# Detection COVID-19 of CT-Scan Image for Hospitalized Iraqi Patients based on Deep Learning

**Dalia Shihab Ahmed**
Department of Computer Science, College of Science, Mustansiriya University, Baghdad, Iraq.
E-mail: dalia_shihab@uomustansiriyah.edu.iq

**Hanan Abed Alwally Abed Allah**
Department of Computer Science, College of Science, Mustansiriya University, Baghdad, Iraq.
E-mail: hanan.cs88cs@uomustansiriyah.edu.iq

**Samira Abdul-kader Hussain**
Department of Computer Science, College of Science, Mustansiriya University, Baghdad, Iraq.
E-mail: samiracs@uomustansiriyah.edu.iq

**Ishraq Khudhair Abbas**
Department of Computer Science, College of Science, Mustansiriya University, Baghdad, Iraq.
E-mail: Eshraqkhudair77@gmail.com

*Received August 20, 2021; Accepted December 02, 2021*
*ISSN: 1735-188X*
*DOI: 10.14704/WEB/V19I1/WEB19071*

## Abstract

Due to the conditions in which countries experienced the outbreak of the Coronavirus and our problem in diagnosing the disease, some of them relied on swabs to know if a person was infected, and also their dependence on symptoms such as temperature, rapid heartbeat, pressure, coughing and other symptoms similar to the normal flu, but this method is failure sometimes, therefore it was the best way for early detection and diagnosis of cases of COVID-19, as well as the accurate segregation of non-COVID-19 patients at cost and in the early stages of the disease, is a major difficulty in the current COVID-19 pandemic. Although widely used in diagnostic centres, radiation-based diagnostic techniques have drawbacks when it comes to disease newness. As a result, deep learning models are commonly used for X-ray interpretation by medical and computational researchers. Deep learning models can identify COVID-19, a critical task for treatment options based on diagnostic data these days. On the other hand, advances in artificial intelligence, machine learning, deep learning, and medical imaging methods enable outstanding performance, especially in detection, classification, and segmentation issues. These advances have allowed clinicians to more accurately monitor the human body, improving diagnosis and non-surgical patient examination. There are a variety of imaging methods that can be used to identify COVID-19, but we choose to use computerized tomography (CT) because it is the most commonly used. In addition, to detect COVID-19, we use a deep learning model based on a Convolutional Neural Network (CNN). Two samples of the tested data were used, where one of these data was collected from Al-Karkh Hospital in Baghdad, which consisted of 40 people, samples were taken according to their critical condition. The system was trained and tested on the basis of this dataset, where we used CNN three times, once to extract the feature and twice for the classification process. The results showed that the accuracy of the system reaches 100% because this system depends on the Bayes rule and it is not possible error.

## Keywords

COVID-19, CNN, CT Scan, AI, DL.

## Introduction

Recently, the globe has seen tremendous advancements in technology, which plays an essential role in industrialized nations. Nowadays, all aspects of everyday life, including education, business, marketing, military, communications, engineering, and health care, rely on modern technological applications. The health care center is a critical area that must heavily rely on modern technology, from identifying symptoms to precise diagnosis and computerized patient triage. Coronavirus-2 (SARSCoV-2) causes severe respiratory infections and respiratory problems, resulting in the new coronavirus disease 2019 (COVID-19) in humans, which was reported as the first case in December 2019 in Wuhan, China. Later, SARS-CoV-2 expanded globally and infected millions of people, prompting the World Health Organization (WHO) to declare the epidemic a global pandemic as the number of affected individuals continues to rise (Liao, J., et al., 2020). As of the 16th of December 2020, the total (global) coronavirus cases with recorded fatalities were about 73,806,583. COVID-19 is a virus that causes severe respiratory illnesses ranging from the ordinary cold to life-threatening infections such as severe acute respiratory syndrome (SARS) and Middle East Respiratory Syndrome (MERS) (MERS). Major symptoms of COVID-19, according to WHO reports, are similar to those of the ordinary flu: fever, fatigue, dry cough, shortness of breath, pains, and sore throat (Ghaderzadeh, M., & Asadi, F., 2021). The similarity of COVID-19 symptoms to flu symptoms makes early identification of the coronavirus challenging. The coronavirus, like other viruses and bacteria, has been shown to cause pneumonia in certain individuals, and the therapy for coronavirus-induced pneumonia differs from that for other kinds of pneumonia. Furthermore, individuals with bacterial pneumonia need antibiotic therapy, while those with viral pneumonia may be managed with intensive care (Kara, M. et al., 2021) As a result, precise and early identification of COVID-19-induced pneumonia is critical for saving human lives and halting the global pandemic epidemic. AI has made significant progress in several areas thanks to its machine learning (ML) basis. For example, a paid service that generates radiology reports using AI. Similarly, extensive AI research and use in medicine has been documented over many years (Rehman, A., et al., 2021). AI-based online or mobile apps for automated diagnosis may significantly assist doctors in minimizing mistakes, providing remote and inexpensive diagnosis in understaffed and underequipped regions, and improving healthcare speed and quality (Fauci, A.S., et al., 2020). In the case of COVID-19 radiography, ML techniques may be used to analyse CXR pictures in order to identify the aforementioned COVID-19 infection indicators as well as the negative consequences on the patients' lungs. This is especially important given how the epidemic pushed health systems to their breaking point, often to the point of collapse. Deep learning AI allows for the creation of end-to-end models that learn and find categorization patterns and features across many processing levels, eliminating the need to extract features manually. The rapid spread of the COVID-19 pandemic has required the development of novel approaches to meet the outbreak's increasing healthcare needs. To this aim, a slew of new COVID19 detection models have been suggested. The diagnostic paradigm in these techniques is mostly based on CXR and CT images (Velavan, T.P. & Meyer, C.G., 2020). Medical imaging, on the other hand, is gaining popularity in the computer aided analysis of pulmonary diseases. The detection of malignant nodules has been made possible by automated analysis of computed tomography (CT) images. In turn, radiographic analysis has shown to be effective in detecting TB symptoms (Mehanna, H., et al., 2020), as well as other cardiothoracic abnormalities (Harizi, I. et al., 2021).

## Literature Review

In this paragraph, the range of research that has been used COVID-19 detection and deep learning will be reviewed as shown below:

Introduce the differential privacy by design (dPbD) architecture and talk about how it fits into the federated machine learning system. Concentrate on the issue scenario of COVID-19 imaging data privacy for illness detection using computer vision and deep learning techniques. Address the assessment of the suggested architecture of federated machine learning systems, as well as how the differential privacy by design (dPbD) framework may improve data privacy in federated learning systems while still allowing for scalability and resilience. Suggest that scalable differentially private federated learning architecture is a viable approach for developing a secure, private, and collaborative machine learning model like the one needed to fight the COVID19 problem have been presented by author (Ulhaq, A., & Burmeister, O., 2020).

Covid-19 was detected using a categorization of Covid-19, pneumonia, and normal chest X-Rays. We examined the performance of five different Convolutional Pre-Trained Neural Network models (VGG19, VGG16, Xception, Resnet50, and InceptionV3). VGG19 and VGG16 exhibit accurate categorization performance. Both models can distinguish between three types of X-Rays with an accuracy of more than 92 % t. Another aspect of our research included utilizing Decision Tree Regress or to determine the effect of meteorological variables (temperature, humidity, solar hour, and wind speed) on the pandemic. Temperature, humidity, and sun-hour all had an 85.88 percent effect on Covid-19 escalation and a 91.89 % impact on mortality due to Covid-19, with humidity having an 8.09 % impact on death. We also attempted to forecast an individual's mortality due to COVID-19 based on age, gender, nation, and location using the Logistic Regression, which has a model accuracy of 94.40 % have been presented by author (Haque, A.K.M. et al., 2021).

A comparison of machine learning and soft computing models for forecasting the COVID-19 epidemic as an alternative to the susceptible–infected–recovered (SIR) and susceptible–exposed–infectious-removed (SEIR) models Two models out of a broad variety of machine learning models examined yielded promising results (i.e., multi-layered perceptron, MLP; and adaptive network-based fuzzy inference system, ANFIS). Based on the findings presented here, and given the extremely complex nature of the COVID-19 epidemic and the variability in its behaviour between countries, this research recommends machine learning as a viable method for modelling the outbreak have been presented by author (Ardabili, S.F. et al., 2020).

Conducted an experiment utilizing an adaptive genetic algorithm with fuzzy logic (AGAFL) model to predict heart illness, which aids practitioners in early diagnosis of the condition. They used the suggested model on the UCI heart disease dataset and discovered that it outperformed existing techniques have been presented by author (Reddy, G.T, et al, 2020).

