual linkages were first introduced as a new idea.	164	ImageNet	3.52	229 × 229 × 3
Xception	First, a convolution in-depth, and then a convolution in points	71	ImageNet	0.055	229 × 229 × 3
Residual attention neural network	Layered blocks; interconnected layers	452	CIFAR-10, CIFAR100	3.90, 20.4	40 × 40 × 3
DenseNet	Multiple layers stacked on top of one another	201	CIFAR-10, CIFAR100, ImageNet	3.46, 17.18, 5.54	224 × 224 × 3
Competitive squeeze and excitation net-work	The channel resizing was accomplished using a combination of residual and identity mappings.	152	CIFAR-10 CIFAR-100	3.58 18.47	32 × 32 × 3
MobileNet-v2	Inverted residual structure	53	ImageNet	-	224 × 224 × 3
CapsuleNet	Pays attention to particular relationships between features	3	MNIST	0.008555	28 × 28 × 1
HRNetV2	High-Resolution representation	-	ImageNet	5.4	224 × 224 × 3
FractalNet	Presented Drop-Path, a new regularization method	40, 80	CIFAR-10 CIFAR-100	4.60 18.85	32 × 32 × 3
WideResNet	Reduced the height and widened the base.	28	CIFAR-10 CIFAR-100	3.89 18.85	32 × 32 × 3
Squeeze-and-	Modeledinter- dependencies between channels	-	ImageNet	2.25	229 × 229 × 3

excitation networks					224 × 224 × 3 320 × 320 × 3

Chapter 2
Literature Review

2.1 Literature Review

The primary focus of our thesis work in this project is on developing A Deep CNN Biomedical Imaging Technique for Detecting Infected Covid Patients who are infected. Chest x-ray pictures are the primary tool in our work to determine whether or not a patient is infected with COVID.

The fundamental goal of this investigation is to plan and investigate the problem at hand to arrive at a suitable solution for humans with the aid of A.I. Deep learning techniques might help solve the problem if implemented correctly. As a subset of artificial neural networks, convolutional neural networks are often used in deep learning to analyze visual input. Convolutional neural networks (CNNs), also known as "Shift Invariant" or "Space Invariant" Artificial Neural Networks, are named for the shared-weight architecture of convolution kernels or filters that slide along input features and provide translation-equivariant responses, also known as feature maps. CNN may also be called "Shift Invariant" or "Space Invariant" ANNs. Most convolutional neural networks are not translation-invariant since they down-sample the input. This contradicts the common-sense view that these kinds of networks would be.

Given that our problem is to detect causality from an image dataset, CNN is one of the best techniques to provide a high-accuracy forecast with variable outcomes. In many cases, the quality of the supplied picture datasets prevents the CNN model from performing as expected, leading to an inaccurate prediction from the resultant model. Since the CNN model is a convolutional neural network, this is true. This is a severe challenge for deep learning. The proposed CNN paradigm gets us closer to our aims in this area.

Within the realm of artificial intelligence, our objective was to accurately anticipate as much as possible using our model while maintaining a high level of precision of 100%, which is the primary emphasis of our investigation. This test technique is more accurate and exact than many others, and it can detect COVID infections in the lungs when they are still in the early stages of the disease.

2.2 Methodology

A research methodology is a systematic strategy for addressing the research problem. One way to look at it is as an inquiry into the practice of doing scientific research. There, we dissect the researcher's typical workflow and the thought processes that go into each step. The researcher's knowledge of research methodology is just as necessary as research methods and procedures. It is not enough for a researcher to know how to create an index or test, how to calculate the mean, mode, median, or standard deviation, or how to apply a particular research technique; they must also understand which methods and techniques are relevant and which are not, what they mean and indicate, and why. Choose one example result, for instance.

1. Paper Selection: Paper selection is like reading a large number of research papers that I like; the task, as I understand it, is to choose several articles from them. The capacity to create a compelling proposal topic is a crucial talent. For anything to be engaging, it must be focused and confined yet big enough to uncover various relevant information. "A Deep CNN Biomedical Imaging Technique for Detecting Covid Infected Patients" is the subject of my thesis. It seeks to develop the most general axioms or what might be considered scientific hypotheses. Choosing this subject before, our primary emphasis is on the points are -

1. Choose a topic that resembles my objective
2. Is idea generation equivalent to the opinion?
3. This subject will facilitate comprehension of the text.
4. Is it adaptable to work?
5. What does it mean to construct a research statement?

2. Classifier selection: Among many papers, we chose a particular classifier out of many others, which is the Convolutional Neural Network (CNN) and its other architectures. because this classifier had better classification prediction than the other classifier.

To fully grasp the model's efficacy, we look at its train accuracy, test accuracy, validation loss, validation loss, precision, recall, f-1 score, support, and confusion matrix. I was making a list in response to this discovery. To determine whether these findings are more precise and superior in any other respect, I want to compare them to those of other researchers.

2.3 Paper review

The purpose of this part was to provide the reader a sense of the state of the art in the field by providing an overview of some relevant works. One of the most powerful DL models, convolutional neural networks (CNNs) have shown their superiority over traditional approaches in various fields, including as image classification and pattern recognition [51, 52].

1. Ali et al. [53] constructed a deep convolutional neural network (CNN) for the classification of COVID-19 Chest X-ray pictures into normal and COVID-19 classes, using the ResNet50, InceptionV3, and Inception-ResNetV2 models. Results from C.T. scans were shown to correlate well with those from the PCR method, they said. Fifty COVID-19 patients' chest X-rays were downloaded from a public repository on GitHub that was made available by (Dr. Joseph Cohen [54].

2. Prabira et al. [55] have developed a deep feature and support vector machine-based approach for detecting COVID-19 in X-ray pictures (SVM). GitHub, Kaggle, and the Open-I repository were scoured for X-ray pictures. Despite just using a tiny sample size of pictures, they were able to draw a conclusion on the relative performance of many CNN models by extracting their deep feature maps.

3. Maghdid et al. [56] proposed a simple CNN of 16 layers only to detect COVID-19 using both X-ray and C.T. scans and reported good performance but the dataset used is small.

4. The work of Fei et al. [57] focused on segmenting COVID-19 CT scans using a deep learning approach known as VB-Net and reported dice similarity of 91%±10%.

5. Xiaowei et al. [58], by feeding the Resnet18 model image patches centered on areas of interest, we were able to construct an early prediction model for classifying COVID-19 pneumonia from Influenza-A viral pneumonia and healthy patients using pulmonary C.T. images. The CNN model achieved a maximum accuracy of 86.7% for C.T. pictures.

6. In Shuai et al. [59] by putting image patches centered on areas of interest into the Resnet18 model, we could generate an early prediction model for distinguishing COVID-19 from Influenza-A viral pneumonia and healthy patients in pulmonary C.T. pictures. For C.T. pictures, the CNN model achieved a maximum accuracy of 86.7%.

7. In [60] X-ray pictures from patients with coronavirus, pneumonia, and healthy controls were utilized to test many different CNN architectures, all of which are used for various medical image classifications. A total of 224 COVID-19 photos, 700 non-COPD19 pneumonia images, and 504 standard images were used to train the CNNs, which achieved an overall accuracy of 97.82 percent.

8. Wang and Wong [61] researchers looked at a dataset called COVIDx and a neural network architecture called COVID-Net, both of which were developed to identify instances of COVID- 19 in open-source chest X-ray pictures. Four types of chest radiography pictures are included in the dataset: Normal X-rays, which represent instances with no infections; Bacterial X-rays; Viral X-rays, which represent pneumonia caused by viruses other than COVID-19; and COVID-19 X-rays. They reported an overall accuracy of 83.5 percent for these four courses. According to the data, the non-COVID-19 class had the lowest positive predictive value (67.0%), while the Normal type had the most significant (90.0%) (95.1 percent).

9. As required to improve the previous studies Muhammad and Hafeez [62] meets this need by introducing a second CNN that requires less parameters but performs better. These authors utilized the same dataset as in [63] to create a COVID-ResNet that is both open-source and more effective than COVID-Net in differentiating between COVID-19 and the other four pneumonia cases.

