COVID-19 detection based on X-ray image and deep learning technique

A project

Submitted to the Council of College of Science for woman, University of Babylon in Partial Fulfillment of the Requirement for Degree of higher Diaploma of Science in Computer

By

Teba Hassan Hadi

Supervised by

Dr. Ali Yakoob Al-Sultan

2021 A.D.   1443 A.H

بِسْمِ اللهِ الرَّحْمَنِ الرَّحِيمِ

قَالُواْ سُبْحَانَكَ لاَ عِلْمَ لَنَا إِلاَّ مَا عَلَّمْتَنَا إِنَّكَ أَنتَ الْعَلِيمُ الْحَكِيمُ

صَدَقَ اللهُ الْعَلِيُّ الْعَظِيمُ

سورة البقرة (٣٢)

Dedication

To who taught me my first words the joy of the spirit and

the essence of life

… My affectionate mother

To that who yanked a living from the cruelty of the rocks

… My dear Father

To that vast space and land that sprouted from the blood of

the martyrs to

… The beloved homeland

To those who bear on them the building of the generation

of the future

… Our esteemed teachers

To all those, I dedicate my work

Supervisor Certification

I certify that project entitled**" Diagnosis of Covid-19 Based on Deep Learning Technique"** was prepared at the Department of computer Sciences/ College of Science for Women/ University of Babylon, by **(Teba Hassan Hadi)** as partial fulfilment of the requirements for the degree of Higher Diploma in Computer Science.

Signature:

Name: Dr. Ali Yakoob Al-Sultan

Date: /12 /2021

Address: University of Babylon/College of Science for Women

The Head of the Department Certification

In view of the available recommendations, I forward the research entitled "**Diagnosis of Covid-19 Based on Deep Learning Technique**" for debate by the examination committee.

Signature:

Name: Dr. Farah M. Hassan, Ph.D.

Date: /12 / 2021

Address: University of Babylon/College of Science for Women

Certification of the examing committee

We are the member of the examing committee, certify that we have read this project entitled "*COVID-19 Detection Based on X-ray Image and Deep Learning Technique*" and after examining the higher diploma student (*Teba Hassan Hadi Al-Hamdani*) in its content in 27/12/2021, and that in our opinon it is accepted as a project for the degree of higher diploma in science\computer science with a degree (excellent).

Committee Chairman: **Committee Member:**

Signature: **Signature:**

Name: Suhad Ahmed Ali   **Name:** Salah Mahdi Al-Obaidi

Scientific order: Prof., PhD PhD   **Scientific order: Lecturer, PhD**

Date: / 1 /2022   **Date:**   / 1 /2022

Committee Member (Supervisor):

Signature:

Name: Ali Yakoob Al-Sultan

Scientific order: Lecturer, PhD

Date: / 1 /2022

Date of Examination: 27\12\2021

Deanship authentication of collegeof science for women.

Approved for the college committee of grade studies.

Signature:

Name: Faez Ali Rashid

Scientific order: Prof., PhD

Address: Dean of College Science for Women

Date: / 1 /2022

Acknowledgments

First of all, I am grateful to the Almighty Allah, the owner of grace and favor who was and is still my support in preparing and achieving this project. My deep love and massive thanks go to the masters of Creatures, the Greatest prophet Mohammed and his progeny (peace be upon them). Praise is to Allah, who illuminated me the way of science and gave me the patience to continue.

Also sincere appreciation and love go to all my family members whose have encouraged me to go through this study and provides me with love and pure affection.

I wish to express my thanks and deep gratitude to my supervisor Dr. Ali Yakoob Al-Sultan for his advices, support, encouragement, and supervision throughout this work to be in the best manner.

I would also like to acknowledge University of Babylon - Collage of Science for Women - Department of Computer Science for providing support in conducting this thesis.

Finally, I would like to thank the people who gave me help and advice.

Abstract

A new Coronavirus Disease-2019 (COVID-19) has been spread quickly as an acute respiratory distress syndrome (ARDS) among individuals worldwide. Moreover, the number of COVID-19 test kits available in hospitals is limited as compared to the growing number of cases every day. Therefore, it is necessary to introduce an automatic detection system as a fast alternative diagnostic method to avoid COVID-19 from spread among people.

The purpose of this project is to propose an automated method based on Convolutional Neural Network (CNN) for identifying of COVID-19 pneumonia-infected patients using chest x-ray images.

The proposed system consis of three stages. The first stage is the preprocessing stage which begins with data augmentation technique, converting x-ray images into gray scale image, resizing, Contrast Limited Adaptive Histogram Equalization (CLAHE) technology for enhancement and Normlization was used to standrize the data. The second stage, The CNN based on x-ray Images was used for features extraction. In the final stage, SoftMax function is employed for the classification of COVID-19 pneumonia infected patients.

The proposed method has been tested on dataset and the results showed high accuracy of COVID-19 classification and reached (100%) in the test dataset (220 x-ray images). The proposed method is fast in diagnosis and the x-ray image prediction time takes approximately (1) second for each image with relatively small number of trainable parameters (6,446,210).

Table of Abbreviations

Abbreviation	Meaning
ARDS	Acute Respiratory Distress Syndrome
COVID-19	Coronavirus Disease-2019
CXR	Chest X-ray
CNN	Convolutional Neural Network
CT	Computed tomography
DA	Data Augmentation
DL	Deep Learning
DNN	Deep neural network
FFNs	feedforward neural networks
FC	Fully Connected
FN	False Negative
FP	False Positive
ML	Machine Learning
MLP	Multi-Layer Perceptron
RT-PCR	Reverse Transcription-Polymerase Chain Reaction
RNN	Recurrent neural networks
ReLU	Rectified Linear Unit
TN	True Negative
TP	True Positive
WHO	World Health Organization

Table of Figures

Table of Tables

Table of Algorithm

Algorithm	Title of Algorithm	Page
Algorithm 3.1	Pre-processing	31
Algorithm 3.2	Training CNN model	35

Table of content

Chapter One

Introduction

1.1 Introduction

Currently, coronavirus is considered a central problem because it critically affects the economy and worldwide health systems [1]. The Covid-19 virus uses the air to spread itself in the community while a Covid-19 infected person coughs or sneezes. Therefore, a standard symptomatic procedure of inverse transcription-polymerase chain reaction is accepted worldwide to diagnose or detect viral nucleic acid of suspected individuals.

The X-Ray of Chest or/and CT scan images technologies can be employed to Covid-19 suspected individuals because the reverse transcription-polymerase chain reaction (RT-PCR) test takes time compared to the spreading of the virus in the community. Beside, time taken by RT-PCR test, false-positive errors, and shortage of test kits compared to coronavirus infected persons makes it inefficient. Usually, the Chest X-Ray (CXR) method is the best option because it helps the radiologists to identify the chest pathology without introducing the patient to high CT scan radiation [2-5]. As a result, an automatic detection method must be implemented as a rapid replacement diagnosis option to avoid Covid-19 from extending between people.

Deep Learning including such Convolutional Neural Network (CNN), has changed the expectations in many AI applications in data processing by reaching human-level accuracy in many tasks, including medical image analysis [6]. Therefore, CNN was used in this project because it does not require manually features extraction as well as their use of shared weights and the local connection. These two features exceedingly decreased the number of parameters in the network and thus the training time will be reduced [7]. The main objective of this project is to improve

the accuracy of the deep learning model than the current methods by using fewer parameters, and thus the computational power will be less.

1.2   Problem Statement

A novel coronavirus (COVID-19) has spread rapidly among people as an acute respiratory distress syndrome (ARDS) among individuals worldwide. Moreover, the health sector's resources are limited, such as the number of Covid-19 test kits available in hospitals as compared to the growing number of cases every day, the number of mechanical ventilators for patients that need it. Therefore, it is necessary to introduce an automatic detection system as a quick alternative diagnosis option to prevent Covid-19 from spreading among people.

1.3 Project Objectives

Bulding a system to automatically identify Covid-19 disease by using a CNN model. Improving the accuracy of detection of Covid-19 virus in X-ray images using deep learning and data augmentation technique. Evaluating the proposed technique results and compared to the existing methods.

1.4 Related works

In the related work, several studies have been proposed regarding the discovery of Covid-19. These studies are mainly related to extracting characteristics through X-ray images. As for the techniques that have been used to extract the characteristics, most of the researchers used Convolutional Neural Network models such as VGGNet, ResNets, and DensNet). As for the techniques to classify Covid 19 disease, most researchers used machine learning methods.

Apostolopoulos et al (2020) [8], trained a CNN model to classify the Covid-19 disease. The used method of training is Transfer Learning. The

used dataset is a collection of various available datasets, that are collected by the researchers. Total number of images are 1427 x-ray images. The best results obtained by the researchers are 96.78%, 98.66%, and 96.46% of accuracy, sensitivity, and specificity respectively.

Zhang et al (2020) [9], proposed system to classify Covid-19 that is consisting of three parts. The first part is what they called it Backbone network, which is 18-layer CNN to learn and extract the x-ray images features. The second part is what they called it "Classification head", which is an Artificial Neural Network of input layer, 100 neuron hidden layer, and a single neuron output layer. The third part is what they name it "Anomaly detection head", Which is similar in the design to the Classification head part. The Classification head gives the class of the image, and Anomaly detection head gives a score for each classification. The dataset is collected by the researchers directly from hospitals and online. The results showed (96.00%) accuracy, and (96.00%) sensitivity for detecting positive Covid-19. And (70.65%) accuracy, and (70.6%) specificity for negative Covid-19 (normal case).

Adhikari's (2020) [10], proposed system consists of two design stages. Both of them are based on DenseNet design. However, the first stage will read and classify x-ray images, while the second one will read and classify CT images. The obtained results are (99.0%) accuracy, and (94.1%) specificity using the DenseNet network on the X-rays images. And (87.0%) accuracy and (86.5%) specificity on CT Scans.

Alqudah et al. (2020) [11], used different CNN designs to extract the features from x-ray images, and used different classifiers to classify the images, Including Softmax Classifier, Support Vector Machine-(SVM), K-Nearest Neighbor-(KNN), and Random Forest-(RF). The employed dataset

is a collection of different datasets available online, that are collected by the researchers themselves. Totally, the best results obtained are (98.0%) and higher for all accuracy, sensitivity, specificity, and precision.

Khan et al. (2020) [12], used Xception CNN model, that is pre-trained on the ImageNet dataset. This model is trained on a dataset that is collected by the researchers from online available datasets. Their results are (89.6%) accuracy, (93.0%) precision, and (98.2%) recall for four classes of lung diseases. And for three classes they achieved (95.0%) accuracy.

Hemdan et al. (2020) [13], tried different popular architectures of CNN with slight changes and modifications. However, only two architectures showed promising and similar results. Which are VGG19 and DenseNet. With (89.0%) and (91.0%) F1-score for negative and positive classes of Covid-19 respectively. The used dataset is collected directly by the researchers.

Apostopolus et al. (2020) [14], used the CNN architecture of Mobile Net. Where they trained it on a 3905 x-ray images. For seven classes of lung diseases, they got (87.66%) overall accuracy. However, for the Covid-19 classification, they got (99.18%) accuracy, (97.36%) Sensitivity, and (99.42%) Specificity.

Sahinbas et al. (2020) [15], suggested using pre-trained CNN model (i.e. Transfer Learning). They used the pre-trained architectures of VGG16, VGG19, ResNet, DenseNet, and InceptionV3. VGG16 showed the highest results among the others with (80.0%) accuracy.

Jamil M et al. (2020) [16], designed, developed and trained their own CNN model. The training and testing dataset is obtained from online website (Kaggle). Their final result gave a total accuracy of (93.0%).