Suggest n-COV-net is a deep learning neural network-based technique that may be utilized for identifying COVID-19 by studying X-rays of patients and looking for visual indications identified in COVID-19 patients' chest radiography images have been presented by author (Panwar, H., et. al, 2020).

To investigate COVID-19 in New York City, researchers used an unsupervised machine learning algorithm to identify commonalities across zip codes. They utilized feature selection and clustering methods to identify connections between the COVID-19 trends and mobility, socioeconomic, and demographic characteristics have been presented by author (Khmaissia, F. et al., 2020).

The suggested framework, dubbed the collaborative shared healthcare plan (CSHCP), is a generalized collaborative framework for assessing people's cognitive health and fitness. When compared to current research, the new framework offers promising results have been presented by author (Javed, A.R. et al, 2020).

Try to build a multi-classifier DL system using nine distinct CNN architectures and three different Majority Vote methods. Because the final choice is a composite of judgments included in each CNN design, the goal of this study is to optimize classification performance while minimizing mistakes. In this research, many CNN models are evaluated, including the Majority Vote model and Conventional CNN. According on the results of the 5 K-Fold test, the suggested model obtains an accuracy value of average F1-Score 0.992 and Accuracy 0.993. CNN's Soft Majority Vote approach is the best have been presented by author (Fibriani, I., et al., 2020).

Using a machine learning algorithm, nations with similar COVID-19 infection patterns may be grouped together. They collected data from COVID-19 cases from 155 nations using unsupervised machine learning methods (k-means), and utilized the K-mean clustering algorithm and principal component analysis (PCA) to categorize the countries have been presented by author (Carrillo-Larco, R.M., & Castillo-cara, M., 2020).

COVID-19 Behaviour was studied using an unsupervised machine learning model to categorize textual clinical reports into four groups. They generated features using term frequency/inverse document frequency (TF/IDF), bag of words (BOW), and report length, and then utilized these features in conventional machine learning methods to improve outcomes. They discovered that it improved testing accuracy have been presented (Khanday, A.M.U.D, et al., 2020).

Implementing CNN-tailored DNN machine learning algorithm to identify COVID-19 cases using patient's CXR or chest CT scan images, they discovered that the proposed model achieves overall high accuracy when compared to other models like ResNet, Inception V3 and MobileNet and have been presented by author (Mukherjee, H., et al, 2021).

Using Indian data to investigate the relationship between COVID-19 transmission rates and climatic factors using a gradient boosting model (GBM). After adjusting the parameters of the GBM model, it was optimized have been presented by author (Manavalan, R., 2020).

An incentive-based method to channel isolation is suggested, which aids those in need during these difficult times, as well as a blockchain-based solution to avoid data manipulation have been presented by author (Manoj, Mk, et al, 2020).

A method for categorizing and evaluating COVID-19 symptom predictions utilizing the Adaptive Neuro-Fuzzy Inference System (ANFIS), which aids in the early detection of Coronavirus Disease. The authors discovered that the support vector machine (SVM) method outperforms all other classifiers in terms of prediction accuracy have been presented by author (Ardabili, S.F., et al., 2020).

A machine learning algorithm to identify hidden similarities amongst COVID-19 cases in order to forecast the risk of infection They utilized their model to determine the main parameter that was used to discover hidden patterns between instances (dimensionality reduction), then used the unbiased hierarchical Bayesian estimator to apply their model have been presented by author (Iwendi, C., et al 2021).

Comparing the most prominent deep learning-based feature extraction frameworks, such as VGGNet, ResNet, NASNet, DenseNet, and NASNet, on COVID-19 chest X-rays patients to aid in COVID-19 automated identification. They discovered that the DenseNet121 feature extractor combined with the Bagging tree classifier produced the greatest results have been presented by author (Kassania, S.H., et al, 2021).

Providing an overview of deep learning applications utilized in healthcare over the past decade and reviewing state-of-the-art research efforts connected to COVID-19 medical image processing deep learning applications. Finally, they addressed the difficulties of using deep learning in COVID-19 medical image processing have been presented by author (Bhattacharya Sweta, et al. 2021).

## Deep Learning Classification

Artificial neural networks is algorithms inspired by the structure of the human brain, learn from huge quantities of data in deep learning, a subset of machine learning. Artificial intelligence refers to when machines may perform tasks that would normally require human intelligence. It involves machine learning, which allows computers to learn from experience and develop skills without the need for human intervention. Similar to how to learn from experience, the deep learning algorithm will perform a task repeatedly every time it is modified slightly to improve the outcome. Referred to as “deep learning” were because neural networks have multiple (deep) layers that enable learning. Any problem that requires "thinking" is one that deep learning can learn to solve. Deep learning allows machines to solve complex problems even when using a very diverse, unstructured, and interconnected data set. The more learning algorithms, the better they perform (Wang, T., Huan, J., & Li, B., 2018). As shown in Figure (1).

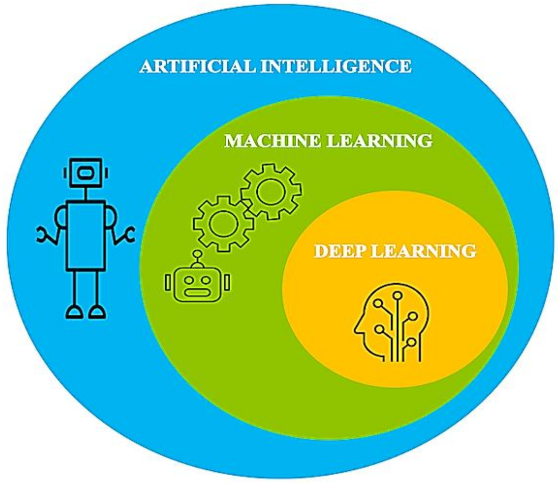


**Figure 1 AI self-learning**

## Convolutional Neural Network (CNN)

In computer vision problems, CNN is the most widely used, CNN is a special type of forwarding Feeding neural network, as it is a solution to many problems of computer vision and artificial intelligence. It is called "convolutional neural networks" because convolution is based on at least one of its layers as a basic stage. An input and output layer, as well as numerous hidden layers between them, make up a CNN. Convolutional layers, pooling layers, and completely linked layers are all examples of layers. The number and kind of layers employed in CNN designs vary depending on the application (Lowe, D.G. et al., 2004). CNN layers can have anywhere from a few to hundreds of filters that comb through the input and evaluate all channels.

CNN sequential layers extract ever more abstract features. Beginning with edges and corners in the first layers and progressing to whole faces and artifacts in the deeper layers. During training, the network discovers for itself which features should be extracted to solve the problem. CNN has the advantage of not requiring advanced experience or human effort in feature design, making them a flexible tool for a variety of computer vision problems (Kattenborn, T., et al., 2021). Figure (2) shows the architecture of Convolutional Neural Networks (CNN).

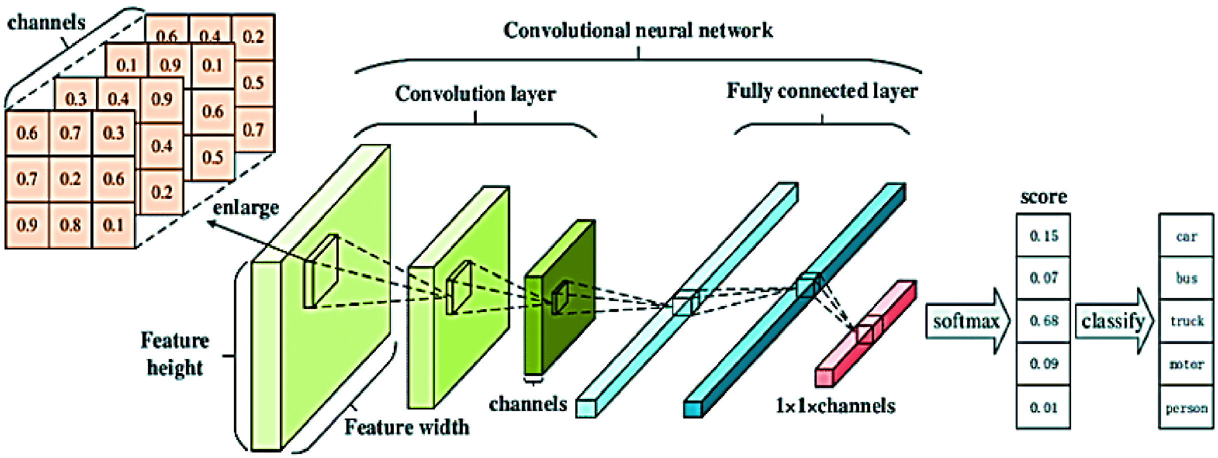


**Figure 2 Architecture of CNN**

Many of the terms and terminologies associated with CNN are explained in the subsections below:

## I. CNN Input

In several cases, CNN is used to image processing. In computer systems, each image is represented by a matrix of pixel values. The width and height of the gray image can be used to calculate the size of the input image. Also, the size of a colored image has a third dimension (depth), which refers to the number of input image channels RGB (Red, Green, and Blue channels); as a result, the image can be described by three matrices (Zhang, Q.,et al.2019) .Figure (3) illustrates RGB images.