10. While Zhang et al. [64] used the ResNet-18 model as a feature vector to extract helpful feature representations from a C.X.R. picture. Those extracted characteristics were fed into a multi-layer perception system. Most accurately (96%), a dataset of one hundred pictures from seventy patients was generated.

11. In paper [65] have suggested constructing a new deep architecture named COVID-Net using an automated method that analyzes C.X.R. pictures for signs of COVID-19. This model is able to correctly categorize 13,975 CXR images from a database with an accuracy

of 93.3%. One of the method's main selling features is that, with some careful thought, you may find a happy medium between, say, Accuracy and computational cost.

12. Hemdan et al. [66] established the COVIDXNet DL framework for identifying C.X.R. photos infected with the COVID-19 virus. Seven DL methods were evaluated using just a 50-image dataset (e.g., MobileNetV2, ResNetV2, VGG19, DenseNet201, InceptionV3, Inception, and Xception). The best results were achieved by DenseNet201, which had an accuracy rate of 91%.

13. Narin et al. [67] implemented rigorous binary classifications using five-fold cross-validation. The best performance is achieved using the pre-trained ResNet-50 approach, which achieves an accuracy of 98%, a specificity of 100%, and a recall of 96%.

14. Reference [68] optimized seven convolutional neural networks (CNNs) for COVID detection; they were InceptionV3, ResNet50V2, Xception, DenseNet121, MobileNetV2, EfficientNet-B0, and EfficientNetV2.

17. Additionally [69] built a CNN model with minimal power requirements for embedded use.

Chapter 3
Methodology

3.1 Introduction

The term "method" refers to a certain kind of "special processes or approaches." Its primary function is to recognize, select, process, and evaluate information about a subject area. When writing a research report, a user may also use the word "methodology" to assist the reader in critically assessing a study's overall validity and dependability. One of the subfields of methodology is known as qualitative methodology, while the other is known as quantitative methodology. A qualitative technique may be employed to comprehend people's perspectives about an event that has transpired or a candidate running for president. For example, suppose we may state like provided example like to grasp the viewpoints of people, in contrast to this. In that case, a quantitative technique is often used where the aims and targets of the study are confirmatory.

In the next section, we will employ deep convolutional neural networks (CNNs) and transfer learning to develop a system capable of analyzing chest X-ray pictures for the presence of COVID. In the first part of the procedure, contrast-constrained adaptive histogram equalization was used to improve the quality of the dataset (C.L.A.H.E.) [81] and modifying the brightness through the deployment of the white balance approach (W.B.). Because CNNs are known to have excellent performance in identifying anomalies in medical imaging, our team decided to create a custom CNN on top of our model. After that, we employed the transfer learning method to compare it to the model we had presented.

3.2 Dataset

The illness known as COVID-19 affects every single person on the planet, and a correct and prompt diagnosis of the condition is urgently required. Therefore, to construct an effective AI-based diagnostic system, we gathered chest x-ray pictures from a variety of sources so that we could train our CNN model to automate the whole diagnosis process. This was done so that we could save time. As a result, we have collected photos of chest rays from various sources, including published studies, and integrated these individual images into a single complete dataset that the scientific community can use. This dataset includes 1823 pictures of a chest X-ray taken from Kaggle, the most extensive dataset repository, is the source for these pictures. Annotated images of 1823 chest X-rays obtained from the posteroanterior (P.A.) angle are included in this collection. Optical coherence tomography (OCT) and computed radiography (C.R.) pictures with labels are used to diagnose viral pneumonia and other conditions. In addition to the 536 COVID-19 photographs and the 619 viral pneumonia images, the collection also contains 668 shots of healthy individuals. Patients with COVID-19 involved in the study were between 18 and 75.

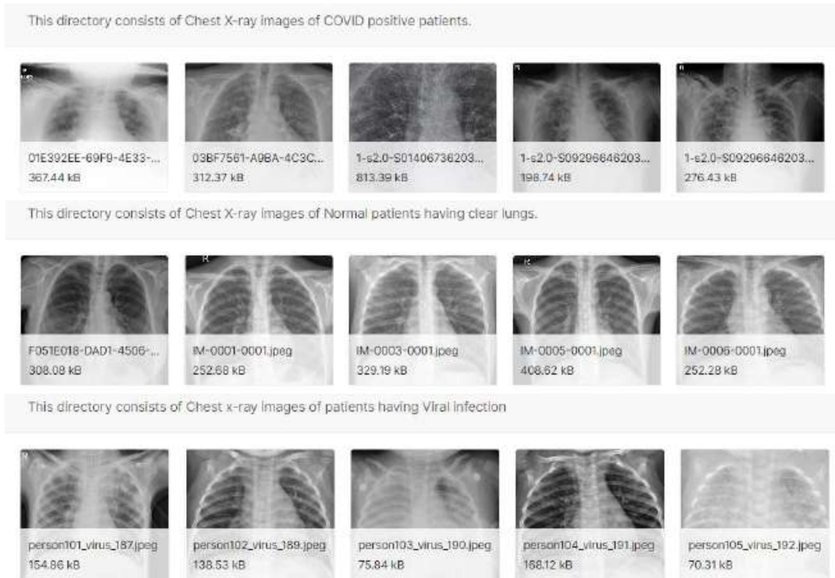


Figure 3.1: From left to right, several example chests x-rays from the dataset we utilized for this study: Covid and Non Covid.

3.3 System Setup

The preparation of data, the running of experiments, and the model analysis are all done using the programming language Python. TensorFlow and Keras are used in order to bring the suggested architecture to life. In addition, NumPy is used for mathematical operations on the architecture, while OpenCV is used for image processing. Both of these programs are utilized by the architecture. In addition, we used the matplotlib library to create visually appealing graphs. Python 3.8 is used for this work

3.4 Data Pre-processing

The overall quality of each of the photographs continues to become better. This study utilizes two approaches to increase the overall picture quality and standardize the data.

3.4.1 Image Enhancement process:

In order to increase the visual perception quality of medical pictures for the purpose of sickness diagnosis, image enhancement is a must. If the image's histogram is equalized, the intensity is distributed uniformly across all of the pixels, which might result in a more pleasing final product. In certain cases, the information conveyed by the white pixels might be lost due to the high contrast that occurs in the white region [82].

3.4.1.1 Image standardization:

The goal of this method is to limit the amount of variance (light and brightness) introduced into the photographs throughout the process of data acquisition by standardizing all of the images included in the data. This study employs a method of image modification known as a template image (*T*) [83[22]], in which the appearance and feel of the pictures included within the dataset are altered to match the appearance and feel of a previously defined image. As a result, the standardized image (I_s) may be acquired through the following methods:

$$I_s = \left(\frac{\sigma_T}{\sigma_I} \times m\right) + \mu_t$$

where,

$$m = I - \mu_I$$

The average intensity is μ_I and μ_T of pixel values in an input picture and in the template, image is denoted by 'σ_I' and 'σ_T' respectively. An input image's variance intensity of pixel values is denoted by I, while the template image's variance intensity of pixel values is denoted by T. Figure 3 is an illustration of one of the standardized pictures that has been created.