1.5 Layout of Project

In addition to chapter one, the reset of this project includes four chapters, which have been organized as follows:

- **Chapter Two, "Theoretical Background":** describes the (Data Augmentation), and the technique of classification for the extraction of X-ray features by Deep Learning (Convolutional Neural Network), and evaluation criteria used in this project.

- **Chapter Three, "Proposed System":** An overview of the structure and execution of classification algorithms that are used in the proposed diagnosis Covid-19 system is demonstrated in this chapter.

- **Chapter Four, "Results and Discussions":** A description of the various experiments of each step of the work was introduced in this chapter. Moreover, a discussion of the evaluations and results obtained from the execution of the suggested method.

- **Chapter Five, "Conclusions and Future Works":** A summary of the study project is presented in this section. Besides, this section shows the future work to be undertaken in this respect.

Chapter Two

Theoretical Background

2.1   Introduction

This chapter provides an overview of the various techniques and processes used in the COVID-19 detection system that has been suggested. This chapter is arranged as follows: Firstly, image processing techniques such as converting images to Grayscale images, enhancement to show the features, normalization to standardize the data. The data augmentation technique is used to increase the dataset to prevent the overfitting problem. Secondly, description of Deep Learning, Convolutional Neural Network with their equations are presented. A description of the performance metrics such as Specificity, Sensitivity, Accuracy, F1-score, and precision are also covered.

2.2   Image Processing Techniques

This section covers the image processing methods that are used in this project for pre-processing and improving the input images.

2.2.1 Converting Color Image (RGB) to Grayscale Image

Most of the advantages of converting a color image to gray-scale domain are for data reduction, because the grayscale domain has only one channel rather than three as in RGB domain, which leads to speed the processing. In this project, the luminance method is used for gray-scale converting as explained in equation (2.1) [17].

$$Gray\,(i, j) = (0.2989 * R) + (0.5870 * G) + (0.1140 * B) \qquad (2.1)$$

Where R=read, G=green, B =blue.

2.2.2 Contrast Limited Adaptive Histogram Equalization (CLAHE)

The enhancement of images is one of the most significant steps in the range of medical image discovery and study, which enhances the

quality of images from a human perspective, eliminates noise and blurring, improves contrast, and shows the image's details. CLAHE is the extended version of the adaptive histogram equalization algorithm (AHE). It was developed to eliminate the overamplification of noises that occurs in the AHE method. CLAHE adopts a technique for reducing contrast amplification that is implemented for each adjacent pixel, which then forms a change mechanism to minimize the noise [18].

The main idea of this method is first to divide the input image into small parts of equal and non-overlapping sizes, which are called tiles. Then, the histogram is computed for each of these parts because this method does not use the global histogram and depends on the local histogram. Next, this method clips the histogram at specified values to overcome the noise over-amplification problem and distribute it regularly to other tiles before calculating the cumulative distribution function. After that, the cumulative histogram is computed in order to make the equalization. The CLAHE technique applies two parameters, the first, clips limit (CL), which is a numerical value specifying the noise amplification. The second is the number of tiles (NT), a numerical value specifying the number of non-overlapping sub regions. The computation of CLAHE is performed in the equation (2.2).

$$p = (p_{max} - p_{min}) * p(f) + p_{min} \qquad (2.2)$$

Where, p represents pixel value after applying CLAHE, p_{max}, p_{min} represents the maximum and minimum pixel value of an image respectively and $p(f)$ represents cumulative probability distribution function after the clip limit [19] [20]. as shown in Figure (2.1) [21].
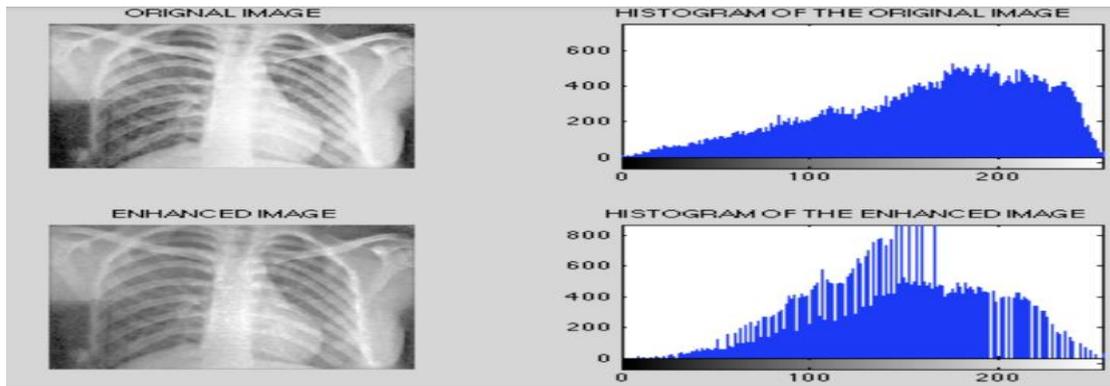
Figure (2.1) CLAHE method on the X-ray image with histogram [21].

2.2.3 Normalization

Normalization is a method used in image processing that changes the range of pixel intensity values. Images with poor contrast due to glare are one example of this phenomenon. The terms "normalization" and "contrast stretching" are often used interchangeably, as the phrase "histogram stretching". Dynamic range expansion is the term used in more general fields of data processing, such as digital signal processing, to refer to this technique.

For a variety of applications, the goal of dynamic range expansion is often to place an image or other kind of signal into a more familiar or normal range for the senses, thus the term normalization. When dealing with a collection of data, signals, or images the aim is often to preserve the dynamic range of the collection in order to avoid mental distraction or fatigue [22].

$$x_n = \frac{x_r - x_{\min}}{x_{\max} - x_{\min}} \qquad (2.3)$$

Where x_n indicates to the value of normalized intensity, x_r refers to the value of a pixel intensity, x_{max} and x_{min} indicates to the minimum and maximum intensity values of an image, respectively [23].

10

2.3   Data Augmentation

The imbalanced datasets can be regarded collectively as a problem when concerning computer vision and image classification issues. The problems of underfitting and overfitting may occur if there is a deficiency of images in each category. This will exceedingly affect the efficiency and performance of deep learning, particularly convolutional neural networks. To avoid the above problems, a technique for augmenting data within covid-19 datasets has been suggested to improve the classifier's performance. Data augmentation is a method that increases the number of images used for training the neural network. That is, creating new data for categories that have fewer numbers in the data set. This process succeeds the constraining effect on data to prevent an asymmetric representation and successfully escape overfitting complications. Appropriate data augmentation techniques can help improving deep learning model strength [24].

The deep CNN models need a tremendous number of data for sufficient training and have seen improved results on larger data sets [25].

There are various techniques to augment classic data such as zooming, flipping, shifting, rotation, add noise, and transformation to be implemented to the original images. Some of the methods are:

1. **Horizontal and Vertical Flip**

    For Data Augmentation, both vertical and horizontal flipping approaches are extremely successful and popular. In terms of execution, augmented data is one of the easiest techniques that has been proved to be beneficial in a dataset. At the same time, horizontal axis flipping is considered to be more usual than vertical axis flipping since it is more suitable for most projects owing to the

high chance of finding images flipped horizontally as illustrated in Figure (2.2) [26].

2. Rotation

Rotations are performed by rotating the image to the right or left on an axis between (0 and 360). As illustrated in Figure (2.2). The maximal angle of rotation is adjusted by the degree of rotation. It is useful whenever it is slight because the network must recognize the object in any direction in the image. This precise rotation of the image can cause problems for certain applications. As the rotation angle increases, the data label is no longer retained after conversion, and Background noise is being introduced. If the background noise was too various compared to the other areas of an image, then the networks can learn incorrect features [27].

The background noise problem may be handled by utilizing the fill mode (contrast) method to transform the resulting blank area into white pixels after rotation [28].

3. Zooming

Zoom Augmentation will randomly zoom in or out of the image, and adds new pixels for the image when zooming out [29], as shown in Figure (2.2).

4. Fill Mode

To retain the image's quality, there are numerous choices for filling in the empty spaces generated by data augmentation techniques (such in zooming and rotation) with various pixel values. There are many methods for filling the empty space, including:

A. **Nearest:** in this method, the empty space is filled with the values of the nearest pixels. Notice that this is the default choice for Keras.

12

B. **Reflect:** in this method, the empty space will be filled with a reflection of the original image, in the opposite order as the known pixels.

C. **Constant:** this method fills in all the empty pixels in the image with a constant value.

D. **Wrap:** this method is somewhat similar to the reflect mode. However instead of reflecting the pixels in opposite order, it will copy the pixels in their normal direction [30].
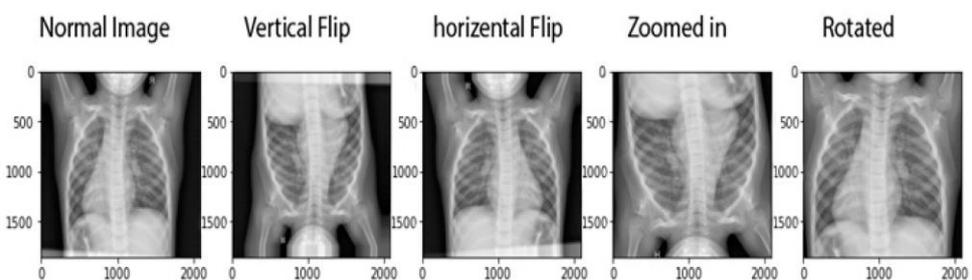
Figure (2.2) several techniques for data augmentation [30].

2.4   Machine Learning Models

It is one of the Artificial Intelligence sections that depends on computer science, statistics, and mathematics. the basic goal of Machine Learning and statistical modelling is to enable computer programs to learn from data, and then make appropriate decisions based on the model that has been learned by a prior experience or prior skills. The machine learns directly from the fundamental input data structure and becomes more intelligent [27].

2.4.1 Machine Learning Methods

Machine Learning methods are divided into three categories: supervised, unsupervised, and reinforcement learning.

1. **Supervised Learning**

     The data is labelled in Supervised Learning. This implies that the Machine Learning algorithm is trained on data with known input values (features or variables, denoted as *x*) and output values (target, written as *y*). The supervised algorithm's job is to match the inputs in a function to produce a precise output ($y = f(x)$). The objective is to achieve the greatest possible match between *x* and *y*, such that when a new value is input into the model, it produces the desired result [31]. Supervised Learning is one of the most popular learning methods. It may also be split into two categories:

  A. Classification: is a group of data related to each other where the intention is to divide the points of this data into a collection of predefined categories based on some characteristics of the data. This method could be achieved by different algorithms, such as Neural Network, Support Vector Machines, k-nearest-neighbors, Random forest, Naive Bayes.

  B. Regression: is used to forecast real values that are referred to as continuous values. This method is achieved by different algorithms, such as Decision Trees, Linear Regression, Assembly methods.

2. **Unsupervised Learning**

     Common algorithms within unsupervised learning are the clustering problem algorithm such as k-mean, the algorithm of Apriority for association rule learning, and the dimensional reduction algorithm. Models are trained in unsupervised learning based on input data only, without these inputs being labeled i.e., the model is not given the ground truth label during training. It is the exact

opposite of supervised learning, where input data is divided into a group of elements that share the same attribute [32].

3. **Reinforcement Learning**

Reinforcement Learning is defined as action-based learning. An agent who, in a given situation, takes steps to optimize rewards. The agents are expected to determine the fastest possible route to obtain the reward. This type of algorithm does not require any data for learning. Instead, it requires a function that can calculate the reward. One of the most important reinforcement learning applications is the path exploration function, which is used in computer vision to locate a particular room or location. In video games, another application is to find the right movements to gain the game [33].