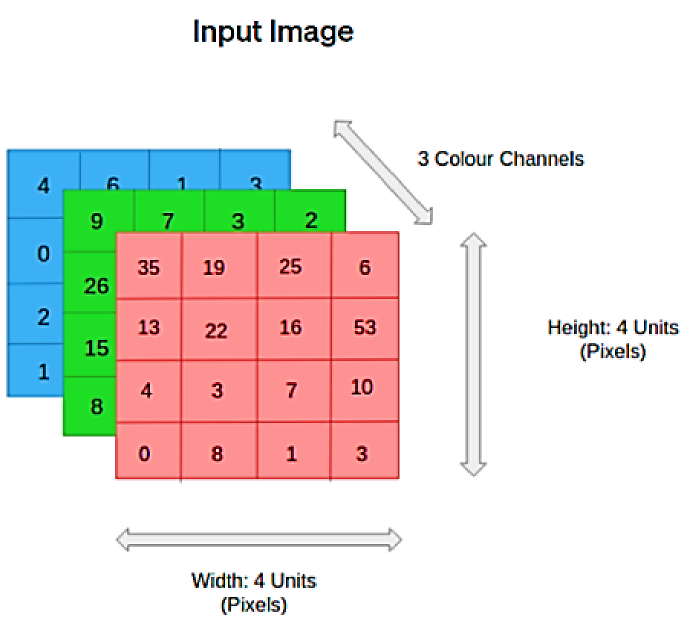


**Figure 3 RGB Image**

## II. Convolution Operation and Filter

A filter, also known as a kernel, is a small matrix of real entries. It's smaller than the original image. The filter is usually of order 1×1, 3×3, 5×5, and 7×7.These filters correspond to the neural network's weights, and they are tuned during the training phase. Convolution, in mathematical terms, is an operation to combine two given functions by integration which shows how the shape of one function is modified by the other. Figure (4) shows the Convolution operation.

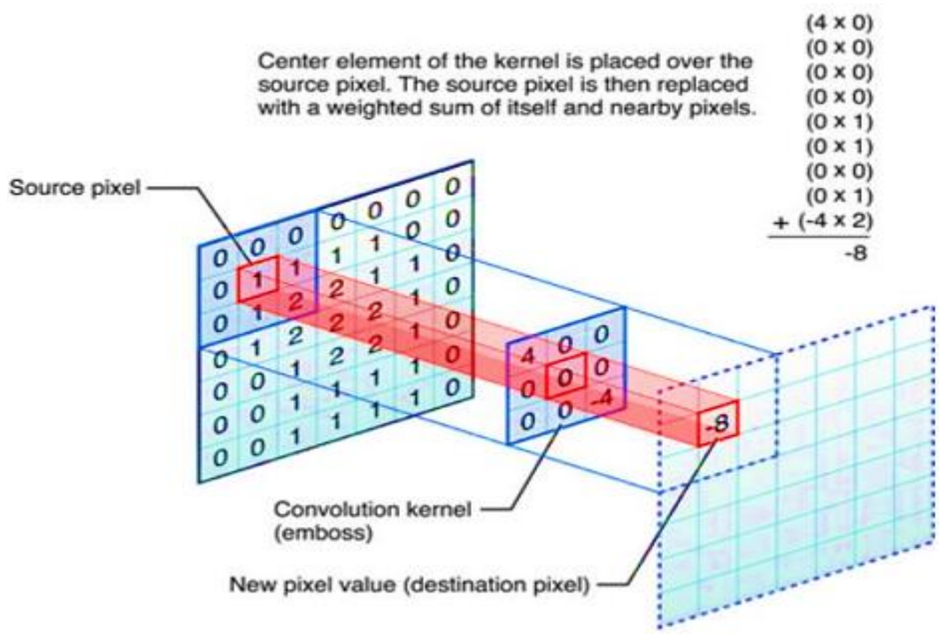


**Figure 4 Convolution operation**

The kernel is a move from the left to the right of the image, and the convolutional process begins at the top-left corner of the image. When the kernel reaches the image's top-right corner, it moves one object downward before moving from left to right again. This procedure is repeated until the kernel reaches the bottom-right corner of the input image. The feature map is the output of this operation (Williams, A.A., 2019). The filter's movement through the image is determined by a value called stride. For example, if the stride is set to 1, the filter will only travel one pixel across the image.

## III. Image Features

Edges, lines, and interest points in images provide a variety of details about the image's content. They describe local regions in an image and are used in a variety of image analysis applications such as reconstruction, identification, matching, and so on (Balodi, T. 2021).

## IV. Zero Padding

Since the image size will be shirked on any convolution step, zero paddings is a process of adding zeros to the matrix of the input image to allow for the preservation of the original input size. Padding comes in two types. The first is referred to as valid, which means there is no padding and the convolutional layer is never pad at all, so the input size would not be maintained. The second type is known as the same, and it means that the initial input image is padded until it is convolved. As a result, the output size is identical to the input size. Figure (5) illustrates Zero padding. Briefly, Padding can be seen in our input volume; padding is required to make our kernels fit the input matrices. We sometimes do zero paddings, which requires adding one row or column to each side of zero matrices, or we can cut out the part of the image that does not fit, also known as valid padding (Huang, K., et al., 2019).

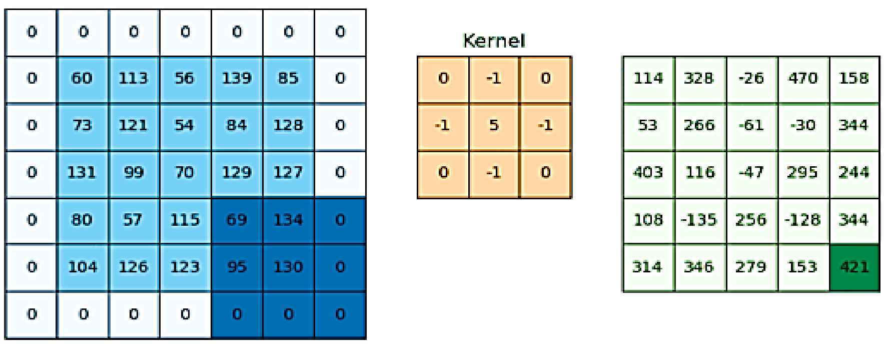


**Figure 6 Zero Pagging**

## V. Epoch, Batch Size, and Iterations

The use of epochs, batch sizes, and iterations becomes necessary when the data is too large, which is a common occurrence in machine learning. To solve this problem, we must break the data into smaller sizes and feed it to our model one at a time, updating the weights of the neural networks at the end of each phase to match it to the data. An epoch is a period of measurement that denotes how long the network will continue to process data. One epoch means that the entire dataset was used to train the network for a single pass through the network. Every epoch, the batch size determines how many pairs of input/output the network is presented with. While the iterations are the numbers of batches needed to complete one epoch (Kundu, R., et al., 2021).

## VI. Cost Function

The cost function is used to provide feedback about how well the network is performing. This is the function that the network is attempting to reduce, and it is the product of the deep learning process. An optimizer is needed to minimize the network's cost function. The term ADAM Optimizer refers to the (Adaptive Moment Estimation Optimizer), which is a common option in this case. Instead of using the conventional stochastic gradient descent method, ADAM is an optimization algorithm that may be used to update network weights repeatedly depending on training data (Liu, R. W., et al 2021).

## Layers of CNN

Layers of CNN There are many layers in a convolution neural network. The most important layers are explained in the subsections below:

### a) Convolution Layer

The main aim of the convolution process is to extract features from the input image so that the spatial relationship between pixels in the image is maintained by learning image properties using a filter (kernel) on the input data. The convolution neural Network's building block is the convolution layer. It performs a convolution operation on the input image. It's used to extract the input image's features. The first convolution layer extracts low-level features like lines, corners, and edges. Next-level layers extract higher-level features from the input image. The equation (1) present the layers equation:

$$a^1 = \sigma(b + w * a^0) \tag{1}$$

Where $a^1$ denotes the set activations output from the feature map, while $a^0$ denotes the input activations, σ denotes for activation function, w denotes for weight, b denotes for bias and * is the convolution operation. The kernel's height and width are smaller than the input image's height and width. To build a feature map, the kernel slides over the image (convolve with). The sum of the product by the kernel element and the original image is called convolution.