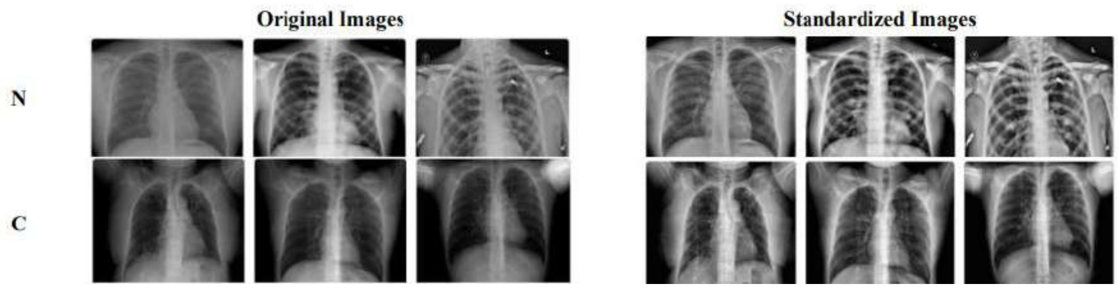


Figure 3.2: The two columns on the left represent the original picture, while the two columns on the right represent the standardized images for the image classes (N= noncovid and C= Covid)

3.4.1.2 White Balance:

It's the image processing procedure that is conducted to a digital photograph in order to correct the colors in the photo so that they are accurate. Some of the pictures seemed black because the lighting conditions in the medical imaging were poor, and the technology used to capture the image does not detect light as accurately as our visual system does. Picture processing and correction are two methods that may assist in guaranteeing that the colors in the finished image accurately reflect those in the original natural image. This procedure aims to enhance the image's contrast to facilitate meaningful information extraction using deep convolutional neural networks (DCNNs). The white balance algorithm will tweak the tones of the five selected colors. The conventional convolution operation (left) and the depth-separable convolution operation are shown in Figure 1. (right). The recommended neural network solution (COVIDLite), which independently expands the red, green, and blue channels, is seen as being trained in Figure 2. Fewer than 0.05 percent of the picture's total pixels utilize the colors at the end of the three channels, thus, those colors are discarded. Meanwhile, the spectrum of colors that remains is distorted. By adhering to this approach, rarely-seen pixel colors six won't be able to negatively impact the higher and lower limit value during channel. To achieve this, a white-balancing technique was implemented in Python with the help of the NumPy and OpenCV libraries [84, 85].

The steps of the White balance algorithm can be summarized as:

$$\boldsymbol{C_b} = \boldsymbol{k_{0.05}}(\boldsymbol{C}) \quad (1)$$

$$\boldsymbol{F_{upd}} = \boldsymbol{Clip}\left(\frac{(\boldsymbol{C}-\boldsymbol{C_b}) \times \mathbf{255}}{\boldsymbol{C_a}-\boldsymbol{C_b}}, \mathbf{0}, \mathbf{255}\right) \quad (2)$$

$$\boldsymbol{C_a} = \boldsymbol{k_{100-0.05}}(\boldsymbol{C}) \quad (3)$$

Where the expression Pi ($\boldsymbol{C}$) stands for the ith percentile being taken from channel $\boldsymbol{C}$ and the $\boldsymbol{Clip}$ (., min, max) operation illustrates conducting a saturation operation within the range of min and max values. $\boldsymbol{C}$ and $\boldsymbol{F_{upd}}$ Signify, respectively, the input channel pixel values as well as the updated channel pixel values following the operation.

3.4.1.3 Contrast-limited Adaptive Histogram Equalization

This process aims to enhance the image's contrast so that deep convolutional neural networks (DCNNs) can more effectively extract relevant data from it. A white balance algorithm will be applied to the currently selected five colors. Figure 1 is a side-by-side comparison of a standard convolution (left) and a depth-separable convolution (right) (right). The recommended neural network solution (COVIDLite) in Figure 2 undergoes training to independently extend the red, green, and blue channels. A small percentage of the picture's pixels (less than 0.05%) are not using the colors found at the end of the three tracks. Thus, those colors are discarded. Additionally, the existing color palette is distorted. This method eliminates the negative impact on the higher and lower limit value during channel stretching caused by pixel colors six that are seldom encountered near the channel's conclusion. [84]. As for the technical details, this solution was programmed in Python with the help of the NumPy and OpenCV libraries to construct a white balancing technique [85].

The A.H.E. algorithm segments the input image into a set of smaller images called tiles. The intensity remapping function may be calculated by generating a histogram for each tile. Separate histograms represent different parts of the image. The extreme amplification used in this method causes visual noise. C.L.A.H.E. is functionally equivalent to A.H.E., with the addition that it clips the histogram at certain levels to limit the amplification before computing the cumulative distributive function [86].

The overamplified portion of the histogram is redistributed differently over the histogram. In one of the earlier researches, C.L.A.H.E. had excellent results in improving chest C.T. pictures, and it was regarded as beneficial in analyzing a broad range of different medical imaging [86]. The following is how the calculation of C.L.A.H.E. is carried out [87]:

$$p = (p_{max} - p_{min}) * P(f) + p_{min}$$

where p represents the pixel value after the C.L.A.H.E. algorithm has been applied, p_{max} and p_{min} represent an image's maximum and lowest pixel values, respectively, and P(f) represents the cumulative probability distribution function.

Figure 3.3: The two columns on the left represent the original picture, while the two columns on the right represent the contrast-enhanced versions of the image classes.

3.4.2 Data augmentation

A key benefit of deep learning is that the learning accuracy rises with the quantity of the dataset. We used several data pretreatment and augmentation techniques to generate a new sample from the current ones and enhance the dataset's quality, hence reducing the likelihood of overfitting and underfitting.

Table 1 lists the parameters used in the image enhancement process, and Figure 4 shows the final result of the data enhancement.

Table 3.1. The image augmentation parameter

Method	Settings	Description
Rescale	1/255	Image reduction during the augmentation process
Zoom range	0.05	Sample a section from the original image. then resize this section to the original image size
Rotation range	25	Randomly rotate the image during training in 25 degrees
Width shift range	0.05	The horizontal translation of the images by 0.05%
Height shift range	0.05	The vertical translation of the images by 0.05%
Shear range	0.05	Clips the image angles in a counterclockwise direction
Horizontal flip	True	Flip the image horizontally

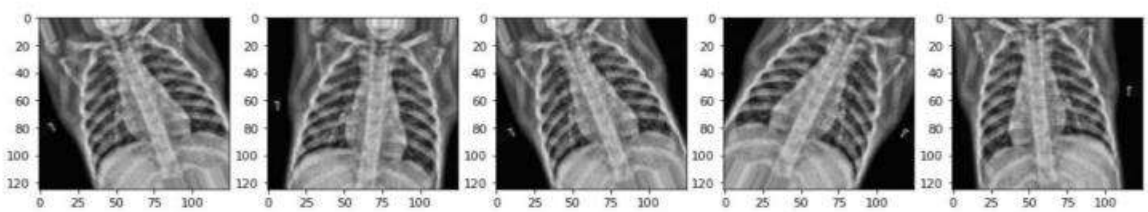


Figure 3.4: The result of data augmentation

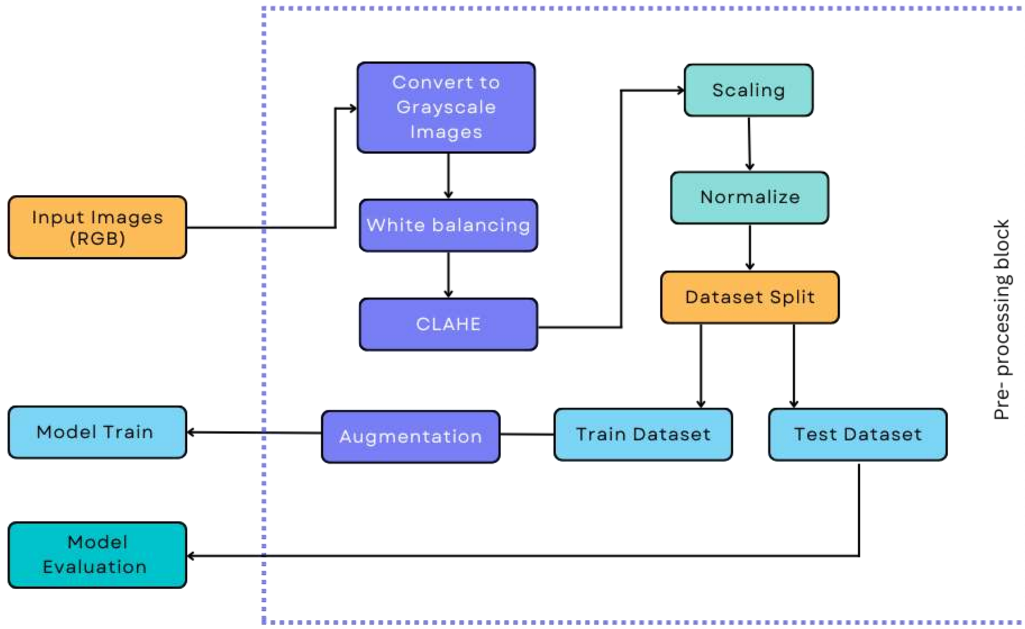


Figure 3.5: Preprocessing technique for our work in the diagram

3.5 Deep custom CNN model for work

CNN's are deep neural networks (D.N.N.) used for image recognition [44]. To construct a CNN model, inputs must be converted into a CNN-compatible format. Images are often regarded as matrices. During training, the model attempts to comprehend the distinctions between the categories, enabling it to predict the labels of unknown pictures. CNN utilizes three layers to perform its purpose effectively: convolutional, Pooling, and fully linked (F.C.). The convolutional layer with Pooling captures the most distinguishable features, while the fully connected layers do the classification operation.