2.4.2 Activation Functions

The activation function is fundamental in transmitting information to a neural network to learn and handle any complicated tasks. There are various kinds of activation functions, some linear and others nonlinear, and the output value typically ranges between [0, 1] or [-1, 1]. In the hidden-layers, In the hidden-layers, the most widely used activation functions are the sigmoid function and the Rectified Linear Unit (ReLU) which are illustrated in Figure (2.3) [34]. ReLU is a non-linear function that substitutes all image pixels whose value is negative in the activation map with zero value. This leads to the improve the time complexity and reduces the occurrence of overfitting [35].

$$f(\text{x}) = \frac{1}{1 + e^{-x}} \qquad\qquad (2.4)$$

$$f(x) = \max(0.x) \qquad\qquad (2.5)$$
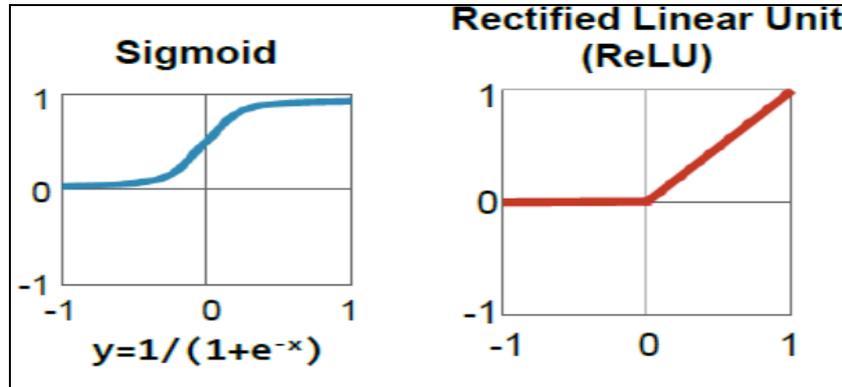
Figure (2.3) curve of the Transfer Function [35].

The output layer must make educated guesses about the likelihood of different classifications. As a result, the SoftMax activation function is the most frequently employed activation function in the output layer, as shown in the equation (2.6) [35].

$$Softmax(x_i) = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \; for \; j = 1, ..., K. \qquad\qquad (2.6)$$

Where K is the number of classes, z_j is the production corresponding to class j.

2.4.3 Loss Function

The Lost Function (also referred to as the Cost Function) measures the model ability to predict the presented data's right output. Several used "Loss Functions," in machine learning, such as the "mean squared error (MSE)" and Cross-Entropy [36]. The Cross-Entropy is the most specific Loss Function of Classification, since it determines the classification algorithm's performance depending on the probability of the class falls within the range [0, 1], and is described as following.

$$\text{loss} = -\sum_{i=1}^{n} y_i \log \hat{y}_i \qquad (2.7)$$

Where *n* represents the number of classes, *y* represents the true value, and $\hat{y}$ represents the predicted value [32].

2.4.4 Back-Propagation Algorithm

It is a Supervised Learning technique, that is frequently utilized for training Neural Networks (NNs), resulting in its efficiency and simplicity of construction. This algorithm makes use of the Gradient Descent method to minimize the mistakes in its predictions made by Neural Networks.

Starting with the initialization of all weights and biases in the network by assigning them a random number, the training process proceeds via two phases [37]. The network is trained using the weights and biases assigned to them:

1. **Forward Propagation**

   In forward propagation, the input signals (values) are used to calculate the output of the model, by applying the equations of (2.8) weights multiplication, and any equation used in activation function (equations 2.4, 2.5).

   $$Z_j = \sum_{i}^{N}(w_{i,j} \cdot x_i) + b_j \qquad (2.8)$$

   Where *Z* refer to the output, *W* refer to the wegiht, *X* is the input and *b* is the bias

2. **Backward propagation**

   It is the process of updating the weights, depending on the error value between the real and expected output (Cost Function). This is accomplished via the use of Adam algorithm.

An epoch is a period of time in which all the training dataset is passed forward, and then, the weights are updated. Epochs are repeated as necessary in order to optimize the weights.

Since there may be large dataset, and hence it is impractical to calculate error value for all the dataset at once, the dataset is divided into smaller random chunks called batch. The weights are updated for each batch. This is called iteration. So the iteration is the single weights updating [38].

2.4.5 Deep Learning

Deep Learning technique is a branch of Machine Learning that simulates complex abstract concepts in data by using a multi-layer architectural design, most commonly from neural networks, and non-linear transforms in its algorithms. The goal of using such techniques is to achieve "real" Artificial Intelligence, which means that a machine can learn how to perform extremely complex tasks in a way comparable to how the human brain works through layers of neurons. Computationally, construction and training Deep Learning are intensive. recent developments in applications based on general-purpose graphic processing units (GPUs), the rapid advancement of Machine Learning algorithms, processing of signals and information, and the growing amount of data that is used in training are all reasons that have increased the popularity and success of Deep Learning as show in figure (2.4) [39].
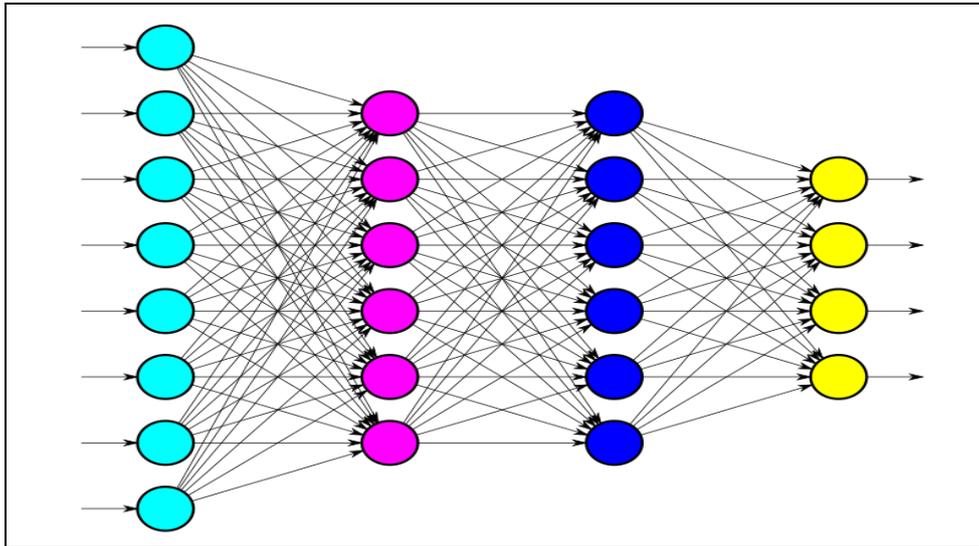
Figure (2.4) deep network architecture [39].

2.4.5.1 Deep Neural Networks (DNNs)

DNN is a feed-forward Artificial Neural Network that can be thought of as stacked neural networks or networks with many layers, this means that the Deep Neural Network between input and output layer has many hidden layer. Because DNN requires extensive computation times for execution and a large number of data inputs during training samples, the number of weights in these networks will be in the thousands or millions. Convolutional Neural Networks have outperformed all other DNNs in computer vision applications, particularly images categorization [40] [41].

2.4.5.2 Convolutional Neural Networks (CNNs)

CNN is a type of Deep Neural Network that is most popular employed in Computer Vision. It is similar to a Multi Layer Perceptron (MLP), except that it can mix many locally linked layers for extract the features with some fully connected layers for classification [42] as show in fiure (2.5). CNN is most commonly used with medical images due to its ability to process large amount of the datasets and no need to extract the features manually.

19
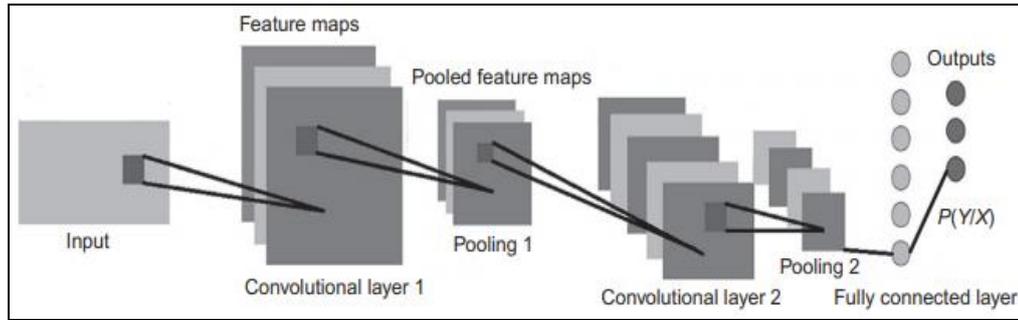
Figure (2.5) convolution Neural Network [42].

CNN's many components or core building pieces are briefly described as follows:

1. **Convolutional layer**

    It is the essential layer to build a CNN model. The primary goal of this layer is to extract the features from the original input image, which is achieved by the mathematical operation called "convolution" which refers to the merging of two functions to produce a new function. Convolutional layer has three matrices: the first is the input image, which is converted into a matrix (to be three-dimensional or grayscale two-dimensional), and the second is the filters matrix, also named kernel. The third matrix is the result of sliding the filter matrix across the input image in horizontal and vertical stages by using dot product. This third matrix is called Feature Map [43].

Figure (2.6) Convolutional operation on the image [43].

Multiple convolution filters are applied for a single input. The resulting Feature Maps are then combined to obtain the final result for a single Convolutional Layer, where this final result represents input data to the next layer. Each value of the filter matrix is a weight that is given by default. These values must be dif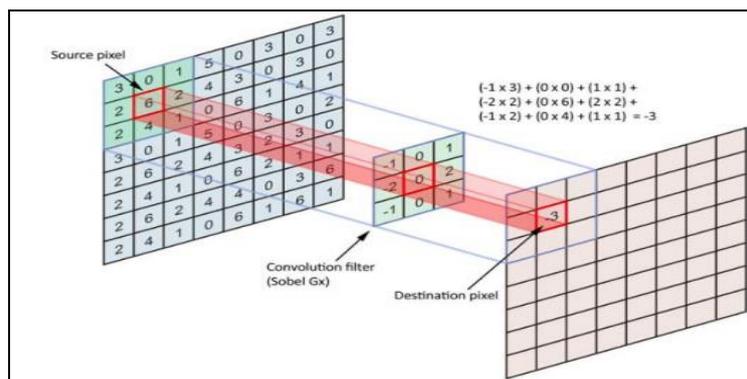ferent from one filter to another to give different characteristics or features to each matrix from the feature map matrices [44].

The convolution's outputs are sent into the non-linear Activation Function ReLU as shown in equation (2.5) [35].

Three hyper parameters influence the output volume: depth, stride, and padding [45]:

1. **Depth:** refers to the filters number employed in the convolution process, which is represented by the depth of the output volume. Each filter learns something new from the input, such as edges, blobs, and colors.

2. **Stride:** refers to the amount of steps the filter is slid onto the input image. When the value of the stride equal to one, the filter is moving one pixel every-time. On the other hand, if we slide a filters around the input image with stride of two, the filters are moves two pixels every time. This will result in reduced production volumes in terms of space.

3. **Padding:** Padding allows to regulatory the output size. When convolution is applied onto input, the size of the output is reduced, resulting in information loss. So padding adds zero values around the input matrix to avoid this. Valid Convolution and Same Convolution are two available options too. The Same Convolution implies that the output size remains the same as the input size. while Valid Convolution mean that there is no padding applied.

Below a formula is used to determine the output size of each Feature Map [45]:

$$output = \frac{(n+2p-f)}{s} + 1 \qquad (2.9)$$

n refer to the image size, whereas *p* refers to the padding, *f* refers to the filter size, and *s* is a stride.