### b) Subsampling Layers

This method called also known as pooling layers or downsampling, is responsible for reducing the spatial size of convolutional layers that produce feature maps. The features' resolution is reduced by the pooling layer. Pooling can be done in two ways: maximum and average. The input image is divided into non-overlapping two dimensional spaces in both forms. We can deduce from the names that max-pooling extracts the maximum value from the filter and average pooling extracts the average value. Pooling is used to minimize dimensionality. Figure (7) shows the max and average pooling. So when we pool an image, we're not taking out all the values; instead, we're taking a summary of all the values present.

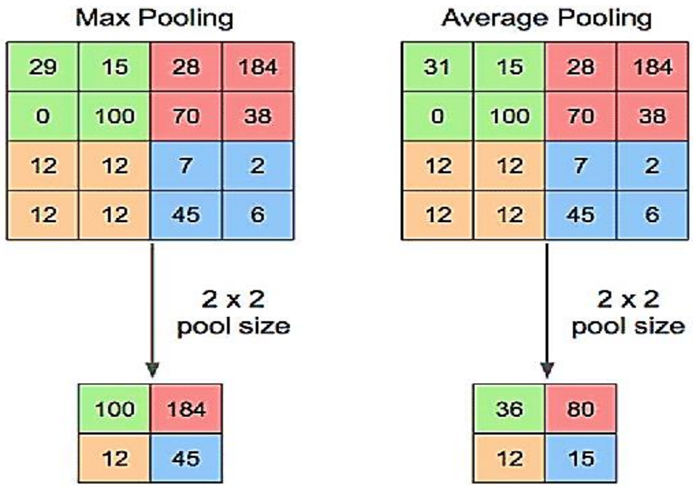


**Figure 7 Max or average pooling**

## c) Fully Connected Layer

The Fully Connected (FC) layer, which connects the neurons between various layers, includes the weights and biases as well as the neurons. Before the output layer, these layers make up the last few of a CNN's architecture. This flattens and feeds the FC layer the input image from the previous layers. Further FC layers are applied on top of the flattened vector to perform mathematical function operations. The categorization procedure gets under way at this point.

## d) Dropout

Overfitting in the training dataset is common when all features are linked to the FC layer. Using a model that performed well on the training data and now performs poorly on fresh data is called overfitting. Dropout layers, which remove certain neurons from the neural network during training to decrease the model's size, are used to solve this issue. The neural network loses 30% of its nodes when it reaches a dropout of 0.3.

## e) Activation Function Layer

When an image has been completely convolved with a filter from a previous layer, the output is passed via an activation function. The activation function is used to give the network non-linear properties, allowing it to solve more complex non-linear problems than its linear counterparts. The Rectified Linear Unit (ReLU) is the most commonly used activation function in CNNs. The activation functions of neurons must be non-linear for the multi-layer network to model complex non-linear functions. Since the output of a network with only linear activations is simply the linear combination of the input, this is the case. It also means that several layers can be merged into a single one, Figure (8) shows the ReLU function, and it is defined as the equation (2) [36]:

$$R\,(z) \;=\max\,(z,0) \qquad (2)$$

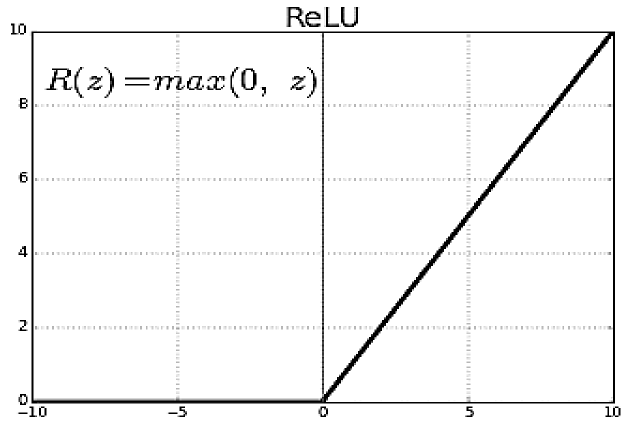


**Figure 8 Activation fuction**

## The Proposed Method

The proposed system includes the discovery and classification of infected persons using (CNN) where the proposed system consists of a dataset that has been collected, this dataset needs to process preprocessing and then enter (CNN) three times once to extract the learned characteristics and twice for the classification process shown in Figure (9).

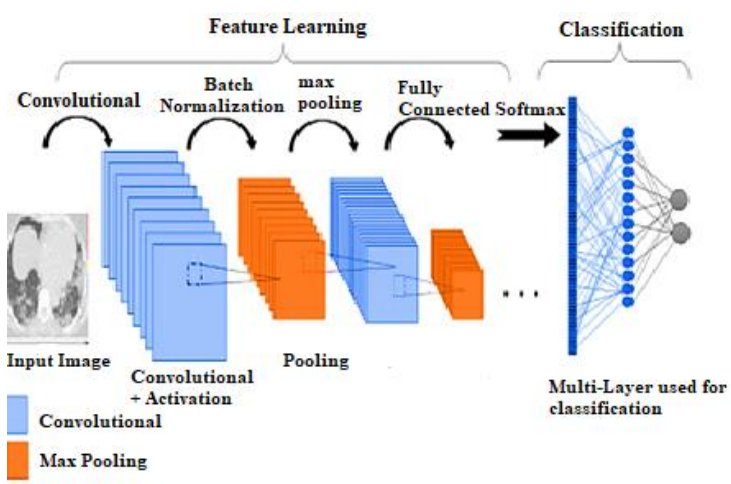


**Figure 9 The Proposed Method of covid -19 detection**

### A. Image Acquisition

In order to perform their assigned task, vision systems always begin with picture acquisition. Once acquired, there are a variety of processing techniques that may be utilized to carry out a variety of image-related activities. The reason why picture capture is always the first step in the workflow sequence is because processing is impossible if there is no image. Images may be obtained in a variety of ways, including the use of cameras or scanners. The captured picture must maintain all of its characteristics. In this research, two types of data were used, one of which was downloaded from the Google Standard website and the other data collected from an Iraqi hospital for people with and without Covid-19 disease.

### 1. Standered Dataset

A total of 2482 CT scans were included in the SARS-CoV-2 CT scan dataset, which included 1252 CT scans positive for SARS-CoV-2 infection (COVID-19) and 1230 CT scans for patients who were not infected with SARS-CoV-2. The data was collected from real patients in Sao Paulo, Brazil, hospitals. The aim of this dataset is to encourage research and development of artificial intelligence methods that can analyze a person's CT scans to identify whether he or she is infected with SARS-CoV-2.
www.kaggle.com/plameneduardo/sarscov2-ctscan-dataset.

This dataset is accessible at www.kaggle.com/plameneduardo/sarscov2-ctscan-dataset. For training and testing, a total of 2481 CT-scan pictures were utilized. There are 1252 CT scans of COVID-19-positive patients and 1229 CT scans of COVID-19-negative patients in the SAR-CoV-2 CT-scan dataset. A sample of lung CT scan images from the dataset is shown in Figure (10).

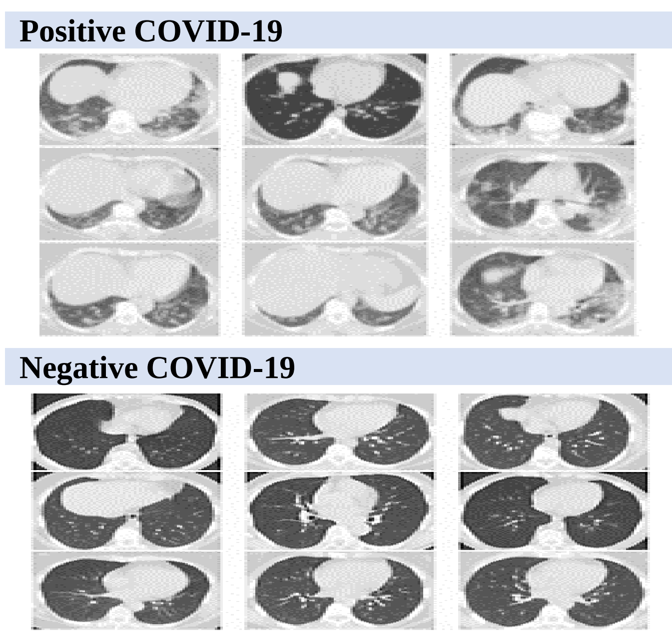


**Figure 10 Sample of CT scan datasets**

## 2. Real COVID-19 Dataset

In this research, the system was tested on a real dataset, where samples were taken from Al-Karkh Hospital in Baghdad for some people infected or un infected, where 40 samples were taken for different people, each person for whom the cross-sectional image was taken according to the critical condition reached, and the range between 3-8 CT scan images was shown. Figure (11), (12) shown samples of real covid 19 dataset.