Now, it is just necessary to adapt the final classification layer to one suitable for the picture classification problem. In the present study, a custom deep CNN model is developed.

In addition, we devised a transfer learning strategy utilizing the ImageNet dataset to overcome limited data and training time limits. Figure 2 is a schematic illustration of a traditional CNN consisting of pretrained vgg16, Densenet, and Inceptionv3 models for predicting normal (healthy) and COVID-19 classes from chest X-rays.

We initialize the parameters for our CNN with a batch size of 16 heights and a width of 224, so the input layer has 224*224*3 neurons, with 12 intermediate hidden layers, 4 Convolution2D layers (32, 64, 64,128), 3 Max_Pooling2D layers (64, 64, 128), 3 Dropout layer (64, 64,128), 1 Flatten layer (86528), 2 Dense layers (64, 1), and two output neurons for the binary classification as shown in Figure 2. Our model was trained between 1 and 300 epochs, and all the available data is passed through the neural network. We used Relu as the activation function, while the optimizer function was 'Adam', and we used Accuracy as metrics. We train over 4 million parameters.

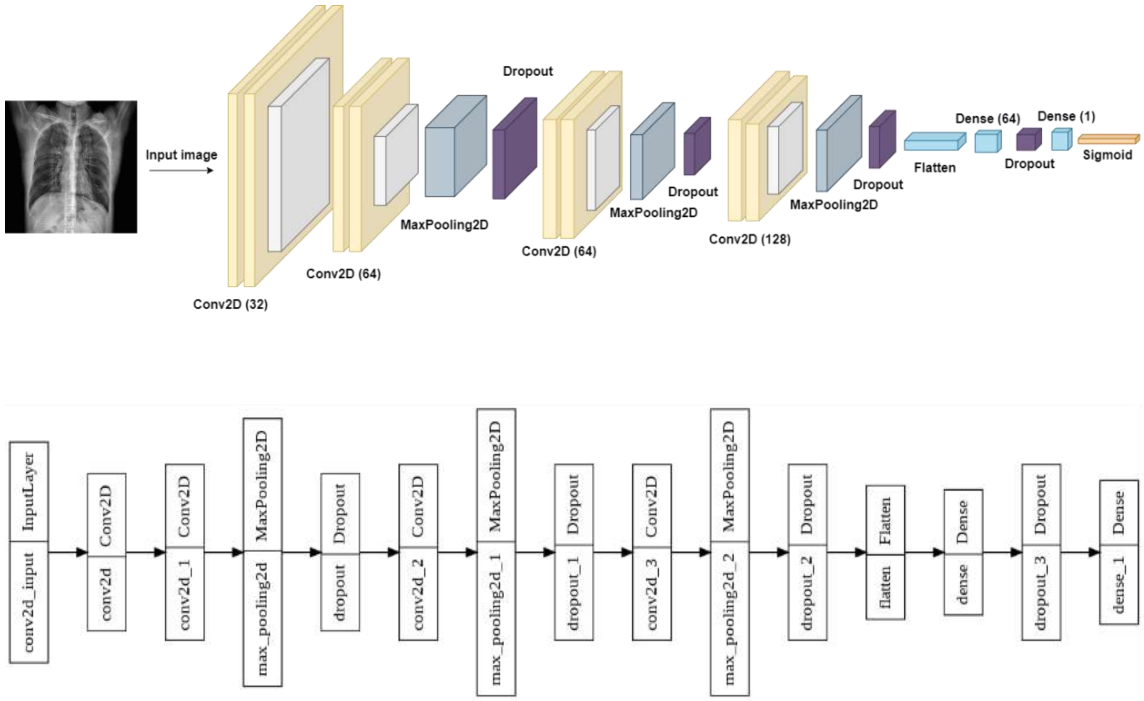


Figure 3.6: Proposed custom Convolutional Neural Network model layers

Table 3.2: Summary of the proposed model

Layer	**Output shape**	**Parameter**
conv2d (Conv2D)	(None, 222, 222, 32)	896
conv2d_1 (Conv2D)	(None, 220, 220, 64)	18496
max_pooling2d (MaxPooling2D)	(None, 110, 110, 64)	0
dropout (Dropout)	(None, 110, 110, 64)	0
conv2d_2 (Conv2D)	(None, 108, 108, 64)	36928
max_pooling2d_1 (MaxPooling2D)	(None, 54, 54, 64)	0
dropout_1 (Dropout)	(None, 54, 54, 64)	0
conv2d_3 (Conv2D)	(None, 52, 52, 128)	73856
max_pooling2d_2 (MaxPooling2D)	(None, 26, 26, 128)	0
dropout_2 (Dropout)	(None, 26, 26, 128)	0
flatten (Flatten)	(None, 86528)	0
dense (Dense)	(None, 64)	5537856
dropout_3 (Dropout)	(None, 64)	0
dense_1 (Dense)	(None, 1)	65

Chapter 4
Result analysis

4.1 Overview

To put into practice our Deep CNN architecture and transfer learning strategy, we use the Covid IEEE dataset. The prediction accuracy, precision, and recall of the combined CNN technique to categorizing covid or non-covid anomalies in the lung are shown in Table 4.1 below. This table presents the results of a comparison between the suggested design and the fundamental CNN architecture, as well as demonstrates the performance of the proposed architecture. According to the findings of our experiment, utilizing merely CNN architecture was sufficient to categorize lung problems correctly. The CNN design at its most fundamental level only achieves an accuracy of the test of 98.38%, whereas the VGG-16, inception v3, and Dense-net method reach 61.99%, 66.08%, and 71.93%. It has been shown that the performance measurement of Accuracy and recall is also more significant for transfer learning when compared to the basic CNN architecture.

4.2. Hardware Information

To put our Deep CNN model through its paces, we use both the high-performance and remotely accessible platform Google Colab and our own computer.

- Model: HP Probook 440 G7
- Intel Core i5 processor (1.60 GHz - 4.20 GHz)
- 8GB Ram DDR4, Storage: 1TB HDD + 256 m.2 SSD

4.3 Model training

Once our data is imported, we split it up as follows: 60% for training, 20% for testing, and 20% for verifying our results. Next, we downscale all images to 225x225x3 so the network can handle them. We used the data augmentation approach to produce fresh training samples after deciding on a BATCH size of 32 and an EPOCH of 300 (both workstation-specific). We used CNNs to independently classify the given data, then allocate it to a particular class.

To compare the three CNNs, we used the Accuracy, sensitivity, specificity, and F1 score as our metrics of choice. These standard variables will be discussed in detail below.

4.4 Analysis of our suggested model's functionality

The present study used batches of 32 samples and 300 epochs to train the network. The estimated training and validation accuracy was 99.38% and 98.60%, respectively. The test accuracy is 98.13%. In our suggested model, training and validation performance is similar.