2. Max Pooling Layer (subsampling or down sampling)

After Convolution Layers, CNNs frequently utilize the Pooling Layer. The technique of down sampling a collection of neighboring pixels into a single pixel is known as the Pooling Layer (also known as subsampling or down sampling). After the Convolution Layer, the pooling layer is usually used to decrease the size of the image activation maps. Pooling Layer have various types. Most common types are Max- Pooling and Average- Pooling. As illustrates in Figure (2.7), The Max-Pooling chooses a spatial area (sub-region) such as a 2x2 window and chooses the largest value from each window's corrected activation maps. Down sampling reduced the size of the activation maps image from 4×4 to 2 x 2. On the other hand, Average Pooling gives the average value for each part of the region. Overfitting may be solved by pooling layers, and Max-Pool has proved to be the most effective [46].
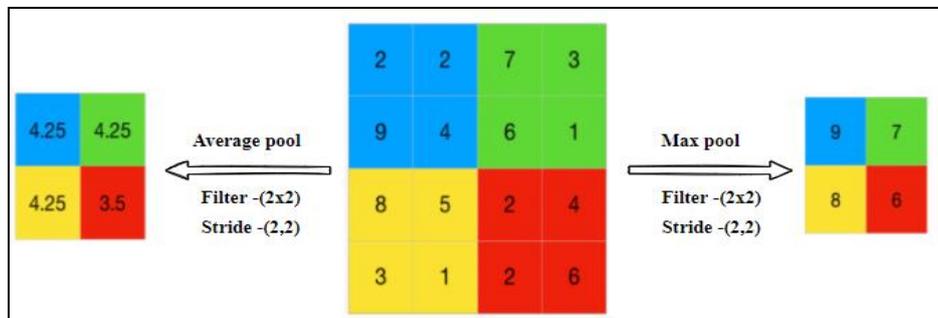

Figure (2.7) Max Pool with 2×2 kernel size and stride 2.

3. Fully Connected Layer

It is employed for Classification, that is similar to the work of the MLP where each node in the certain layer has a fully connected to all

the nodes in the next layer. It is Feed-Forward Artificial Neural Network (ANN). Fully Connected Layer takes Feature Maps as input from last pooling layer, which are flatten to be vector(1D) then enter to the Fully Connected Layer to construct the final output of the CNN model [47].

2.4.5.3   Optimization Algorithm (Adaptive moment estimation (Adam))

By adjusting the weights and bias values in the model, the optimizer can assist to minimize the loss function's output error. The Adam Optimizer is one of the most significant performance optimization techniques used in deep neural networks.

"Adaptive Moment Estimation" as Adam refers to it, is another tool that computes adaptive learning rates for each parameter and saves the average exponential decay of preceding square gradients while maintaining the average exponential decline of preceding gradients. Adam surpasses every other optimization method when it comes to rapid convergence and very complicated neural networks [48].

2.4.6 Regularization Techniques

Several techniques have been proposed to prevent the problem of overfitting called regularization.

- **Overfitting problem**

    Deep neural networks have a high capacity to represent learning. Lack of control over the learning process in deep neural networks can cause over-fitting problems. Over-fitting means that models have a poor generalization capability, that results in the poor predictability of test data although high output in training or validation data is achieved by the model. It happens when there is

23

a large gap between the error in training and the error in testing. Figure (2.8) provides an illustration of the overfitting problem. This state suggests that the deep neural network model is equipped to have a strong ability to match the train data rather than know the data patterns [49].
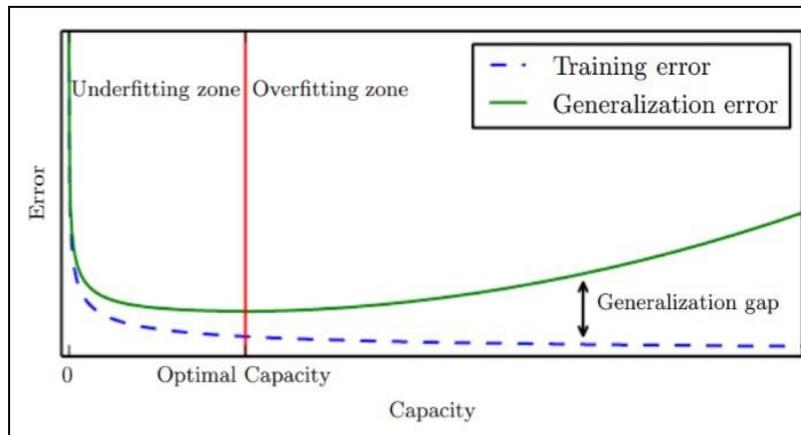
Figure (2.8) Overfitting in deep neural networks

There are plenty of regularization techniques available to improve deep generalization capability.

1. **Dropout**

    It is a regularization method for DNN models to deal with the issue of overfitting. The Dropout method is known to be a way of partially combining several different neural networks in an efficient way. It is clarified to drop out the neurons in the hidden layers of the neural network. Dropping out of neural neurons, means that neurons and their related incoming and outgoing neural network connections are immediately lost. The figure (2.9) depicts the concept of a dropout's life. When the dropout technique is applied to the multi-layer perceptron in the hidden layers, there is a likelihood that the neurons will be skipped in the hidden layer [50] [51].
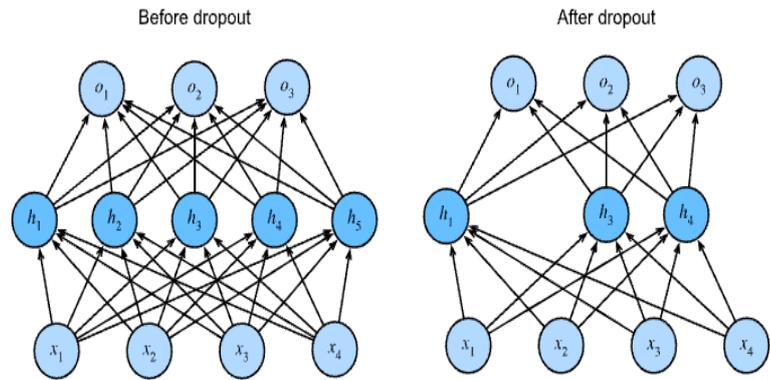
Figure (2.9) the dropout influence in a network.

2. Data Augmentation

The most efficient method that prevents overfitting for training the model on vast quantity of data with a variety of types. This could be accomplished by data augmentation, which entails employing a variety of approaches to make the dataset size large. More information on data augmentation strategies can be found in Section (2.3).

2.5   Performance Metrices

Four parameters are used to calculate the prediction error:

- True Positive (TP) refers to positive states that have been accurately classify as positive.
- False Positive (FP) is a term that refers to negative states that have been mislabeled as positive.
- True Negative (TN) is the correct way to classify a negative diagnosis.
- False Negative (FN) refers to situations that are mistakenly categorized as negative but are actually positive.

Several criteria were used to evaluate the classification algorithms' performance [52]:

1. The accuracy checks the number of correctly classified instances, whether positive or negative instances.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad (2.10)$$

2. Sensitivity is the rate at which positive samples are identified Correctly.

$$Sensitivity = \frac{TP}{TP + FN} \qquad (2.11)$$

3. Testing the proper true positive from the anticipated positives determines the accuracy of the performance of the model.

$$Precision = \frac{TP}{TP + FP} \qquad (2.12)$$

4. Specificity is the percentage of identification of negative examples correctly.

$$Specificity = \frac{TN}{TN + FP} \qquad (2.13)$$

5. For computing a balanced mean output, the F1-score displays a combination of accuracy and sensitivity.

$$F1\ score = 2\frac{Precision * Sensitivity}{Precision + Sensitivity} \qquad (2.14)$$

26

Chapter Three

The proposed system

3.1 Introduction

This chapter explains the methods and techniques used to build the proposed system to detect Covid-19 by using X-ray images. The proposed system will be arranged as follows:

The First stage, pre-processing was done on the data which begins with data augmentation which is used to increase the number of dataset, and prevent overfitting problem, converting X-ray images into grayscale images, the Contrast Limited Adaptive Histogram Equalization (CLAHE) technology was applied to enhance contrast and demonstrate the features, resize the image into (224 * 224) pixels and Normalization was employed to standardize the data.

The Second stage, the (CNN) based on augmentation technique was applied as a deep feature extraction technique and to identify X-ray features and classify the Covid-19 disease positive or negative using softmax function.

The final stage, the proposed system is tested in order to evaluate its accuracy and performance, the evaluation criteria such as *Accuracy, Specificity, precision, recall and F1-score,* were used as criteria to estimate the efficiency of the proposed model.

3.2 The Proposed Covid-19 Diagnosis System

The proposed system includes several steps to implement it such as preprocessing on the x-ray images, the CNN using for features extraction, and then classification by using the softmax. Moreover, evaluation the performance of the proposed system by using several criteria. Figure (3.1) shows the proposed system diagram.
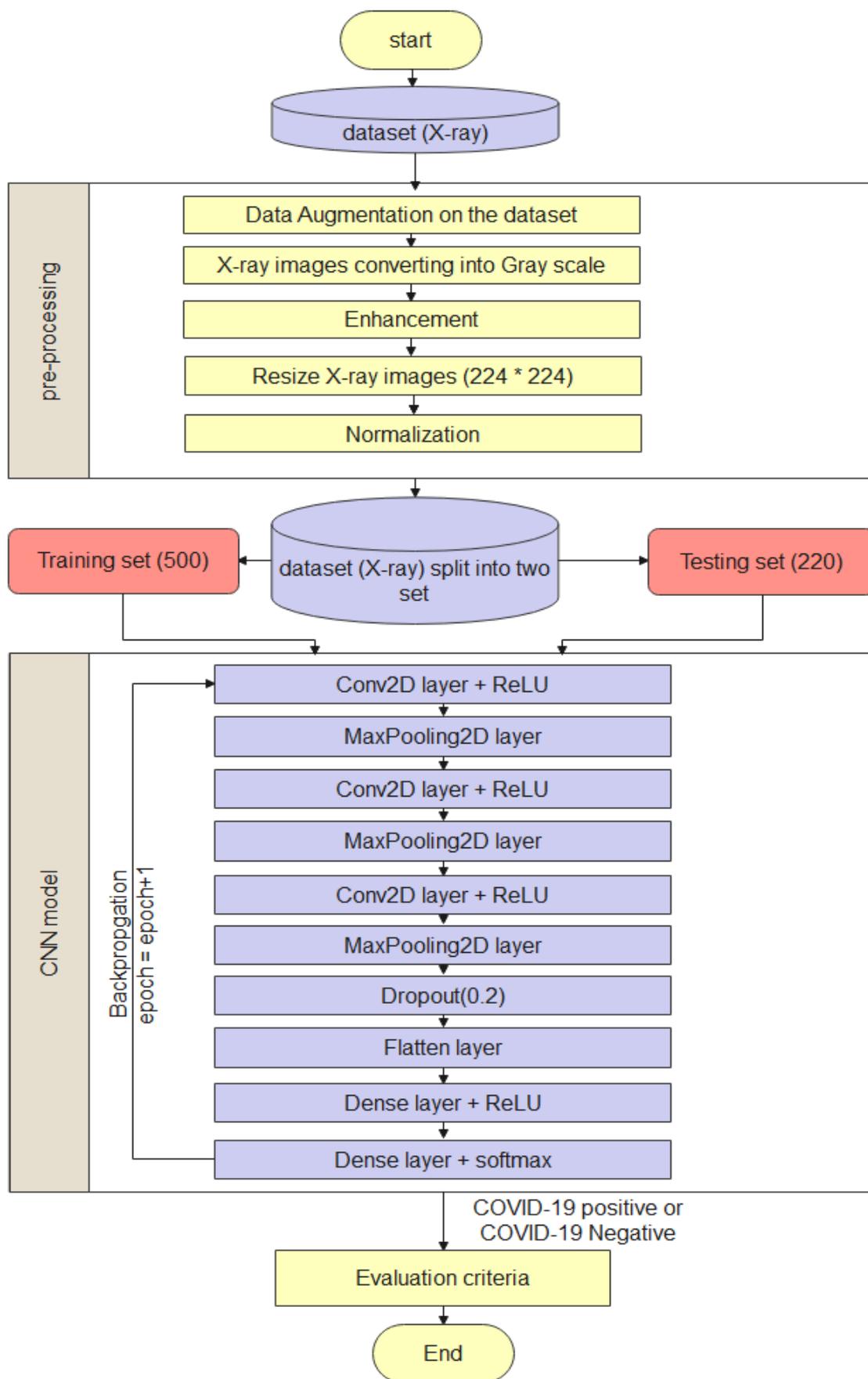
Figure (3.1) proposed system