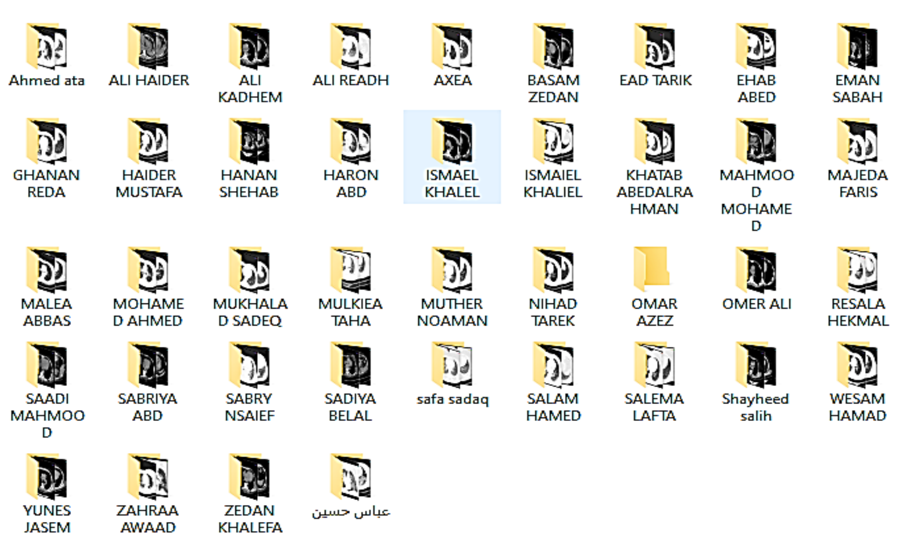


**Figure 11 Collection Dataset**

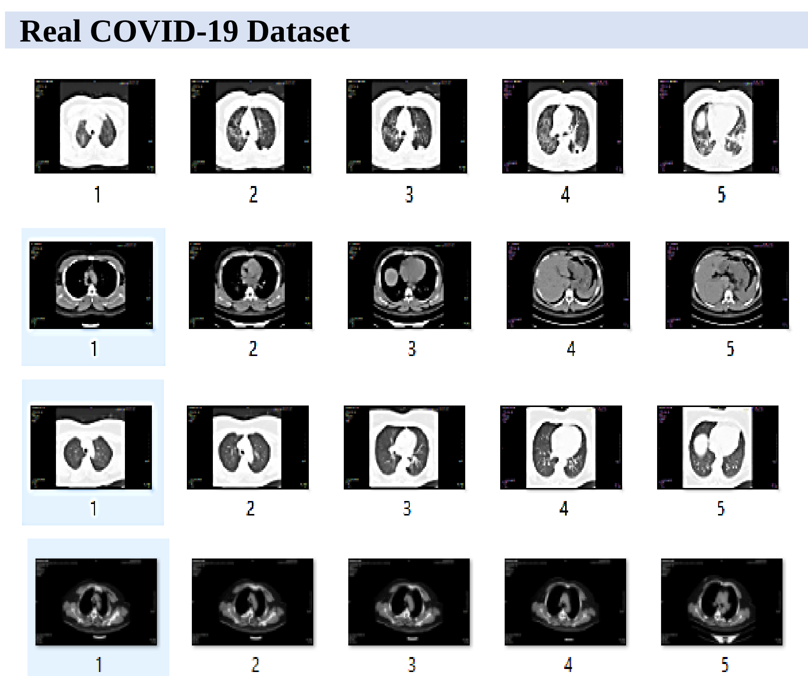


**Figure 12 Sample of ral dataset**

## B. Pre-processing Stage

Although the CT scan images are rarely noisy, it is possible that the process of collecting the dataset may lead to reduce the details of the image, so we need to process the images using one of the filters. Where bilateral filter was employed to enhance the picture and remove noise while maintaining image accuracy and edges. This enhanced Gaussian technique was employed, in which the filters were improved by doubling the usage of another Gaussian filter owing to numerous iterations and varied pixel density, i.e. pixels with a density comparable to the ones in the centre are only included to compute an intense density value. As a consequence, the covid 19 CT scan image edges are preserved using this method, since neighboring pixels put on the opposite side of the edge for pixels near to the edges and therefore significant variations in density look blurry when compared to the centre pixel.

## C. Feature Extraction

The extraction of features from samples such as COVID19 CT scan images is the most critical stage in the identification of the person infected or not for Covid 19. The system, in its most basic form, employs a convolutional neural network (CNN), with the convolutional layer serving as the eight feature extraction step. Figure (13) depicts the CNN architecture and explains each layer in detail:

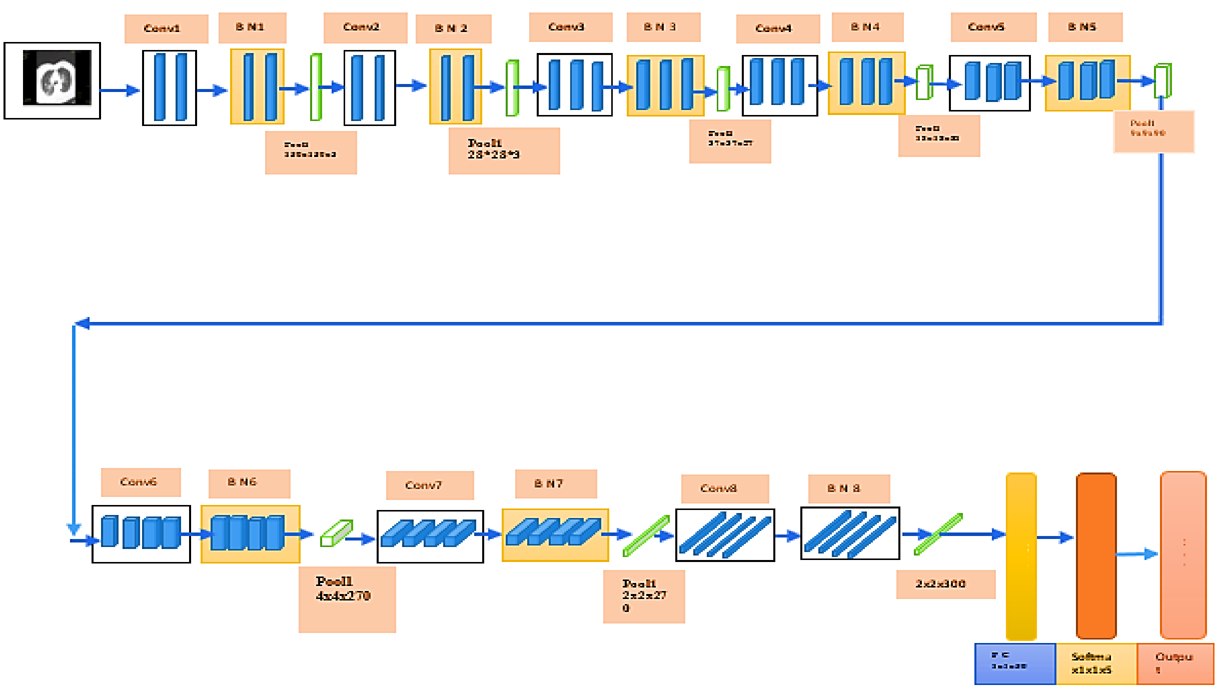


**Figure 13 Feature extraction using CNN**

### 1. Input Layer

The image size is 28 x 28 x 3, which is the same as the image's width, height, and channel size. There are three color channels since the digit data is made up of Covid-19 images. It is unnecessary to shuffle this layer's data since the train network does it for you at the beginning of training.

### 2. Convolutional Layer

To create a feature map, the feature extraction layer moves a weight slide mask over the original picture and performs dot product multiplication. Weight is produced at random, then after several iterations through the BN, Pooling, and RleU Layers, the weight varies until the optimal weight for this picture is found; this weight reflects the feature.

### 3. Batch Normalization Layer (BN)

This layer is used to speed up the training process and eliminate network initialization sensitivity by minimizing a large number of channels. First, the activation of each channel is normalized by subtracting the mean of the mini batch and dividing on the standard deviation of the mini batch, then the layer input is shifted by offset, and finally scaled by factor. Between the convolutional and RleU layers, this layer is utilized.