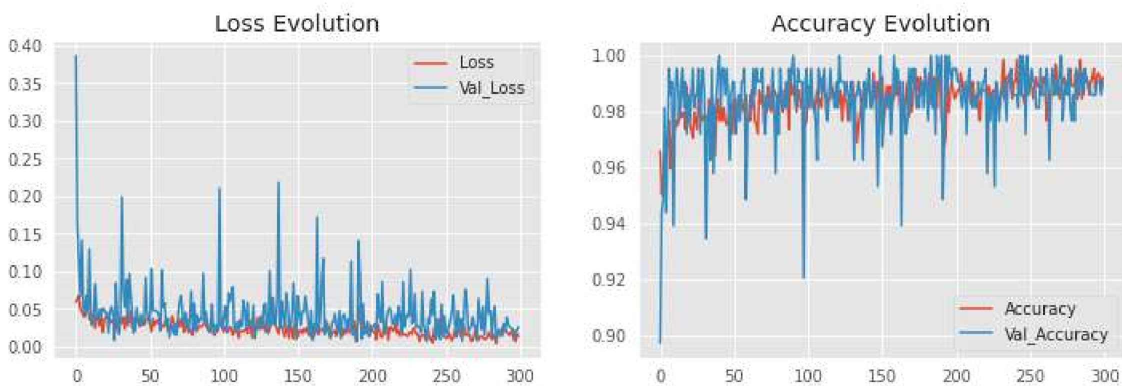
4.4.1 Accuracy with loss graph of the suggested model

Training and validation accuracy improves up to 300 epochs when it stabilizes. Since the training and validation accuracies are comparable, there is no overfitting. Figure 5 (a and

b) and Table 3 [training accuracy and loss as a function of epoch] illustrate this point. One advantage of the proposed architecture is that it remembers the best-trained model across all 300 training epochs.

Table 4.1: Accuracy & loss vs epochs for training & validation

Epochs	Accuracy		Loss	
	Train	Validation	Train	Validation
50	97.20	99.07	0.0400	0.0408
100	97.67	99.53	0.0309	0.0234
150	98.76	100	0.0195	0.0098
200	98.45	99.53	0.0288	0.0207
250	98.91	98.60	0.0144	0.0408
300	99.22	99.07	0.0134	0.0262



(a) (b)

Figure 4.1: (a) Train and Validation loss vs. Epochs (b) Train and Validation accuracy vs. Epochs for the proposed COVID-Deep CNN

4.4.2 Performance Metrics of proposed model:

Accuracy in classifying data is tabulated in the confusion matrix [52]. True positives (T.P.), false positives (F.P.), false negatives (F.N.), and true negatives (T.N.) are all shown with the other evaluation measures (T.N.). Below, we define each of these phrases.

- The number of photos we made an accurate class prediction (COVID or normal) is denoted by True Positives (T.P.).
- True Negatives (T.N.) indicate that the photographs do not belong to the specified category.
- In the case of F.P., we incorrectly predicted a class when the photos in question did not belong to that category (sometimes referred to as a "type I mistake").
- We thought these pictures didn't fit a given category, but they did. This is an example of a false negative (F.N.) (also known as a "type II mistake").

Accuracy (equation (8)), recall (equation (9)), precision (equation (10)), and F1-score (equation (11)) were computed to evaluate the effectiveness of the models.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \tag{8}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{9}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{10}$$

$$\text{F1 score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{11}$$

Figure 4.2: Confusion Matrix for the proposed model

As a consequence of this, we can see that our model can make accurate predictions. A few inaccurate predictions were all that the model needed. However, it provides accurate predictions for COVID as well as non-COVID. COVID has 107 accurate predictions, and the one inaccurate prediction does not meet the criteria for a non-covid prediction. Aside from that, the COVID answer is the incorrect answer for the non-covid three predictions, but the rest of the answers are correct for the real non-covid answers.

4.4.3 Performance evaluation

Results from the experiments summarized in table 2 show that the proposed method outperforms the baseline method on the preprocessed data, concerning measures of Accuracy, precision, recall, f1-score, and support score for both covid (0 = covid, 1 = non-covid).

Table 4.2: Model performance evaluation score

	Covid	Non- covid	Accuracy	Macro avg	Weighted avg
precision	0.972727	1.000000	0.985981	0.986364	0.986364
recall	1.000000	0.971963	0.985981	0.985981	0.985981
f1-score	0.986175	0.985782	0.985981	0.985979	0.985979
Support	107.000000	107.000000	0.985981	214.000000	214.000000

4.4.4 Transfer Learning performance evaluation

So, this section shows the experiments that were done and the results of those experiments so that the performance of the proposed method can be judged. There are three deep learning architectures in use: VGG-12, Densenet, and Inceptionv3. Each model's default parameters are set during the training process, and the training is done with the training data. The test data comprises X-rays test images from two different types. After the tests were done, the results are shown in Tables 2–4 as the confusion matrix.

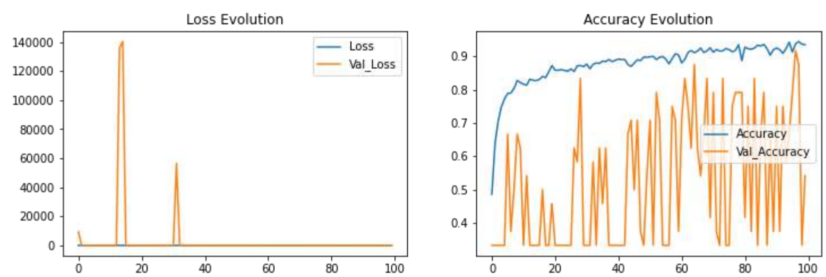


Figure 4.3: Accuracy and loss graph vs epochs of VGG16 model

Table 4.3: The performance of VGG16

		Actual Class		Precision
		Non Covid	**Covid**	
Predicted class	***Non Covid***	35	12	0.63
	Covid	9	30	0.61
Recall		0.70	0.60	

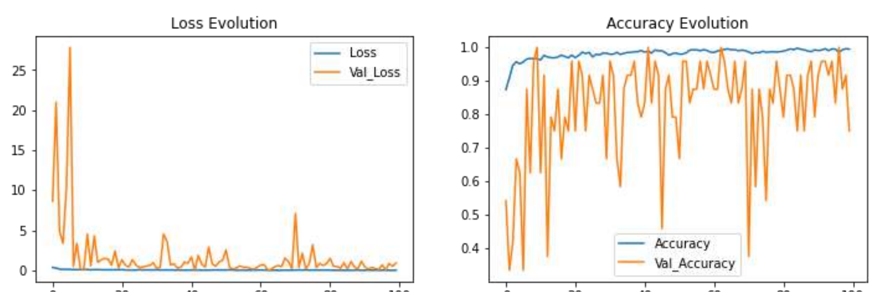


Figure 4.4 : Accuracy and loss graph vs epochs of Densenet model

Table 4.4: The performance of Densenet

		Actual Class		Precision
		Non Covid	**Covid**	
Predicted class	***Non Covid***	37	11	0.60
	Covid	8	28	0.68
Recall		0.77	0.56	

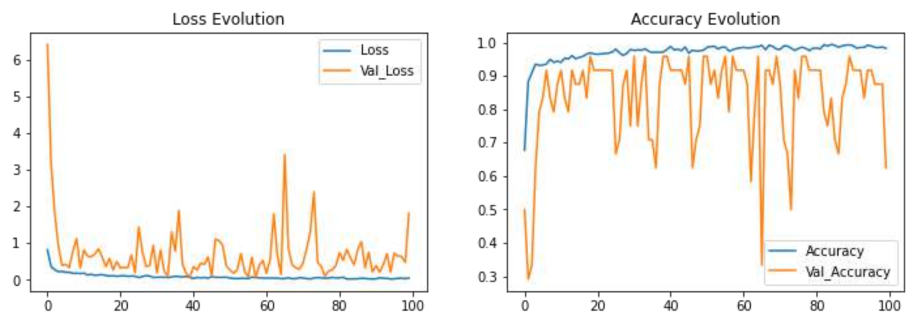


Figure 4.5: Accuracy and loss graph vs epochs of Inceptionv3 model

Table 4.5: The performance of Inceptionv3

		Actual Class		Precision
		Non Covid	**Covid**	
Predicted class	***Non Covid***	36	10	0.64
	Covid	5	32	0.74
Recall		0.72	0.64	

Chapter 5
Comparison

5.1 Introduction

This is the best way to analysis with our work by comparing other work or models; we will compare our work and other recent work.