3.3 Pre-Processing stage

The Preprocessing stage is important in the Covid-19 classification model. It has been performed on the X-ray dataset to make the dataset suitable for the training process and to initialize it for the extract the features. It consists of five steps: Data Augmentation method, convert X-ray images to Gray-Scale images, x-ray Images enhancement using CLAHE technology, resize X-ray images to (224*224) pixels, execute the normalization. As shown in Algorithm (3.1) general steps used in data pre-processing.
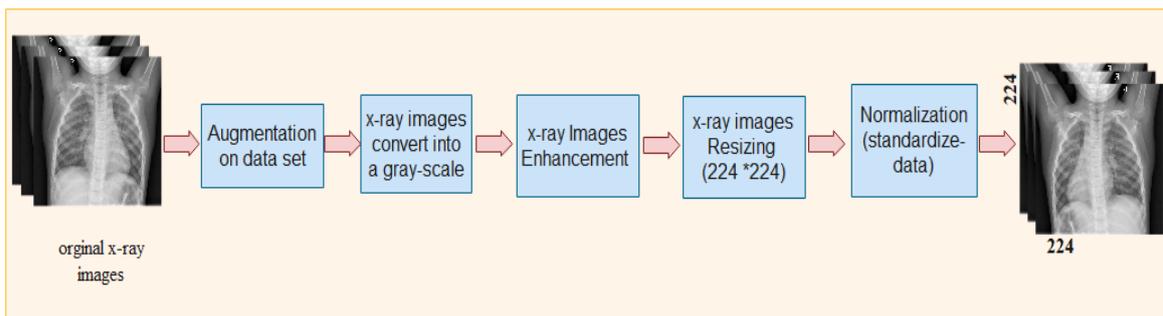
Figure (3.2) Steps of the preprocessing stage.

3.3.1 Data Augmentation

This method is used in the pre-processing stage to increase the proposed system's efficiency for precise evaluation. It increased the dataset by supplying more images to be used in the training and testing stages. Moreover, it will tackle the problem of overfitting. In the proposed system, the horizontal flip technique has been used in Data Augmentation.

3.3.2 Dataset Converting to Grayscale

Transforming X-ray image into a gray scale X-ray image would decrease processing time in the following step and simplify processing of the pixels. Input images of X-ray are transformed to a gray scale format. Meaning, instead of dealing with three channels in colored images (RGB),

the model will deal with only one channel (as shown in the equation (2.1) section (2.2.1)).

Algorithm(3.1) Pre-Processing
Input: *dataset*
Output: preprocessed_dataset

START
 	
    Step1: Data augmentation create new horizontally flipped image from dataset.
 	
    Step2: Convert X-ray Images to Gray-Scale Images.
 	
    Step3: Gray-Scale Images enhancement by applying CLAHE technology using the Equation (2.2).
 	
    Step4: Resize dataset into 224*224 pixels.
 	
    Step5: Normalize each pixel in dataset from [0-255] range into [0-1] range.
 	
STOP

3.3.3 X-ray Images Enhancement

In this step, x-ray image preprocessing performed using contrast limited adaptive histogram equalization (CLAHE) technique, as Clip Limit is equal to 2.0 and Tile Grid Size is equal to (8,8) are specified. (CLAHE) is an effective contrast enhancement method that effectively increases the contrast of the image, eliminates noise and blurring, to sharpen the features,

and showing the details of the image such as edge and boundary detection without changing the natural structure of the x-ray image.

3.3.4   Dataset resize

The CNN receives the images in the first layer, Where the original sizes of the images in the dataset are different from one image to another, and the sizes are unified in CNN layers to be suitable for more processing in the deep learning systems, and the resize images would benefit to decrease the computational cost and improve processing efficiency. Therefore, the images in the dataset have been resized into (224 * 224) pixels.

Besides, the selected size is considered a middle ground, that is not very large so it will not make the processing and training of the model slow, and not very small that will eliminate the important features of the images.

3.3.5   Data Normalization

It is a fundamental step in the images preprocessing. Due to the fact that the CNN receives and processes the images in the range of [0-1]. In order to achieve this result, each pixel is rescaled from the range [0-255] into [0-1] by dividing each pixel by the value of 255 (as shown in equation (2.2) section (2.2.2)).

3.4 Feature extraction stage

In order to extract the features from the images by the model, CNN uses self-learnable values of the filters used in feature extraction. The number and size of the filters are set by the developers of this system. However, the supervised learning process will set the values of the filters based on the training images. in the following an explanation of layers according to their type:

A. Convolutional Layers: each one of the three convolutional layers, contains a set of feature-maps which its number and size are notated in "output size" column in Table (3.1), each feature-map is a set of 2-dimension matrix of neurons when each neuron produced by summation and convoluting a one learnable kernel of weights with all feature-maps in the previous layer (or input image in the case of layer 1). This operation is used for features extraction. Each convolutional layer followed by Rectified Linear Unit (ReLU) as activation function.

B. Max-pooling layers: It is located between convolutional layers its main job is dimensionality reduction by reduce the size of a feature-map to quarter, with retention the most interesting information.

C. Vectorization and Fully Connected Layer: the last max-pooling layer output is a 3D matrix of 64 feature-maps each of them with 28×28 neuron. This 3D matrix is vectorized by column scan to produce a long vector with the length of $28 \times 28 \times 64 = 50176$ neuron, which it passes as input to a Fully Connected layer (FC1) which output 128 neurons and then passes as input to Fully Connected layer (FC2) which output 2, Each one to represent a single class (Covid-19 positive, and Covid-19 negative).

Table (3.1): The proposed structure of CNN, the input and output sizes columns are described in (rows × columns × feature maps). The kernel is specified as (rows×columns; stride).

No.	Layers	Input size	Output size	Kernel
1	Conv2D + Relu	(224, 224, 1)	(224, 224, 16)	3x3; 1
2	MaxPooling2D	(224, 224, 16)	(112, 112, 16)	2x2; 1
3	Conv2D + Relu	(112, 112, 16)	(112, 112, 32)	3x3; 1
4	MaxPooling2D	(112, 112, 32)	(56, 56, 32)	2x2; 1
5	Conv2D + Relu	(56, 56, 32)	(56, 56, 64)	3x3; 1
6	MaxPooling2D	(56, 56, 64)	(28, 28, 64)	2x2; 1
7	Dropout (0.2)	(28, 28, 64)	(28, 28, 64)	/
8	Flatten	(28, 28, 64)	50176	/
9	Dense + ReLU	50176	128	/
10	Dense + Softmax	128	2	/

3.5 Learning stage

For the diagnosis of Covid-19 infected cases, features of chest X-ray images are used to correctly classify the patients whether they refer to the infected state or not. According to the extracted attribute, classification is used to introduce diagnosing prediction (Covid-19 Positive, Covid-19 Negative).

Finally, achieving the diagnosis of Covid-19 in the output layer (Dense layer 2) is a fully connected layer have two nodes (outputs), Each one is responsible for giving a probability value for each of the classes (Covid-19 Positive, Covid-19 Negative). The output of the Dense layer 2 is passed into the Softmax activation function, which calculates the probability (through the forward-pass showed in chapter 2) for every

category depending on the input X-ray image. The probabilities are transmitted into the loss function equation (2.6) to calculate the error value, which will be applied to adjust/update the filters, weights, and bias values through the backpropagation process of training the proposed architecture.

In training CNN model, X-ray image classification uses the Adam optimization algorithm to update the weights, filters, and biases values, loss function (sparse categorical cross entropy) to evaluate the network, the used learning rate value is the default value of (0.001). And 15 epochs are used in the training stage, and 32 batch size. Algorithm (3.2) shows the steps of training CNN model.

The train is performed on the augmented data, the train begins with preprocessing as described in algorithm (3.1). The following step is to pass the X-ray images on the Convolutional Neural Network model in the forward direction which loss function will be calculated. And the backward direction will update the values of weights, biases, and filters to decrease the error between the predicted output of the CNN model and the actual label of the training sample. The output of a Convolutional Neural Network training algorithm is a trained set of weights and filters for all layers of the network architecture. These trained weights and filters are stored to be used by the network later in the testing stage.

Algorithm(3.2) training CNN model
Input: *Labeled_training_dataset, CNN_model, epochs, batch_size*
Output: Trained CNN model
START • *N* = number of images in *Labeled_training_dataset*

- *Iterations = N / batch_size*
- For *i*=1: *epochs*
  - For *j*=1: *Iterations*

- Pass the images into Convolutional layer - Pass the feature map from previous step into Max-pooling layer - Pass the feature map from previous step into Convolutional layer - Pass the feature map from previous step into Max-pooling layer - Pass the feature map from previous step into Convolutional layer - Pass the feature map from previous step into Max-pooling layer - Drop 20% randomly selected connections - Convert the multidimensional feature map into 1-D vector (Flatten layer) - Pass the vector from previous step into Dense layer - Pass the vector from previous step into Dense layer	Forward Pass
- Call equation (2.7) to calculate the Loss Function - Use Adam algorithm to update the weights, filters, and biases of *CNN_model*	Backward Pass

- End *j*
- End *i*

End

In the test stage the Convolutional Neural Network, the next step is to pass the X-ray images on the Convolutional Neural Network model in

the forward direction only to extract the features and then classify these images into Covid-19 Positive and Covid-19 Negative by using the trained weights in the fully connected layers and the trained filters in the convolution layers that were stored in the training phase and applied later in the test stage.

3.6 System Performance Measures

In this project, the accuracy, precision, Sensitivity, Specificity and F1 score are computed by the equations mentioned in section (2.5) to evaluate the effectiveness of the proposed system. figure (3.3) shows confusion matrix of classifier system.

Figure (3.3) Confusion Matrix.

Where True Positive (TP) represents the positive states that are correctly labeled as positive states, False Positive (FP) denotes the negative states that are incorrectly labeled as positive states, False Negative (FN) indicates the positive cases that are incorrectly classified as negative, and True Negative (TN) represents the right classification of negative diagnosis.

Chapter four

Results and discussions

4.1   Introduction

The obtained results from the system should be organized and compared to show and calculate its real performance for the evaluate. In this chapter, the results of the proposed system are described in different aspects and by using a variety of methods. For the proposed system, there are two main results: first one when the training and testing are carried out on the original dataset. And second one when the training and testing are carried out on the dataset after using Data Augmentation. Moreover, the results are compared with the results of other researches worked on the same dataset. The used evaluation metrics includes Accuracy, Sensitivity, Specificity and F1-score. Furthermore, some explanations and results translations are described. The system executing was carried out with the Python programming language version (3.9.0).