### 4. Max Pooling Layer

The max pooling layer has been used to get rid of excess and unwanted features, and it returns an important feature by sliding a mask with known dimension over the feature map that results from the previous convolutional layer, but the max is empty, so the result is the highest value lie under this mask at each stride.

### 5. RleU Layer

Since the images are naturally non-linear, and contain non-linear features such as color and border, the rectifier function is applied to increase non-linearity of the image, i.e. RleU layer is used to ensure only robust feature by taking only positive numbers and convert all negative numbers to zero.

### 6. Fully Connected Layer

It is a feature vector that contains the most significant information for the input; it collects features from all previous convolutional layers during training and may be utilized for classification afterwards. i.e. train a hidden layer to predict the likelihood of each class.

### 7. Softmax Layer

The Softmax layer output is a probability number ranging from (0-1); each class has its own probability, for example (0.011, 0.005); give the candidate class a high likelihood while decreasing the probability of other classes.

### 8. Loss Function Layer

The loss function is used to calculate the loss (error) at each trade epoch; it is also a key element on which the weight update during backpropagation is dependent, i.e. it displays the difference between anticipated output and true label.

### D. COVID -19 Classification

1. **Convolution Layers:** first option, size filtering, determines the height and breadth of filters that will apply the training function as they scan the images. The filter size in this work is 3 x 3 as shown by the number 3.
2. **Batch Normalize Layer:** Normalized activation and gradient propagation throughout the network, making training network improvements easier. Layers of batch normalization, such as ReLU layers, are employed between twisted layers and nonlinear lines to expedite network training and minimize network configuration. A batch settlement layer may be created using the payment settlement layer.
3. **ReLU Layers:** The BN layer is followed by a nonlinear activation function. The most often utilized activation function is the corrective linear unit (ReLU). The ReLU layer must be built first.
4. **MaxPooling Layer:** After convolutional layers (with activation functions), down sampling is often used to reduce the spatial feature map's size and remove unused spatial information. One way to reduce the number of samples is to utilize max pooling, which generates maxPooling2dLayer. The max pooling layer returns the rectangular input regions specified by the pool Size first parameter. The work's rectangular surface size is [2,2] square feet. Strides in the direction of the given name. The step parameter is defined by a value pair the size of which was determined by It makes use of the training function as it goes through the input.
5. **Fully Connected Layers:** All of the neurons in the preceding layer are interconnected in what's known as a fully connected layer. When looking for larger patterns, this layer combines all the information from previous levels. The picture categorization features are included into the final, fully connected layer. As a consequence, in the final fully connected layer, the target data classes number matches Output Size parameter. This project's output has a total size of ten, which is equal to the number of classes it contains. Fully connected layers may be used to build a totally linked one.

6. **SoftMax Layer:** The SoftMax activation function normalizes the output of the fully linked layer. It is possible that the classification layer will use the Soft Max layer's collection of one-to-one positive integers to calculate classification probabilities. An additional connected layer must be created after the first to make use of Soft Max Layers.
7. **Classification Layers:** Using the likelihood of the SoftMax activation function, this layer computes without sacrificing quality and provides mutually exclusive input classes for each one. The classification layer is created with the help of the classification layer.

## Experimental Results

At this stage, Covid 19 will be detected by going through several stages of the proposed method, and an explanation of this will be given below.

### 1. Convolution Layer

There are (8) filters in this layer, with a filter size of (3 x 3).

Scaling (padding) of the image is the same. The filtering procedure is one pixel, therefore stride = 1, which is the default setting, is appropriate. It's used in this layer kernel, which is also known as the filter. The aim of applying multiple filters in order to extract various characteristics is to identify the existence of specific attributes or patterns in the original image (input). To extract the features, the filter is tiny enough to scan the whole image and perform the necessary arithmetic operations between the filter values and the pixels. Simple and apparent characteristics, such as edges in opposite directions, are extracted using the first hidden layers. The intricacy of the characteristics that must be discovered and retrieved grows as we delve deeper into the network's hidden levels.

**Table 1 Iteration of Convolution layer**

| Iteration | Activation | Pyramid Level |
|---|---|---|
| 1 | 10.12 | 1 |
| 2 | 18.35 | 1 |
| 3 | 48.85 | 1 |
| 4 | 78.25 | 1 |
| 5 | 117.59 | 1 |
| 6 | 127.15 | 1 |
| 7 | 167.59 | 1 |
| 8 | 196.04 | 1 |
| 9 | 235.49 | 1 |
| 10 | 254.93 | 1 |
| 1 | 594.59 | 2 |
| 2 | 612.93 | 2 |
| 3 | 645.28 | 2 |
| 4 | 642.62 | 2 |
| 5 | 751.97 | 2 |
| 6 | 738.31 | 2 |
| 7 | 768.65 | 2 |
| 8 | 799.02 | 2 |
| 9 | 819.35 | 2 |
| 10 | 848.70 | 2 |

## 2. Batch Normalization

For each mini-batch, standardizes the inputs to a layer. This stabilizes the learning process and significantly reduces the number of training epochs needed to build deep networks.

**Table 2 Iteration of batch normalization**

| Iteration | Activation | Pyramid Level |
|---|---|---|
| 1 | 0.0 | 1 |
| 2 | 0.57 | 1 |
| 3 | 0.88 | 1 |
| 4 | 0.91 | 1 |
| 5 | 0.96 | 1 |
| 6 | 1.05 | 1 |
| 7 | 1. 07 | 1 |
| 8 | 1.09 | 1 |
| 9 | 1.10 | 1 |
| 10 | 1.11 | 1 |
| 1 | 1.12 | 2 |
| 2 | 1.14 | 2 |
| 3 | 1.16 | 2 |
| 4 | 1.18 | 2 |
| 5 | 1.20 | 2 |
| 6 | 1.21 | 2 |
| 7 | 1.22 | 2 |
| 8 | 1.24 | 2 |
| 9 | 1.26 | 2 |
| 10 | 1.28 | 2 |

## 3. ReLU Layer

With a rectified linear activation function (ReLU), you get an output equal to the input value directly; otherwise, you get zero. ReLU is a piecewise linear function. It has become the default activation function for many types of neural networks since it is easier to train and often provides better results.

**Table 3 Iteration of ReLU**

| Iteration | Activation | Pyramid Level |
|---|---|---|
| 1 | 0.46 | 1 |
| 2 | 0.66 | 1 |
| 3 | 0.95 | 1 |
| 4 | 1.02 | 1 |
| 5 | 1.04 | 1 |
| 6 | 1.06 | 1 |
| 7 | 1.08 | 1 |
| 8 | 1.11 | 1 |
| 9 | 1.11 | 1 |
| 10 | 1.11 | 1 |
| 1 | 1.11 | 2 |
| 2 | 1.13 | 2 |
| 3 | 1.13 | 2 |
| 4 | 1.14 | 2 |
| 5 | 1.16 | 2 |
| 6 | 1.16 | 2 |
| 7 | 1.15 | 2 |
| 8 | 1.15 | 2 |
| 9 | 1.17 | 2 |
| 10 | 1.18 | 2 |

## 4. Max Pooling

Because more than one filter may be employed, the pooling layer's aim is to keep the activation maps as small as possible. This not only cuts down on calculation time, but it also prevents models from being overfit. The following two functions may be used to decrease the size of big matrices: Calculate the highest possible value in each window by using the formula max. Do an arithmetic average of the data in the single window. Average: The first method, however, is the most popular: Max-pooling. An important aim is to retain bigger values inside each window while erasing the activation map (or feature matrix) to reduce the map's size.

**Table 4 Iteration of Max-pooling**

| Iteration | Activation | Pyramid Level |
|---|---|---|
| 1 | 0.86 | 1 |
| 2 | 1.30 | 1 |
| 3 | 1.51 | 1 |
| 4 | 1.58 | 1 |
| 5 | 1.63 | 1 |
| 6 | 1.67 | 1 |
| 7 | 1.81 | 1 |
| 8 | 1.83 | 1 |
| 9 | 1.86 | 1 |
| 10 | 1.88 | 1 |
| 1 | 1.92 | 2 |
| 2 | 1.94 | 2 |
| 3 | 2.04 | 2 |
| 4 | 1.77 | 2 |
| 5 | 1.90 | 2 |
| 6 | 2.34 | 2 |
| 7 | 2.74 | 2 |
| 8 | 2.88 | 2 |
| 9 | 2.99 | 2 |
| 10 | 3.12 | 2 |

## 5. Convolution-2 layer

This layer has sixteen filters with a filter size of one (3 x 3). The process of filtering the filter is one pixel, thus the value of stride = 1, which is the default value, is the same.