5.2 Compare the proposed model with different model

Figure 14 reveals that the suggested system achieves remarkable results that are more precise than previous approaches. The suggested enhanced Inceptionv3 system is lightweight in comparison to previous models like VGG16 or DenseNet. Existing approaches have lower F1 scores, while our suggested deep CNN system has higher precision, accuracy, and sensitivity.

Table 5.1: Comparing our proposed model with simulated transfer learning approaches

Model	**Precision**		**Accuracy**
	Covid	Non-covid	
Proposed Deep CNN	0.972	1.00	99. 38%
VGG16	0.61	0.63	61.99%
Densenet	0.68	0.60	71.73%
Inceptionv3	0.74	0.64	66.08%

Here we get our top-notch accuracy and precision by comparing other transfer learning methods we have applied. However, we can see in detail in the table above and obtained our best accuracy of 99.38%

5.2 Compare the proposed model with recent work

Diverse DL techniques have been applied in several investigations to detect COVID-19 infections in C.X.R. photos. [70–80], to the extent seen in Table 1. COVID-19 identification and diagnostic systems based on C.X.R. photos were found to have significant security weaknesses, emphasizing the demand for further research. For starters, most existing approaches have only been verified on small C.X.R. datasets with a negligible number of COVID-19-positive examples. Due to the tiny sample numbers, the outcomes of the suggested systems cannot be accurately represented. Furthermore, there has been less emphasis on creating and training a bespoke DL model from scratch because of a lack of a substantial dataset comprising a considerable number of C.X.R. pictures with known COVID-19 infection.

The suggested system's performance and reliability are compared with the most recent state-of-the-art COVID-19 detection systems. In this section, we show the proposed first version system's outcomes and compare them with other researchers' work, which is illustrated in Table 9.

Table 5.2: Comparison of our work with recent works

Recent Work	Techniques Used	Number of Classes	Accuracy
Khan et al. [70]	CoroNet	4	89.6%
Ucar and Korkmaz [71]	Bayes-SqueezeNet	3	98.3%
Apostopolus et al. [72]	VGG-19	3	93.48%
Sahinbas & Catak [73]	VGG-16, VGG-19, ResNet, DenseNet, InceptionV3	2	80%
Jamil and Hussain [74]	Deep CNN	2	93%
Alzab et al. [75]	VGG-16	2	
Joaquin. [76]	ResNet-50	2	96.2%
Sethy et al. [77]	ResNet-50 + SVM	3	95.33%
Houssein et al. [78]	hybrid quantum classical CNNs	3	88.6%
Oh et al. [79]	ResNet-18	3	88.9%
Brunese et al. [80]	VGG-16	3	96%
Our proposed work	**Deep CNN**	**2**	**99.38%**

Chapter 6
Conclusion and Future work

5.1 Conclusion

This study aims to develop a new COVID-CNN model that is very good at finding COVID-19 in X-rays of patients' lungs when they have severe symptoms. The model given is robust and correctly sorts X-ray images into COVID-19 and non-COVID categories 99.38 percent of the time. A dataset of 1104 lung X-ray images, of which only 60% were used for training, was used to test how well the proposed model worked. The performance was evaluated in terms of accuracy, sensitivity, precision, and f1 score. Compared to other techniques that are thought to be state-of-the-art, the proposed model was very good at classifying X-ray images into two groups. Radiologists could use the suggested method to help them quickly diagnose COVID-19 and tell it apart from pneumonia that isn't caused by COPD. This research is significant for therapy because it speeds up the time it takes to diagnose and pre-screen for COVID-19 infection before the results of RT-PCR are available. In addition, the monetary weight that must be faced by the cost of identifying the deadly disease may be considerably reduced, which may be a massive relief for many people.

5.2 Our future work

Obtaining access to the relevant data and fixing the AI deep learning architecture were two significant challenges that needed to be surmounted before we could successfully automate the identification of COVID-19. The limitations imposed by the dataset are at the heart of the issues with the model that has been proposed. Because so few high-quality images in the public domain were taken with COVID-19, the researchers had to limit the data they collected. In the not-too-distant future, it will be able to fine-tune the structure of the network by modifying the number of layers, nodes, and parameters in each layer. This will make it feasible to customize the network.

One has to have expertise to choose the learning rate, the number of epochs, the degree of regularization, and the structure and optimization of the network layers and nodes. Combining data from a wide variety of sources is one strategy for improving the overall success of model building in the future. It is also feasible to recognize a mutant COVID-19 infection with a chest X-ray. We have only utilized two different classes for training the model so far. Still, in our future work, which will focus on accuracy and employ a multi-class detection method model, we intend to extend it into more classes. It would be bolstered further by respiratory syncytial viruses and bacteria. Because our COVID DeepNet model only evaluates lung infection, it only applies to the COVID-19 screening process. This implies that it can only be used for those already exhibiting symptoms. Validating new significant symptoms using an appropriate model and combining the results of those validations into the existing model may produce a more accurate forecast.

References

[1] "Neural networks and Deep Learning - Blog - ACS Solutions," ACS Solutions, Feb. 26, 2020. https://acsicorp.com/blogs/fundamentals-artificial-neural-networks are-to-deep-learning-what-atoms-are-to-matter/ (accessed Oct. 06, 2022).

[2] "https://www.britannica.com/technology/artificial-intelligence/Is-strong-AI possible," Corporate Finance Institute, Jan. 15, 2022. https://corporatefinanceinstitute.com/resources/knowledge/other/artificial intelligence-ai/ (accessed Oct. 14, 2022).

[3] "Basic Concepts in Machine Learning - Javatpoint," www.javatpoint.com. https://www.javatpoint.com/basic-concepts-in-machine-learning (accessed Oct. 14, 2022).

[4] Tian H, Chen SC, Shyu ML. Evolutionary programming based deep learning feature selection and network con struction for visual data classifcation. Inf Syst Front. 2020;22(5):1053–66.

[5] "Neural networks and Deep Learning - Blog - ACS Solutions," ACS Solutions, Feb. 26, 2020. https://acsicorp.com/blogs/fundamentals-artificial-neural-networks-are-to-deep-learning-what-atoms-are-to-matter/ (accessed Oct. 06, 2022)

[6] Yao G, Lei T, Zhong J. A review of convolutional-neural-network-based action recognition. Pattern Recogn Lett. 2019;118:14–22

[7] Zhang Z, Cui P, Zhu W. Deep learning on graphs: a survey. IEEE Trans Knowl Data Eng. 2020. https://doi.org/10. 1109/TKDE.2020.2981333

[8] Shrestha A, Mahmood A. Review of deep learning algorithms and architectures. IEEE Access. 2019;7:53040–65

[9] Najafabadi MM, Villanustre F, Khoshgoftaar TM, Seliya N, Wald R, Muharemagic E. Deep learning applications and challenges in big data analytics. J Big Data. 2015;2(1):1.

[10] Goodfellow I, Bengio Y, Courville A, Bengio Y. Deep learning, vol. 1. Cambridge: MIT press; 2016

[11] Goh GB, Hodas NO, Vishnu A. Deep learning for computational chemistry. J Comput Chem. 2017;38(16):1291–307.

[12] Li Y, Zhang T, Sun S, Gao X. Accelerating fash calculation through deep learning methods. J Comput Phys. 2019;394:153–65.