4.2   Hardware and Software Requirements

The proposed Covid-19 Classification System has been executed using personal computer HP with specifications such as Intel(R) Core i7-5500U @ 2.40GHz 2.40 GHz for CPU, 6.00 GB windows10 of RAM, and 64-bit Operating System.

The proposed system has been implemented by using python language (3.9.0) with the TensorFlow library. TensorFlow is an open - application used with python to easily execute the code of CNN. TensorFlow is provided by Google that focuses on efficient tensor research (tensor is the generalization of matrixes and vectors); since the execution of neural networks relies on matrix computation, TensorFlow supports fast execution speeds.

4.3   Dataset

X-ray images were gathered for the study of Dr. Joseph Cohen's online and open-source website [53]. In this dataset, a variety of lung diseases are collected. Theses diseases included Covid-19, "Acute Respiratory Distress Syndrome (ARDS)", "Severe Acute Respiratory Syndrome (SARS)", "Middle East Respiratory Syndrome (MERS)", besides normal lung images. Since this study is focused on Covid-19, only Covid-19 images are selected besides the normal healthy lung images. Which contains 360 X-ray images (180 COVID-19 Positive, and 180 COVID-19 Negative). This dataset is split into two categories: 70% training data (250 X-ray), and 30% test data (110 X-ray). Table 4.1 shows the number and percentage of each class (Covid-19 Negative and Positive) in each part of the dataset (train and test datasets).

Table (4.1) The number of images in training and testing datasets

	Covid-19 Positive	Covid-19 Negative	Total
Train dataset	125	125	250 (70%)
Test dataset	55	55	110 (30%)
Total	180	180	360 (100%)

The X-ray image data set described in table (4.1) is not sufficient for model development because systems that use deep learning need sufficient data for the training sample. So we will apply the Data Augmentation technique on the data set. DA is an effective and significant method for training any algorithm. Horizontal Flip type of DA technique will be implemented in this project (DA techniques were described in Section (2.3).

Table (4.2) The number of images in training and testing datasets after Data Augmentation

	Covid-19 Positive	Covid-19 Negative	Total
Train dataset	250	250	500 (70%)
Test dataset	110	110	220 (30%)
Total	360	360	720 (100%)

4.4   Result of x-ray images enhancement using CLAHE

The CLAHE operation is clearly made the features of the images very clear, without affecting the histogram or the balance of the images in general, as shown in figure (4.1).

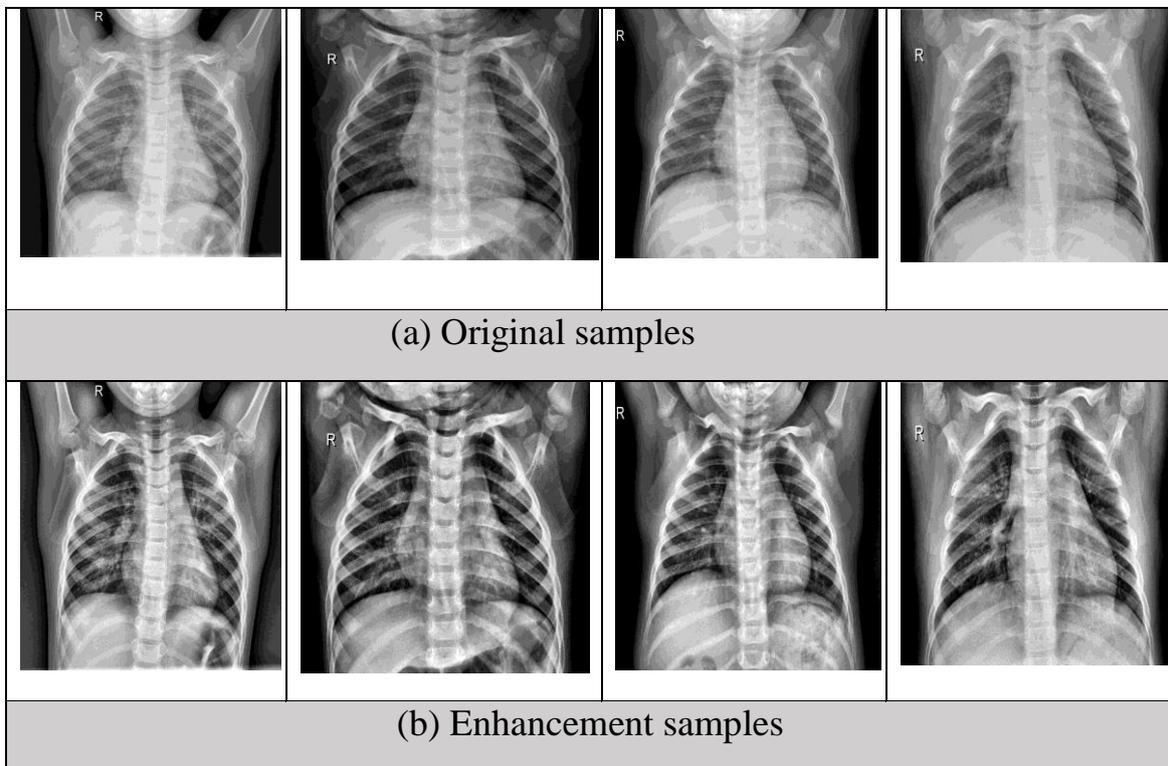(a) Original samples

(b) Enhancement samples

Figure (4.1) Enhancement on x-ray images.

4.5   Initial results of the proposed system

In order to evaluate the system performance, both the training and testing phases results are captured.

Figure (4.2) shown the accuracy and loss curves for the (training and testing) on the original data. Obviously, that there is a gap between training and testing accuracy data (as well between the data of the loss function). This may refer to the existing of overfitting problem, or the model have not trained well enough to distinguish between the different classes.
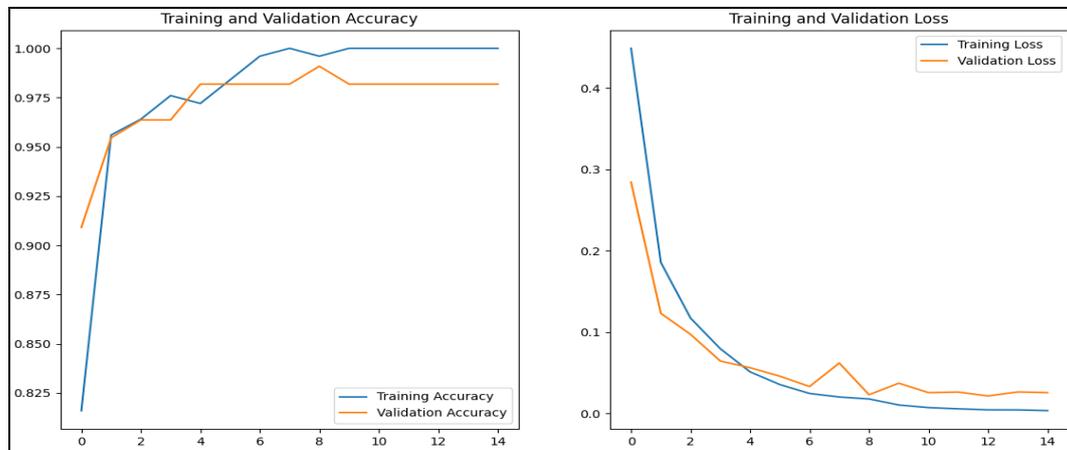
Figure (4.2) curve of the accuracy and the loss on the training and validation
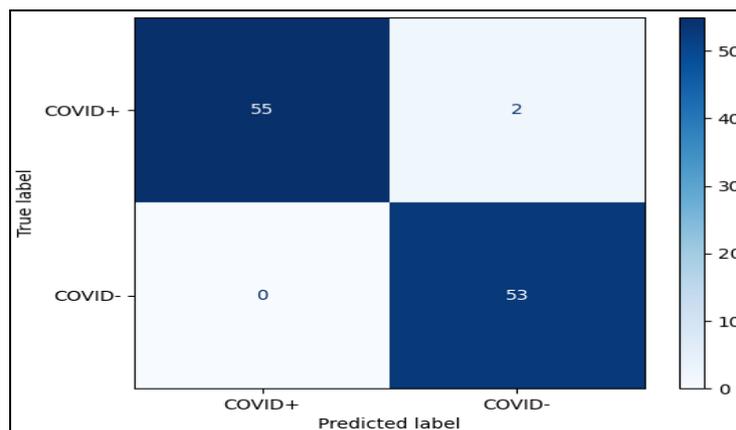

Figure (4.3) confusion matrix (original data)

4.6   Results of Data Augmentation

Since there is a big variation between the training and testing results (training accuracy was better than tetsing accuracy), a suspicion of overfitting is considered. Therefore, Data Augmentation methods are used in the proposed system, to increase the model performance. The Data

Augmentation Method Horizontal Flip is used to reducing overfitting. Figure (4.4) shows the effect of augmentation on a sample of the original images.
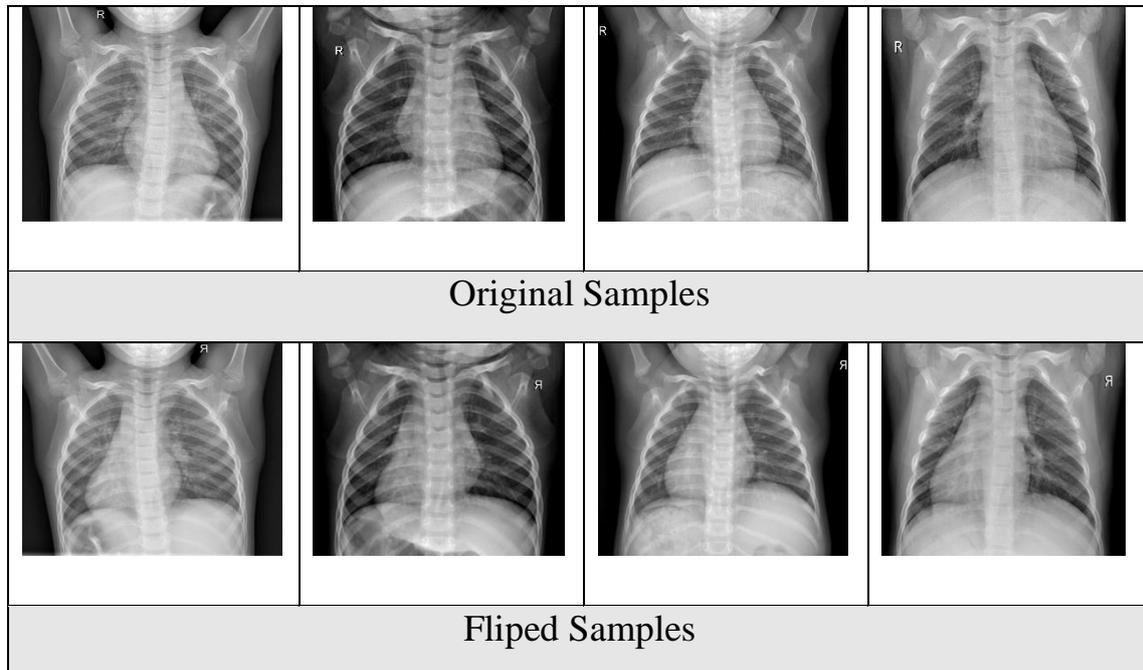
Figure (4.4) applied Augmentation method on X-ray images

4.7   Results of the proposed system after Data Augmentation

This section explains how the proposed system performed after using the Data Augmentation technique.

Figure (4.5) shows the accuracy and loss values for the training and testing dataset after using the Data Augmentation technique. Which could be noticed that there is no gap between the training and testing datasets. Which may refer to the elimination of overfitting problem.