**Table 5 Iteration of Convolution-2**

| Iteration | Activation | Pyramid Level |
|---|---|---|
| 1 | 0.04 | 1 |
| 2 | 0.08 | 1 |
| 3 | 0.14 | 1 |
| 4 | 0.22 | 1 |
| 5 | 0.29 | 1 |
| 6 | 0.35 | 1 |
| 7 | 0.40 | 1 |
| 8 | 0.44 | 1 |
| 9 | 0.48 | 1 |
| 10 | 0.51 | 1 |
| 1 | 0.55 | 2 |
| 2 | 0.58 | 2 |
| 3 | 0.66 | 2 |
| 4 | 0.72 | 2 |
| 5 | 0.80 | 2 |
| 6 | 1.10 | 2 |
| 7 | 1.12 | 2 |
| 8 | 1.14 | 2 |
| 9 | 1.16 | 2 |
| 10 | 1.19 | 2 |

## 6. batch normalization _ 2

As illustrated above, the layer is reused.

**Table 6 Iteration of batch normalization _ 2**

| Iteration | Activation | Pyramid Level |
|---|---|---|
| 1 | 0.00 | 1 |
| 2 | 0.26 | 1 |
| 3 | 0.46 | 1 |
| 4 | 0.91 | 1 |
| 5 | 0.97 | 1 |
| 6 | 1.03 | 1 |
| 7 | 1.07 | 1 |
| 8 | 1.12 | 1 |
| 9 | 1.26 | 1 |
| 10 | 1.28 | 1 |
| 1 | 1.30 | 2 |
| 2 | 1.35 | 2 |
| 3 | 1.39 | 2 |
| 4 | 1.43 | 2 |
| 5 | 1.56 | 2 |
| 6 | 1.58 | 2 |
| 7 | 1.60 | 2 |
| 8 | 1.67 | 2 |
| 9 | 1.71 | 2 |
| 10 | 1.72 | 2 |

## 7. RELU _ 2

As illustrated above, the layer is reused.

**Table 7 Iteration of RELU _ 2**

| Iteration | Activation | Pyramid Level |
|---|---|---|
| 1 | 0.46 | 1 |
| 2 | 0.66 | 1 |
| 3 | 0.95 | 1 |
| 4 | 1.02 | 1 |
| 5 | 1.04 | 1 |
| 6 | 1.06 | 1 |
| 7 | 1.08 | 1 |
| 8 | 1.11 | 1 |
| 9 | 1.11 | 1 |
| 10 | 1.11 | 1 |
| 1 | 1.11 | 2 |
| 2 | 1.13 | 2 |
| 3 | 1.13 | 2 |
| 4 | 1.14 | 2 |
| 5 | 1.16 | 2 |
| 6 | 1.16 | 2 |
| 7 | 1.15 | 2 |
| 8 | 1.15 | 2 |
| 9 | 1.17 | 2 |
| 10 | 1.18 | 2 |

## 8. Max Pooling_2

As illustrated above, the layer is reused.

**Table 8 Iteration of Max Pooling_2**

| Iteration | Activation | Pyramid Level |
|---|---|---|
| 1 | 0.35 | 1 |
| 2 | 1.20 | 1 |
| 3 | 1.51 | 1 |
| 4 | 1.59 | 1 |
| 5 | 1.63 | 1 |
| 6 | 1.67 | 1 |
| 7 | 1.81 | 1 |
| 8 | 1.83 | 1 |
| 9 | 1.86 | 1 |
| 10 | 1.88 | 1 |
| 1 | 1.92 | 2 |
| 2 | 1.94 | 2 |
| 3 | 2.04 | 2 |
| 4 | 1.77 | 2 |
| 5 | 1.99 | 2 |
| 6 | 2.34 | 2 |
| 7 | 2.81 | 2 |
| 8 | 2.88 | 2 |
| 9 | 2.98 | 2 |
| 10 | 3.45 | 2 |

## 9. Fully Connected

The network layer in which all of the neurons in the layer are linked to all of the neurons in the preceding layer.

**Table 9 Iteration of fully connected**

| Iteration | Activation | Pyramid Level |
|---|---|---|
| 1 | 0.35 | 1 |
| 2 | 2.25 | 1 |
| 3 | 3.69 | 1 |
| 4 | 5.08 | 1 |
| 5 | 6.18 | 1 |
| 6 | 7.32 | 1 |
| 7 | 8.92 | 1 |
| 8 | 9.29 | 1 |
| 9 | 9.80 | 1 |
| 10 | 10.13 | 1 |
| 1 | 10.85 | 2 |
| 2 | 11.45 | 2 |
| 3 | 11.69 | 2 |
| 4 | 12.34 | 2 |
| 5 | 13.19 | 2 |
| 6 | 1.57 | 2 |
| 7 | 2.39 | 2 |
| 8 | 3.90 | 2 |
| 9 | 4.98 | 2 |
| 10 | 2.82 | 2 |

## 10. Softmax Layer

To sum up a set of K real numbers, use the softmax function, which takes a set of K real numbers and adds 1. Positive, negative, zero, or more than one input values are converted to values between 0 and 1, enabling them to be interpreted as probabilities by the softmax. There are two ways the softmax may handle tiny or negative input: a little probability, or a high probability. The probability will always be between zero and one no matter whether input is small or negative.

**Table 10 Iteration of Softmax**

| Iteration | Activation | Pyramid Level |
|---|---|---|
| 1 | 0.20 | 1 |
| 2 | 0.60 | 1 |
| 3 | 0.87 | 1 |
| 4 | 0.97 | 1 |
| 5 | 0.99 | 1 |
| 6 | 1.00 | 1 |
| 7 | 1.00 | 1 |
| 8 | 1.00 | 1 |
| 9 | 1.00 | 1 |
| 10 | 1.00 | 1 |
| 1 | 1.00 | 2 |
| 2 | 1.00 | 2 |
| 3 | 1.00 | 2 |
| 4 | 1.00 | 2 |
| 5 | 1.00 | 2 |
| 6 | 1.00 | 2 |
| 7 | 1.00 | 2 |
| 8 | 1.00 | 2 |
| 9 | 1.00 | 2 |
| 10 | 1.00 | 2 |

## 11. Classification Layer

The number of classes is calculated using the output size of the preceding layer. To determine the number of classes K in a network, include a fully connected layer with output size K and a softmax layer before the classification layer. As shown in figure (14).


**Figure 14 Classification of CNN**

## Evaluation of CNN

Utilize of standard measures to determine the performance value of an item. The accuracy of detection of covid 19 in CNN is 100% as shown in Table (11). Where The factors that influence the network topology and how the network is trained are referred to as hyper parameters. The learning rate determines how rapidly a network's parameters are updated. The learning process is slowed by a low learning rate, yet it converges gradually. A faster learning rate accelerates learning but may not converge. A declining Learning rate is usually recommended. The number of epochs is the number of times the network is presented the entire training data while training. The accuracy of the mini-batch reported during training correlates to the accuracy of the specific mini-batch at the given iteration. It is not a running average calculated across iterations. During stochastic gradient descent with momentum (SGDM) training, the method divides the whole data set into separate mini-batches. An iteration is the computation of the gradients of the network for each mini-batch. Moving through every possible mini-batch corresponds to an era. Calculates the error for each example in the training dataset, but only after all training examples have been assessed does the model get updated. When we train the model, the accuracy and loss in the model for validation data may vary depending on the scenario. Typically, as epochs increase, loss should decrease and accuracy should increase. Val loss begins to fall, but val acc begins to rise. This is also great since it implies the model is learning and operating properly.