[13] Yang W, Zhang X, Tian Y, Wang W, Xue JH, Liao Q. Deep learning for single image super-resolution: a brief review. IEEE Trans Multimed. 2019;21(12):3106–21.

[14] Tang J, Li S, Liu P. A review of lane detection methods based on deep learning. Pattern Recogn. 2020;111:107623.

[15] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classifcation with deep convolutional neural networks. Commun ACM. 2017;60(6):84–90.

[16] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2016. p. 770–8.

[17] Bhowmick S, Nagarajaiah S, Veeraraghavan A. Vision and deep learning-based algorithms to detect and quantify cracks on concrete surfaces from uav videos. Sensors. 2020;20(21):6299.

[18] Hasan RI, Yusuf SM, Alzubaidi L. Review of the state of the art of deep learning for plant diseases: a broad analysis and discussion. Plants. 2020;9(10):1302

[19] Ng A. Machine learning yearning: technical strategy for AI engineers in the era of deep learning. 2019. https://www.mlyearning.org.

[20] Rozenwald MB, Galitsyna AA, Sapunov GV, Khrameeva EE, Gelfand MS. A machine learning framework for the prediction of chromatin folding in Drosophila using epigenetic features. PeerJ Comput Sci. 2020;6:307.

[21] Amrit C, Paauw T, Aly R, Lavric M. Identifying child abuse through text mining and machine learning. Expert Syst Appl. 2017;88:402–18.

[22] Hossain E, Khan I, Un-Noor F, Sikander SS, Sunny MSH. Application of big data and machine learning in smart grid, and associated security concerns: a review. IEEE Access. 2019;7:13960–88.

[23] Crawford M, Khoshgoftaar TM, Prusa JD, Richter AN, Al Najada H. Survey of review spam detection using machine learning techniques. J Big Data. 2015;2(1):23.

[24] Deldjoo Y, Elahi M, Cremonesi P, Garzotto F, Piazzolla P, Quadrana M. Content-based video recommendation system based on stylistic visual features. J Data Semant. 2016;5(2):99–113.

[25] Al-Dulaimi K, Chandran V, Nguyen K, Banks J, Tomeo-Reyes I. Benchmarking hep-2 specimen cells classifca- tion using linear discriminant analysis on higher order spectra features of cell shape. Pattern Recogn Lett. 2019;125:534–41.

[27] Liu W, Wang Z, Liu X, Zeng N, Liu Y, Alsaadi FE. A survey of deep neural network architectures and their applica tions. Neurocomputing. 2017;234:11–26.

[28] Pouyanfar S, Sadiq S, Yan Y, Tian H, Tao Y, Reyes MP, Shyu ML, Chen SC, Iyengar S. A survey on deep learning: algorithms, techniques, and applications. ACM Comput Surv (CSUR). 2018;51(5):1–36.

[29] Alom MZ, Taha TM, Yakopcic C, Westberg S, Sidike P, Nasrin MS, Hasan M, Van Essen BC, Awwal AA, Asari VK. A state-of-the-art survey on deep learning theory and architectures. Electronics. 2019;8(3):292.

[30] Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: 2005 IEEE computer society confer ence on computer vision and pattern recognition (CVPR'05), vol. 1. IEEE; 2005. p. 886–93.

[31] Potok TE, Schuman C, Young S, Patton R, Spedalieri F, Liu J, Yao KT, Rose G, Chakma G. A study of complex deep learning networks on high-performance, neuromorphic, and quantum computers. ACM J Emerg Technol Comput Syst (JETC). 2018;14(2):1–21.

[32] Adeel A, Gogate M, Hussain A. Contextual deep learning-based audio-visual switching for speech enhancement in real-world environments. Inf Fusion. 2020;59:163–70.

[33] "What Is Deep Learning? | How It Works, Techniques & Applications," What Is Deep Learning? | How It Works, Techniques & Applications - MATLAB & Simulink. https://www.mathworks.com/discovery/deep-learning.html (accessed Oct. 14, 2022).

[34]"https://towardsdatascience.com/understand-the-architecture-of-cnn 90a25e244c7#:~:text=The%20different%20layers%20of%20a%20CNN%20There%20are,the%20ReLU%20correction%20layer%20and%20the%20fullyconnected%20layer.," What Is Deep Learning? | How It Works, Techniques & Applications - MATLAB & Simulink. https://www.mathworks.com/discovery/deep-learning.html (accessed Oct. 14, 2022).

[35] “Home,” Edge AI and Vision Alliance, Oct. 12, 2022. https://www.embedded-vision.com (accessed Oct. 14, 2022).

[36] “Basic CNN Architecture: Explaining 5 Layers of Convolutional Neural Network | upGrad blog,” upGrad blog, Jul. 28, 2022. https://www.upgrad.com/blog/basic-cnn-architecture/ (accessed Oct. 14, 2022).

[37] “Overfitting vs. Underfitting: What Is the Difference? | 365 Data Science,” 365 Data Science, Aug. 27, 2021. https://365datascience.com/tutorials/machine-learning-tutorials/overfitting-underfitting/ (accessed Oct. 14, 2022).

[38] Iofe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift; 2015. arXiv preprint arXiv:1502.03167

[39] Ruder S. An overview of gradient descent optimization algorithms; 2016. arXiv preprint arXiv:1609.04747

[40] Bottou L. Large-scale machine learning with stochastic gradient descent. In: Proceedings of COMPSTAT'2010. Springer; 2010. p. 177–86.

[41] Hinton G, Srivastava N, Swersky K. Neural networks for machine learning lecture 6a overview of mini-batch gradi ent descent. Cited on. 2012;14(8). 85.

[42] Zhhang Z. Improved Adam optimizer for deep neural networks. In: 2018 IEEE/ACM 26th international symposium on quality of service (IWQoS). IEEE; 2018. p. 1–2.].

[43] Zhang Z. Improved Adam optimizer for deep neural networks. In: 2018 IEEE/ACM 26th international symposium on quality of service (IWQoS). IEEE; 2018. p. 1–2.]

[44] Khan, A.I.; Shah, J.L.; Bhat, M.M. CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest X-ray images. Comput. Methods Programs Biomed. 2020, 196, 105581. [CrossRef]

[45] Ucar, F.; Korkmaz, D. COVIDiagnosis-Net: Deep Bayes-SqueezeNet based diagnosis of the coronavirus disease 2019 (COVID-19) from X-ray images. Med Hypotheses 2020, 140, 109761. [CrossRef]

[46] Apostolopoulos, I.D.; Aznaouridis, S.I.; Tzani, M.A. Extracting possibly representative COVID-19 biomarkers from X-ray images with deep learning approach and image data related to pulmonary diseases. J. Med Biol. Eng. 2020 , 1–8. [CrossRef] [PubMed]

[47] Sahinbas, K.; Catak, F.O. Transfer learning-based convolutional neural network for COVID-19 detection with X-ray images. In Data Science for COVID-19; Elsevier: Amsterdam, The Netherlands, 2021; pp. 451–466.

[48] Jamil, M.; Hussain, I. Automatic detection of COVID-19 infection from chest X-ray using deep learning. medRxiv 2020. [CrossRef]

[49] Alazab, M.; Awajan, A.; Mesleh, A.; Abraham, A.; Jatana, V.; Alhyari, S. COVID-19 prediction and detection using deep learning. Int. J. Comput. Inf. Syst. Ind. Manag. Appl. 2020, 12, 168–181.

[50] Alzubaidi, Laith, Jinglan Zhang, Amjad J. Humaidi, Ayad Al-Dujaili, Ye Duan, Omran Al-Shamma, José Santamaría, Mohammed A. Fadhel, Muthana Al-Amidie, and Laith Farhan. "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions." Journal of big Data 8, no. 1 (2021): 1-74.

[51] Chatfield, K.; Simonyan, K.; Vedaldi, A.; Zisserman, A. Return of the devil in the details: Delving deep into convolutional nets. arXiv 2014, arXiv:1405.3531.

[52] Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. arXiv 2014, arXiv:1409.1556.].