Figure (4.6) shows the confusion matrix of testing dataset after Data Augmentation. Showing a greater result when applying Data Augmentation.
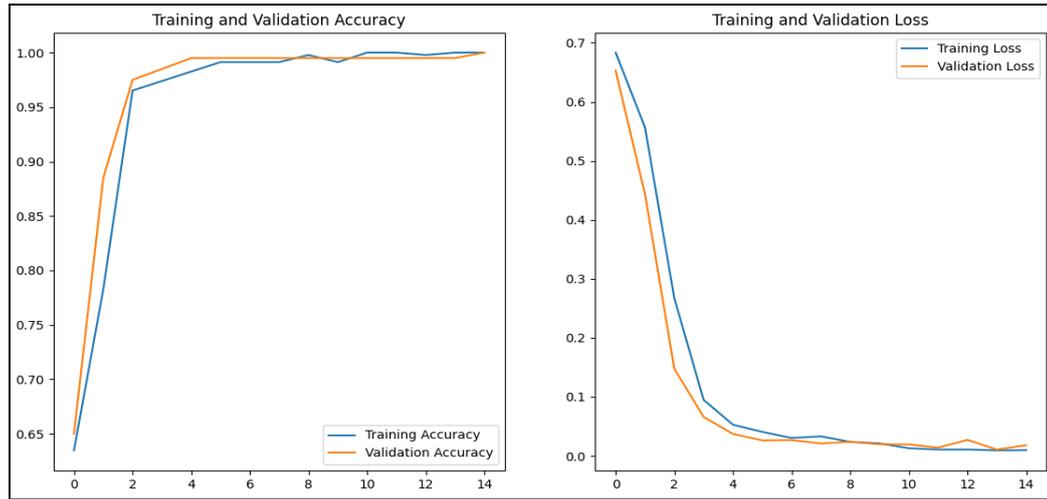
Figure (4.5) curve of the accuracy and the loss on the training and validation after Data Augmentation
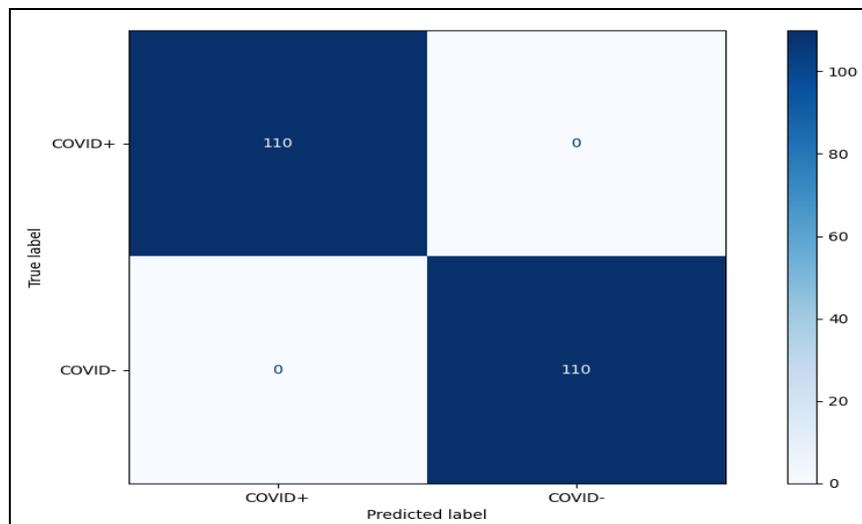
Figure (4.6) confusion matrix (augmented data)

4.8   Results comparison between original data and Data Augmentation

In this section comparison between the model training and performance on the original dataset, and dataset after using Data Augmentation technique.

Table (4.3) shows the accuracy, loss, and processing time when training the system on the original dataset, and when training it on the augmented

44

dataset. Obviously, we can see that the training performance for the augmented dataset is quite better than the original dataset. However, it takes almost double the time to train due to the increase in the number of images to be trained on.

Table (4.3) Model training performance on original and Data Augmentation dataset

| Data	Accuracy	Loss	Training time (sec)
Original data	98.18%	2.38%	65.54
Augmented data	100%	0.38%	108.54

Table (4.4) shows the evaluation criteria when training the CNN model on both the original and the augmentation dataset, by using the metrics of Accuracy, Sensitivity, Specificity, F1-score, and Area Under Curve (AUC).

Table (4.4) Evaluation criteria for model testing on original and augmented dataset

Data	Accuracy	Sensitivity	Specificity	F1-score
Original data	98%	96%	100%	98%
Augmented data	100%	100%	100%	100%

4.9   Comparison of the proposed system results with related works

Table (4.5) contains a comparison of various deep learning-based Covid-19 diagnostic methods with the proposed system performance that uses the same dataset. It should be noticed that the proposed system accomplished higher performance than the existing methods.

Table (4.5) Comparison of the proposed method with other research's

Research	Method used	Accuracy	Precision	Sensitivity	Specificity
Sahinbas, et al.[15]	VGG16	80%	80%	80%	—
Hemdan, et al.[13]	VGG19	90%	83%	100%	—
Monshi, et al. [54]	covidXrayNet	95.82%	96.93%	95.43%	—
Ozturk, et al. [55]	DarkCovidNet	98.08%	98.03%	95.13%	95.3%
Proposed System	CNN	100%	100%	100%	100%

The main reasons for obtaining these achieved results:

A. The design of the CNN architecture shown in Figure (3.1), in which identical blocks of layers were stacked to capture discriminative features and the dropout layer was used after last Max Pooling layer to reduce the overfitting and improve accuracy.

B. The overfitting problem, was prevented through the use of data augmentation technology and Dropout layer, which is the problem that most CNN models suffer.

C. The number of parameters obtained in the model reached (6,446,210), which is not much compared to the number of parameters of other literature models.

Chapter five

Conclusions and future work

5.1 Conclusion

The essential conclusions of the results obtained from the utilization of the suggested method for COVID-19 diagnosis in the X-ray images are as follows:

1. The preprocessing stage is a very important process where, convert X-ray images into Grayscale images, data augmentation technique, resize the images and standardize the data and enhancement the images all these make the work efficiently with CNN to gives higher accuracy results and reduce the processing time.

2. The proposed system used data augmentation methods. This system was implemented on these augmented X-ray images to reduce the issue of overfitting and to increase the efficiency of the proposed system as a high classification accuracy of 100% was obtained for the augmented X-ray images. The accuracy was 98% when applying the method on the original X-ray images.

3. The evaluation of proposed system on the augmented X-ray images: specificity, sensitivity, accuracy have been 100%, 100%, and 100% respectively.

4. The proposed system is fast, the X-ray images prediction time takes approximately (1.22) second for one image, which reduces the problems of handling more Covid-19 cases and this allows the system to be used in real-time.

5. The model achieved higher accuracy with a fewer number of parameters and requiring less computational power.

6. The process of extracting features from images of Covid-19 disease is more accurate, and this is reflected in the efficiency of the system.

5.2   Recommendations for future works

Inspired by this study, further developments could be considered in future works could be:

A. Developing the proposed system to include determining the stages of the disease, whether it is (Early, Medium, and Severe).

B. Using another deep learning technique instead of Convolution Neural Network.

C. Training and testing of the proposed system on another dataset, such as a CT scan.

References

References

[1]. Mackenzie, J. S., & Smith, D. W. (2020). COVID-19: a novel zoonotic disease caused by a coronavirus from China: what we know and what we don't. Microbiology Australia, 41(1), 45-50.

[2]. Ng, M. Y., Lee, E. Y., Yang, J., Yang, F., Li, X., Wang, H., ... & Kuo, M. D. (2020). Imaging profile of the COVID-19 infection: radiologic findings and literature review. Radiology: Cardiothoracic Imaging, 2(1), e200034.

[3]. Liu, H., Liu, F., Li, J., Zhang, T., Wang, D., & Lan, W. (2020). Clinical and CT imaging features of the COVID-19 pneumonia: Focus on pregnant women and children. Journal of infection, 80(5), e7-e13.

[4]. Chung, M., Bernheim, A., Mei, X., Zhang, N., Huang, M., Zeng, X., ... & Shan, H. (2020). CT imaging features of 2019 novel coronavirus (2019-nCoV). Radiology, 295(1), 202-207.

[5]. Chen Nanshan, Zhou Min, Dong Xuan, Qu Jieming, Gong Fengyun, Han Yang, Qiu Yang, Wang Jingli, Liu Ying, Yuan Wei. Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in wuhan, China: a descriptive study. Lancet. 2020;395(10223):507–513. [PMC free article] [PubMed] [Google Scholar]

[6]. Xu, X., Jiang, X., Ma, C., Du, P., Li, X., Lv, S., ... & Li, L. (2020). A deep learning system to screen novel coronavirus disease 2019 pneumonia. Engineering, 6(10), 1122-1129.

[7]. Amyar, A., Modzelewski, R., Li, H., & Ruan, S. (2020). Multi-task deep learning based CT imaging analysis for COVID-19 pneumonia: Classification and segmentation. Computers in Biology and Medicine, 126, 104037.

[8]. Apostolopoulos ID, Mpesiana TA (2020) Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. Phys Eng Sci Med 43:635–640

[9]. Zhang J, Xie Y, Li Y, Shen C, Xia Y (2020) COVID-19 screening on chest X-ray images using deep learning based anomaly detection. arXiv:2003.12338v1

[10]. Adhikari NCD (2020) Infection severity detection of CoVID19 from X-rays and CT scans using artificial intelligence. Int J Comput (IJC) 38(1):73–92

[11]. Alqudah AM, Qazan S, Alqudah A (2020) Automated systems for detection of COVID-19 using chest X-ray images and lightweight convolutional neural networks. https://doi.org/10.21203/rs.3.rs-24305/v1

[12]. Khan AI, Shah JL, Bhat MM (2020) Coronet: a deep neural network for detection and diagnosis of COVID-19 from chest X-ray images. Comput Methods Programs Biomed 196:105581

[13]. Hemdan, E. E. D., Shouman, M. A., & Karar, M. E. (2020). Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray images. arXiv preprint arXiv:2003.11055.

[14]. Apostolopoulos ID, Aznaouridis SI, Tzani MA (2020) 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 40:462–469

[15]. Sahinbas, K., & Catak, F. O. (2021). Transfer learning-based convolutional neural network for COVID-19 detection with X-ray images. In Data Science for COVID-19 (pp. 451-466). Academic Press.

[16]. Jamil M, Hussain I (2020) Automatic detection of COVID-19 infection from chest X-ray using deep learning. medRxiv. https://doi.org/10.1101/2020.05.10.20097063.
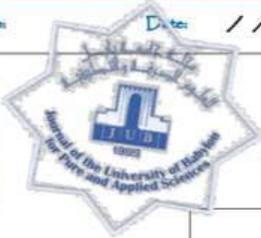
[17]. Saravanan, C. (2010, March). Color image to grayscale image conversion. In 2010 Second International Conference on Computer Engineering and Applications (Vol. 2, pp. 196-199). IEEE.

[18]. Rubini, C., and N. Pavithra. "Contrast enhancement of MRI images using AHE and CLAHE techniques." International Journal of Innovative Technology and Exploring Engineering 9.2 (2019): 2442-2445.

[19]. Siddhartha, Manu, and Avik Santra. "COVIDLite: A depth-wise separable deep neural network with white balance and CLAHE for detection of COVID-19." arXiv preprint arXiv:2006.13873 (2020).

[20]. Campos, Gabriel Fillipe Centini, et al. "Machine learning hyperparameter selection for contrast limited adaptive histogram Equalization." EURASIP Journal on Image and Video Processing 2019.1 (2019): 1-18. doi.org/10.1186/s13640-019-0445-4.