**Table 11 Accuracy of CNN**

| # | Iteration | Time Elapsed (mm:ss) | Mini-batch Accuracy | Validation Accuracy | Mini-batch Loss | Validation Loss | Base Learning Rate |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 00:02 | 11 % | 45% | 3.0615 | 1.6100 | 100 |
| 2 | 10 | 00:04 | 96% | 96% | 0.0721 | 0.0609 | 100 |
| 3 | 20 | 00:06 | 100% | 99% | 0.0077 | 0.0183 | 100 |
| 4 | 30 | 00:08 | 100% | 100% | 0.0096 | 0.0084 | 100 |
| 5 | 40 | 00:10 | 100% | 100% | 0.0041 | 0.0065 | 100 |
| 7 | 50 | 00:11 | 100 % | 100 % | 0.0040 | 0.0049 | 100 |
| 8 | 60 | 00:13 | 100 % | 100% | 0.0033 | 0.0036 | 100 |
| 9 | 70 | 00:15 | 100% | 100.00% | 0.0019 | 0.0028 | 100 |

## Conclusion

Due to the need to know whether a person has Covid 19 or not, it was necessary to create an effective model for that, but this model is not only based on taking the global data set and training it only, but we wanted to train it and apply it to a living system from the reality of the Iraqi environment. The affected person has already done a CT scan, but because we forget that this data may need to pre-process the image because it is possible that there will be noise during transmission, but we get rid of this doubt. The process of image optimization was performed before entering deep learning using bilateral filter and then the image was entered into the system three times of CNN layers, where it was done once in order to obtain the distinctive properties, and eight levels were used to give the best results, and the classification process was performed using 12 levels to give the best results. calendar. Where the system was trained by 80 percent and tested by 20 percent, and the results showed that the accuracy of the system was up to 100%, and this percentage is not error-prone.

## Acknowledgement

The authors' thankful Department of Computer Science, College of Science, Mustansiriyah University, for supporting this work.

## References

Ardabili, S.F., Mosavi, A., Ghamisi, P., Ferdinand, F., Varkonyi-Koczy, A.R., Reuter, U., & Atkinson, P.M. (2020). Covid-19 outbreak prediction with machine learning. *Algorithms, 13*(10).

Balodi, T. (20121). Convolutional Neural Network with Python Code Explanation | Convolutional Layer | Max Pooling in CNN.

Bhattacharya, S., Maddikunta, P.K.R., Pham, Q.V., Gadekallu, T.R., Chowdhary, C.L., Alazab, M., & Piran, M.J. (2021). Deep learning and medical image processing for coronavirus (COVID-19) pandemic: A survey. *Sustainable cities and society, 65.*

Carrillo-Larco, R.M., & Castillo-Cara, M. (2020). Using country-level variables to classify countries according to the number of confirmed COVID-19 cases: An unsupervised machine learning approach. *Wellcome Open Research, 5.*

Fauci, A.S., Lane, H.C., & Redfield, R.R. (2020). *Covid-19 navigating the uncharted.*

Fibriani, I., Widjonarko, W., Prasetyo, A., Raharjo, A.M., & Irawan, D.E. (2020). *Multi Deep Learning to Diagnose COVID-19 in Lung X-Ray Images with Majority Vote Technique.*

Ghaderzadeh, M., & Asadi, F. (2021). Deep learning in the detection and diagnosis of COVID-19 using radiology modalities: a systematic review. *Journal of Healthcare Engineering.*

Haque, A.K.M., Pranto, T.H., Noman, A.A., & Mahmood, A. (2021). Insight about Detection, Prediction and Weather Impact of Coronavirus (COVID-19) using Neural Network. *arXiv preprint arXiv:2104.02173.*

Harizi, I., Berkane, S., & Tayebi, A. (2021). Modeling the Effect of Population-Wide Vaccination on the Evolution of COVID-19 Epidemic in Canada. medRxiv.

Https://data-flair.training/blogs/deep-learning-terminologies/,"le."

Https://developer.apple.com/library/archive/documentation/Performance/Conceptual/vImage/ConvolutionOperations/ConvolutionOperations.html

Https://towardsdatascience.com/covolutional-neural-network-cb0883dd6529.

Https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53.

Huang, K., Hussain, A., Wang, Q.F., & Zhang, R. (Eds.). (2019). *Deep learning: fundamentals, theory and applications.* Springer, 2.

Iwendi, C., Mahboob, K., Khalid, Z., Javed, A.R., Rizwan, M., & Ghosh, U. (2021). Classification of COVID-19 individuals using adaptive neuro-fuzzy inference system. *Multimedia Systems,* 1-15.

Javed, A.R., Sarwar, M.U., Beg, M.O., Asim, M., Baker, T., & Tawfik, H. (2020). A collaborative healthcare framework for shared healthcare plan with ambient intelligence. *Human-centric Computing and Information Sciences*, *10*(1), 1-21.

Kara, M., Öztürk, Z., Akpek, S., & Turupcu, A. (2021). *COVID-19 diagnosis from chest CT Scans: a weakly supervised CNN-LSTM approach.*

Kattenborn, T., Leitloff, J., Schiefer, F., & Hinz, S. (2021). Review on Convolutional Neural Networks (CNN) in vegetation remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing, 173,* 24-49.

Kassania, S.H., Kassanib, P.H., Wesolowskic, M.J., Schneidera, K.A., & Detersa, R. (2021). Automatic detection of coronavirus disease (COVID-19) in X-ray and CT images: a machine learning based approach. *Biocybernetics and Biomedical Engineering, 41*(3), 867-879.

Khanday, A.M.U.D., Rabani, S.T., Khan, Q.R., Rouf, N., & Din, M.M.U. (2020). Machine learning based approaches for detecting COVID-19 using clinical text data. *International Journal of Information Technology, 12*(3), 731-739.

Khmaissia, F., Haghighi, P.S., Jayaprakash, A., Wu, Z., Papadopoulos, S., Lai, Y., & Nguyen, F.T. (2020). An unsupervised machine learning approach to assess the zip code level impact of covid-19 in nyc. *arXiv preprint arXiv:2006.08361*.

Kundu, R., Basak, H., Singh, P. K., Ahmadian, A., Ferrara, M., & Sarkar, R. (2021). Fuzzy rank-based fusion of CNN models using Gompertz function for screening COVID-19 CT-scans. Scientific reports, 11(1), 1-12.

Liao, J., Fan, S., Chen, J., Wu, J., Xu, S., Guo, Y., & Lang, C. (2020). Epidemiological and clinical characteristics of COVID-19 in adolescents and young adults. *The Innovation, 1*(1), 100001

Liu, R. W., Yuan, W., Chen, X., & Lu, Y. (2021). An enhanced CNN-enabled learning method for promoting ship detection in maritime surveillance system. *Ocean Engineering, 235.*

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision, 60*(2), 91-110.

Manavalan, R. (2020). Automatic identification of diseases in grains crops through computational approaches: A review. *Computers and Electronics in Agriculture, 178.*

Manoj, M., Srivastava, G., Somayaji, S.R.K., Gadekallu, T.R., Maddikunta, P.K.R., & Bhattacharya, S. (2020). An Incentive Based Approach for COVID-19 planning using Blockchain Technology. *In IEEE Globecom Workshops (GC Wkshps,* 1-6.

Mehanna, H., Hardman, J.C., Shenson, J.A., Abou-Foul, A.K., Topf, M.C., & AlFalasi, M. (2020). COVID-19 content collection. *The Lancet Oncology, 21*(7).

Mukherjee, H., Ghosh, S., Dhar, A., Obaidullah, S.M., Santosh, K.C., & Roy, K. (2021). Deep neural network to detect COVID-19: one architecture for both CT Scans and Chest X-rays. *Applied Intelligence, 51*(5), 2777-2789.

Panwar, H., Gupta, P.K., Siddiqui, M.K., Morales-Menendez, R., & Singh, V. (2020). Application of deep learning for fast detection of COVID-19 in X-Rays using nCOVnet. *Chaos, Solitons & Fractals, 138.*

Reddy, G.T., Reddy, M.P.K., Lakshmanna, K., Rajput, D.S., Kaluri, R., & Srivastava, G. (2020). Hybrid genetic algorithm and a fuzzy logic classifier for heart disease diagnosis. *Evolutionary Intelligence, 13*(2), 185-196.

Rehman, A., Iqbal, M.A., Xing, H., & Ahmed, I. (2021). COVID-19 Detection Empowered with Machine Learning and Deep Learning Techniques: A Systematic Review. *Applied Sciences, 11*(8).

Ulhaq, A., & Burmeister, O. (2020). Covid-19 imaging data privacy by federated learning design: A theoretical framework. *arXiv preprint arXiv:2010.06177.*

Velavan, T.P., & Meyer, C.G. (2020). The COVID-19 epidemic. *Tropical medicine & international health, 25*(3), 278-280.

Wang, T., Huan, J., & Li, B. (2018). Data dropout: Optimizing training data for convolutional neural networks. *In IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI),* 39-46.

Williams, A.A. (2019). Ordinal Outcome Modeling: The Application of the Adaptive Moment Estimation Optimizer to the Elastic Net Penalized Stereotype Logit. *Journal of Data Analysis and Information Processing, 7*(01), 14.

Zhang, Q., Zhang, M., Chen, T., Sun, Z., Ma, Y., & Yu, B. (2019). Recent advances in convolutional neural network acceleration. *Neurocomputing, 323,* 37-51.