[53] T. Ai et al., "Correlation of Chest CT and RT-PCR Testing in Coronavirus Disease (COVID-19) in China: A Report of 1014 Cases," Radiology, p. 200642, Feb. 2020.

[54] J. P. Cohen, P. Morrison, and L. Dao, "COVID-19 Image Data Collection," arXiv, Mar. 2020.])

[55] P. Kumar and S. Kumari, “Detection of coronavirus Disease (COVID-19) based on Deep Features,” preprints.org, no. March, p. 9, Mar. 2020.],

[56] [H. S. Maghdid, A. T. Asaad, K. Z. Ghafoor, A. S. Sadiq, and M. K. Khan, "Diagnosing COVID-19 Pneumonia from X-Ray and C.T. Images using Deep Learning and Transfer Learning Algorithms," arXiv, Mar. 2020.]

[57] F. Shan et al., "Lung Infection Quantification of COVID-19 in C.T. Images with Deep Learning," arXiv, Mar. 2020.

[58] X. Xu et al., "Deep Learning System to Screen Coronavirus Disease 2019 Pneumonia," arXiv, Feb. 2020.

[59] S. Wang et al., "A deep learning algorithm using C.T. images to screen for Corona Virus Disease (COVID19)," medRxiv, p. 2020.02.14.20023028, Apr. 2020.],

[60] S. Wang et al., "A deep learning algorithm using C.T. images to screen for Corona Virus Disease (COVID19)," medRxiv, p. 2020.02.14.20023028, Apr. 2020.],

[61] L. Wang and A. Wong, "COVID-Net: A Tailored Deep Convolutional Neural Network Design for Detection of COVID-19 Cases from Chest X-Ray Images," arXiv, Mar. 2020.]

[62] M. Farooq and A. Hafeez, "COVID-ResNet: A Deep Learning Framework for Screening of COVID19 from Radiographs," arXiv, Mar. 2020.]

[63] L. Wang and A. Wong, "COVID-Net: A Tailored Deep Convolutional Neural Network Design for Detection of COVID-19 Cases from Chest X-Ray Images," arXiv, Mar. 2020.]

[64] Zhang, J.; Xie, Y.; Li, Y.; Shen, C.; Xia, Y. Covid-19 screening on chest X-ray images using deep learning based anomaly detection. arXiv 2020, arXiv:2003.12338]

[65] Wang, L.; Lin, Z.Q.; Wong, A. Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest X-ray images. Sci. Rep. 2020, 10, 1–12. [CrossRef]

[66] Hemdan, E.E.D.M.A.S.; Karar, M.E. COVIDXnet: A framework of deep learning classifiers to diagnose COVID-19 in X-ray images. arXiv 2020, arXiv:2003.11055.

[67] [Narin, A.; Kaya, C.; Pamuk, Z. Automatic detection of coronavirus disease (Covid-19) using X-ray images and deep convolutional neural networks. Pattern Anal. Appl. 2021, 24, 1207–1220. [CrossRef] [PubMed]]

[68] M. L. Huang and Y. C. Liao, "A lightweight CNN-based network on COVID-19 detection using X-ray and C.T. images," Computers in Biology and Medicine, vol. 146, article 105604, 2022.

[69] T. Sanida, A. Sideris, D. Tsiktsiris, and M. Dasygenis, "Lightweight neural network for COVID-19 detection from chest X-ray images implemented on an embedded system," Technologies, vol. 10, no. 2, p. 37, 2022.

[70] Khan, A.I.; Shah, J.L.; Bhat, M.M. CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest X-ray images. Comput. Methods Programs Biomed. 2020, 196, 105581. [CrossRef]

[71] Ucar, F.; Korkmaz, D. COVIDiagnosis-Net: Deep Bayes-SqueezeNet based diagnosis of the coronavirus disease 2019 (COVID-19) from X-ray images. Med Hypotheses 2020, 140, 109761. [CrossRef]

[72] Apostolopoulos, I.D.; Aznaouridis, S.I.; Tzani, M.A. Extracting possibly representative COVID-19 biomarkers from X-ray images with deep learning approach and image data related to pulmonary diseases. J. Med Biol. Eng. 2020 , 1–8. [CrossRef] [PubMed]

[73] Sahinbas, K.; Catak, F.O. Transfer learning-based convolutional neural network for COVID-19 detection with X-ray images. In Data Science for COVID-19; Elsevier: Amsterdam, The Netherlands, 2021; pp. 451–466.

[74] Jamil, M.; Hussain, I. Automatic detection of COVID-19 infection from chest X-ray using deep learning. medRxiv 2020. [CrossRef]

[75] Alazab, M.; Awajan, A.; Mesleh, A.; Abraham, A.; Jatana, V.; Alhyari, S. COVID-19 prediction and detection using deep learning. Int. J. Comput. Inf. Syst. Ind. Manag. Appl. 2020, 12, 168–181.

[76] Joaquin, A. Using Deep Learning to Detect Pneumonia Caused by NCOV-19 from X-ray Images. 2020 Available online: https://towardsdatascience.com/using-deep-learning-to-detect-ncov-19-from-X-ray-images-1a89701d1acd (accessed on 13 December 2021).

[77] Sethy, P.K.; Behera, S.K. Detection of Coronavirus Disease (COVID-19) Based on Deep Features. 2020. Available online: https://www.preprints.org/manuscript/202003.0300/v1 (accessed on 13 December 2021).

[78] Houssein, E.H.; Abohashima, Z.; Elhoseny, M.; Mohamed, W.M. Hybrid quantum convolutional neural networks model for COVID-19 prediction using chest X-ray images. arXiv 2021, arXiv:2102.06535.

[79] Oh, Y.; Park, S.; Ye, J.C. Deep learning COVID-19 features on cxr using limited training data sets. IEEE Trans. Med Imaging 2020, 39, 2688–2700. [CrossRef]

[80] Brunese, L.; Mercaldo, F.; Reginelli, A.; Santone, A. Explainable deep learning for pulmonary disease and coronavirus COVID-19 detection from X-rays. Comput. Methods Programs Biomed. 2020, 196, 105608. [CrossRef]

[81] H. Lidong, Z. Wei, W. Jun, and S. Zebin, "Combination of contrast limited adaptive histogram equalization and discrete wavelet transform for image enhancement," I.E.T. Image Process, vol. 9, no. 10, pp. 908-915, 2015, doi: 10.1049/iet-ipr.2015.0150.

[82] Md Mehedi Hasan, et al., A Combined Approach Using Image Processing and Deep Learning to Detect Pneumonia from Chest X-Ray Image, 2019 3rd International Conference on Electrical, Computer & Telecommunication Engineering (I.C.E.C.T.E.). IEEE (2019).

[83] Lowe DG. Object recognition from local scale-invariant features. In: Proceedings of the seventh IEEE international conference on computer vision, vol. 2. IEEE; 1999. p. 1150–7. 17. Wu L, Hoi SC, Yu N. Semantics-preserving bag-of-words models and applications. IEEE Trans Image Process. 2010;19(7):1908–20.

[84] White Balance Algorithm with Gdal in C# (2017). URL https://adadevelopment.github.io/gdal/white-balance-gdal. html]

[85] White Balance (2020). URL https://docs.gimp.org/2.10/en/gimp-layer-white-balance. html. Accessed

[86] S. M. Pizer, Contrast-limited adaptive histogram equalization: Speed and effectiveness stephen m. pizer, r. eugene johnston, james p. ericksen, bonnie c. yankaskas, keith e. muller medical image display research group, in: Proceedings of the First Conference on Visualization in Biomedical Computing, Atlanta, Georgia, May 22-25, 1990, IEEE Computer Society Press, 1990, p. 337.].

[87] S.-L. Wong, Y.-P. Yu, N. A.-J. Ho, R. Paramesran, Comparative analysis of underwater image enhancement methods in different color spaces, in: 2014 International Symposium on Intelligent Signal Processing and Communication Systems (I.S.P.A.C.S.), IEEE, 2014, pp. 034–038.