[21]. Duha Amir Sultan, and Alhan Anwar Yonis. "Contrast Enhancement in Gray Level Images". JOURNAL OF EDUCATION AND SCIENCE, 28, 2, 2019, 259-281. doi: 10.33899/edusj.2019.161214.

[22]. Koo, K. M., & Cha, E. Y. (2017). Image recognition performance enhancements using image normalization. Human-centric Computing and Information Sciences, 7(1), 1-11.

[23]. Yin, Z., Wan, B., Yuan, F., Xia, X., & Shi, J. (2017). A deep normalization and convolutional neural network for image smoke detection. Ieee Access, 5, 18429-18438.

[24]. Nisa, M., Shah, J. H., Kanwal, S., Raza, M., Khan, M. A., Damaševičius, R., & Blažauskas, T. (2020). Hybrid malware classification method using segmentation-based fractal texture analysis and deep convolution neural network features. Applied Sciences, 10(14), 4966.

[25]. Nayak, S. R., Nayak, D. R., Sinha, U., Arora, V., & Pachori, R. B. (2021). Application of deep learning techniques for detection of COVID-

19 cases using chest X-ray images: A comprehensive study. Biomedical Signal Processing and Control, 64, 102365.

[26]. Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. Journal of Big Data, 6(1), 1-48.

[27]. Heidari, J. (2019). Classifying Material Defects with Convolutional Neural Networks and Image Processing.

[28]. Pawara, P., Okafor, E., Schomaker, L., & Wiering, M. (2017, September). Data augmentation for plant classification. In International conference on advanced concepts for intelligent vision systems (pp. 615-626). Springer, Cham.

[29]. Narin, A., Kaya, C., & Pamuk, Z. (2021). Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. Pattern Analysis and Applications, 1-14.

[30]. Sarin,Sumit. https://towardsdatascience.com/exploring-image-data-augmentation -with-keras-and-tensorflow-a8162d89b844 (2019).

[31]. Shelag, Almaki Abdusalam Saad. "Classification of diseases on chest x-rays using deep learning". Diss. Near east university, 2018.

[32]. Kaliyugarasan, S. K. (2019). Deep transfer learning in medical imaging (Master's thesis, The University of Bergen).

[33]. Panov, A. I., Yakovlev, K. S., & Suvorov, R. (2018). Grid path planning with deep reinforcement learning: Preliminary results. Procedia computer science, 123, 347-353.

[34]. Wang, Y., Li, Y., Song, Y., & Rong, X. (2020). The influence of the activation function in a convolution neural network model of facial expression recognition. Applied Sciences, 10(5), 1897.

[35]. Wu, Liming. "Biomedical Image Segmentation and Object Detection Using Deep Convolutional Neural Networks". Diss. Purdue University Graduate School, 2019. doi.org/10.25394/PGS.8051702.v1.

[36]. Çalli, E. ""Master's Thesis Faster Convolutional Neural Networks." Radboud University, (2017).

[37]. Li, J., Cheng, J. H., Shi, J. Y., & Huang, F. (2012). Brief introduction of back propagation (BP) neural network algorithm and its improvement. In Advances in computer science and information engineering (pp. 553-558). Springer, Berlin, Heidelberg.

[38]. You, Y., Gitman, I., & Ginsburg, B. (2017). Large batch training of convolutional networks. arXiv preprint arXiv:1708.03888.

[39]. Mattsson, Niklas. "Classification Performance of Convolutional Neural Networks," Uppsala University, (2016).

[40]. Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A. R., Jaitly, N., ... & Kingsbury, B. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal processing magazine, 29(6), 82-97.

[41]. Das, H. S., & Roy, P. (2019). A deep dive into deep learning techniques for solving spoken language identification problems. In Intelligent Speech Signal Processing (pp. 81-100). Academic Press.

[42]. l-Waisy, A. S., Qahwaji, R., Ipson, S., Al-Fahdawi, S., & Nagem, T. A. (2018). A multi-biometric iris recognition system based on a deep learning approach. Pattern Analysis and Applications, 21(3), 783-802.

[43]. Dao, H. (2020). Image Classification Using Convolutional Neural Networks.

[44]. Xia, M., Li, T., Xu, L., Liu, L., & De Silva, C. W. (2017). Fault diagnosis for rotating machinery using multiple sensors and convolutional neural networks. IEEE/ASME transactions on mechatronics, 23(1), 101-110.

[45]. Bačanin Džakula, N. (2019). Convolutional Neural Network Layers and Architectures. In Sinteza 2019-International Scientific Conference

on Information Technology and Data Related Research (pp. 445-451). Singidunum University.

[46]. Gong, W., Chen, H., Zhang, Z., Zhang, M., Wang, R., Guan, C., & Wang, Q. (2019). A novel deep learning method for intelligent fault diagnosis of rotating machinery based on improved CNN-SVM and multichannel data fusion. Sensors, 19(7), 1693

[47]. Ghosh, A., Sufian, A., Sultana, F., Chakrabarti, A., & De, D. (2020). Fundamental concepts of convolutional neural network.

[48]. Abdul Aziz Saleh, M. B. W. (2018). Spoken Arabic digits recognition using deep learning/AbdulAziz Saleh Mahfoudh Ba Wazir (Doctoral dissertation, University of Malaya).

[49]. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112, p. 18). New York: springer

[50]. G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," Jul. 2012, [Online]. Available: http://arxiv.org/abs/1207.0580.

[51]. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1), 1929-1958.

[52]. Narin, A., Kaya, C., & Pamuk, Z. (2021). Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. Pattern Analysis and Applications, 1-14

[53]. Cohen, J. P., Morrison, P., Dao, L., Roth, K., Duong, T. Q., & Ghassemi, M. (2020). Covid-19 image data collection: Prospective predictions are the future. arXiv preprint arXiv:2006.11988.

[54]. Monshi, M. M. A., Poon, J., Chung, V., & Monshi, F. M. (2021). CovidXrayNet: Optimizing data augmentation and CNN

hyperparameters for improved COVID-19 detection from CXR. Computers in biology and medicine, 133, 104375.

[55]. Ozturk, T., Talo, M., Yildirim, E. A., Baloglu, U. B., Yildirim, O., & Acharya, U. R. (2020). Automated detection of COVID-19 cases using deep neural networks with X-ray images. Computers in biology and medicine, 121, 103792.

No: Date: / /2021   Receiving Date: 9/9/2021   Acceptance Date: 28/10/2021   التاريخ: ٢٨ /تشرين الاول /٢٠٢١   العدد: ٣١٤

شهادة قبول نشر

الــى:

السيدة	المدرس الدكتور
طيبة حسن هادي	علي يعقوب يوسف
جامعة بابل /كلية العلوم للبنات/قسم الحاسبات	جامعة بابل /كلية العلوم للبنات/قسم الحاسبات
teba.musa.gsci35@student.uobabylon.edu.iq	wsci.ali.yakoob@uobabylon.edu.iq

نوع وعنوان النتاج العلمي (Article):

Diagnosis of Covid-19 in X-ray images based on Convolutional Neural Network (CNN)

شكراً على إرسالكم نتاجكم العلمي إلى مجلتنا

مجلة جامعة بابل للعلوم الصرفة والتطبيقية

الرقم المعياري الالكتروني:٢٣١٢-٨١٣٥   الرقم المعياري الورقي: ١٩٩٢-٠٦٥٢

يسعدنا ابلاغكم بأنه تمت مراجعة نتاجكم العلمي وقبوله للنشر في **العدد ٣ المجلد ٢٩ للعام ٢٠٢١**. نرفق لكم مستندات التعديلات الأساسية المطلوبة والتي يجب تطبيقها على نتاجكم لاستكمال التصويبات قبل النشر. للمضي قدمًا في عملية النشر، يتطلب من جنابكم إرسال التالي:

١. طلب نشر البحث والتعهد: [عملًا الكترونيا مع الترانيع وحسب الاستمارة المرفقة بالبريد الإلكتروني].

٢. نتاجكم العلمي بصيغته النهائية بعد تعديلات المقيمين [برجى تضمين جميع التعديلات المبينة في الملف المرفق].

٣. ارسال نسخة من وصل رسم نشر البحث [٦ الف دينار عراقي].

٤. في الوقت الحالي، نود أيضًا تذكيركم بسياسات حقوق النشر والوصول المفتوح الخاصة بنا، يرجى الاطلاع على:

[https://www.journalofbabylon.com/index.php/JUBPAS/information/authors]

٥. سيتم استكمال الاستلال ببرنامج Turnitin في المجلة على ان لا تتجاوز النسبة ٢٠ %.

بمجرد نقل نتاجكم العلمي إلى عملية النشر، ستبقيكم هيئة التحرير على اطلاع بتقدم مقالتكم في عملية النشر. واذ نهنئكم بقبول هذا النتاج العلمي للنشر، نأمل ان يستمر تواصلكم معنا ورفدكم لمجلتنا بنتاج فكركم المتميز...

*** نتظر بقبول فائق الاحترام والتقدير ***

الأستاذ الدكتور
علي حسين المرزوكي
رئيس تحرير مجلة جامعة بابل للعلوم الصرفة والتطبيقية
/ ٢٠٢١

ACCEPTED

المستخلص

انتشر مرض فيروس كورونا الجديد ٢٠١٩ (Covid-19) بسرعة باعتباره متلازمة الضائقة التنفسية الحادة (ARDS) بين الأفراد في جميع أنحاء العالم. علاوة على ذلك ، فإن عدد مجموعات اختبار Covid-19 المتاحة في المستشفيات محدود مقارنة بالعدد المتزايد من الحالات كل يوم. لذلك ، من الضروري إدخال نظام الكشف التلقائي كطريقة تشخيص بديلة سريعة لتجنب انتشار Covid-19 بين الناس.

الغرض من هذا المشروع هو اقتراح طريقة آلية تعتمد على الشبكة العصبية التلافيفية (CNN) لتحديد المرضى المصابين بالالتهاب الرئوي Covid-19 باستخدام صور الأشعة السينية للصدر.

يتكون النظام المقترح من ثلاث مراحل. المرحلة الأولى هي مرحلة المعالجة المسبقة التي تبدأ بتقنية زيادة البيانات ، وتحويل صور الأشعة السينية إلى صورة بمقياس رمادي ، وتغيير الحجم ، وتم استخدام تقنية موازنة الرسم البياني المتكيف المحدود (CLAHE) للتحسين والتوحيد لتجميد البيانات. المرحلة الثانية ، تم استخدام شبكة CNN القائمة على صور الأشعة السينية لاستخراج الميزات. في المرحلة النهائية ، تم استخدام وظيفة SoftMax لتصنيف مرضى الالتهاب الرئوي Covid-19.

تم اختبار الطريقة المقترحة على مجموعة البيانات وأظهرت النتائج دقة عالية لتصنيف Covid19 ووصلت إلى (١٠٠٪) في مجموعة بيانات الاختبار (٢٢٠ صورة بالأشعة السينية). الطريقة المقترحة سريعة في التشخيص ويستغرق وقت التنبؤ بالصورة بالأشعة السينية حوالي (١) ثانية لكل صورة مع عدد صغير نسبيًا من المعلمات القابلة للتدريب (٦،٤٤٦،٢١٠).

جمهورية العراق

وزارة التعليم العالي والبحث العلمي

جامعة بابل-كلية العلوم البنات

قسم علوم الحاسبات

اكتشاف كوفيد-١٩ بناءً على صورة الأشعة السينية وتقنية التعلم العميق

مشروع مقدم الى

مجلس كلية العلوم للبنات-جامعة بابل

كجزء من متطلبات نيل درجة الدبلوم العالي في علوم الحاسبات

من قبل

طيبه حسن هادي

بإشراف

د.علي يعقوب السلطاني

٢٠٢١م   ١٤٤٣هـ