DOI: 10.24996/ijs.2025.66.4.24

ISSN: 0067-2904

Prediction of COVID-19 Disease and Infection Rate Based on Dense Net

Saja Ali, Alyaa Al-Barrak
Department of Computer Science, Collage of Science, University of Baghdad, Baghdad, Iraq

Received: 15/1/2024 Accepted: 16/5/2024 Published: 30/4/2025

Abstract

Coronavirus is an RNA (Ribonucleic acid) virus in the coronaviridian family that causes zoonotic and infectious diseases transmitted between animals and evolved between humans. This class of pathogens is responsible for respiratory diseases. Coronavirus refers to the crown-like protrusions on the outside surface of the virus. Corona is an infection that causes breathing difficulties in humans. In epidemics, symptomatic techniques based on graphic design are essential for examining the causes of influence, which leads to better results than primary radioscopy mechanisms for identifying and diagnosing COVID-19 cases. The urgent need to employ artificial intelligence in disease detection arose from this standpoint. In this paper, a system is proposed to diagnose infected persons by building a deep learning model and preprocessing processes homogeneously to investigate CT (coronavirus-computed tomography) scan radiographs using the global SARS-Covid dataset, achieving a 99% accuracy rate in diagnosing and identifying COVID-19 or non-COVID-19.

Keywords: Coronavirus, Disease, CT, Deep Learning, Detection.

التنبؤ بمرض كوفيد-19 ومعدل الإصابة على أساس الشبكة الكثيفة

سجى علي ، علياء البراك
قسم الحاسوب، كلية العلوم، جامعة بغداد، بغداد، العراق

الخلاصة

فيروس كورونا هو فيروس حمض نووي ريبوزي ينتمي إلى فصيلة كورونا فيريديان ويسبب أمراضًا حيوانية ومعدية تنتقل بين الحيوانات وتطورت بين البشر . هذه الفئة من مسببات الأمراض هي المسؤولة عن أمراض الجهاز التنفسي. ويشير فيروس كورونا إلى النتوءات التي تشبه التاج على السطح الخارجي للفيروس. كورونا عدوى تسبب صعوبات في التنفس لدى الإنسان. تعد تقنيات الأعراض المعتمدة على التصميم الجرافيكي ضرورية لفحص أسباب التأثير، مما يؤدي إلى نتائج أفضل من آليات التنظير الإشعاعي الأولية لتحديد وتشخيص حالات كوفيد-19. ومن هذا المنطلق، برزت الحاجة الملحة لتوظيف الذكاء الاصطناعي في الكشف عن الأمراض. في هذا البحث، تم اقتراح نظام لتشخيص المصابين من خلال بناء نموذج التعلم العميق وعمليات المعالجة المسبقة بطريقة متجانسة لفحص الصور الشعاعية للأشعة المقطعية باستخدام مجموعة بيانات SARS-Covid العالمية، مما يحقق دقة تصل إلى 99٪ في الكشف عن فيروس كورونا مصاب او غير مصاب.

*Email: alyaa.al-barrak@sc.uobaghdad.edu.iq

1. Introduction

The global community is currently facing a novel health crisis that threatens public health due to the rapid transmission of the COVID-19 virus. (Coronavirus Disease-2019). COVID-19 has recently spread quickly to several nations across all continents, including South Brazil, Africa, Russia, India, and the United States [1]. The emergence of COVID-19 in December 2019 at the Hunan seafood market in Wuhan, South China, and its subsequent global spread has led to the World Health Organization declaring the virus outbreak a public health emergency of international concern (WHO) [2]. Both the number of infected individuals and the mortality rate are rising. Fever, dry cough, myalgia, dyspnoea, and headache are the most typical symptoms of the novel Coronavirus. Nonetheless, in some cases, the absence of symptoms (asymptomatic) makes the disease a more significant hazard to public health [3]. COVID-19 has been demonstrated to be among the most dangerous ailments severely threatening human civilization [4]. More than 516 million cases have been confirmed globally as of May 2022 [5], with the advancement of modern technology over the past few decades, ingenious tools and facilities have been developed to aid in disease diagnosis, prevention, and control. Specifically, imaging modalities such as CT and X-ray are among the most effective for COVID-19 diagnosis [6]. COVID-CT and SARS-CoV-2 datasets test the model. This scientific paper aims to propose a medical system that can detect infected people from non-infected people in covid disease, despite the media not shedding light on the pandemic at present, as in the past, now that the disease is still developing rapidly and also causes deaths in the elderly. Therefore, the aim of the research was early detection based on artificial intelligence.

2. Related Work

Hamza Abu Owida et.al.2022 [7]. Suggested a deep learning approach based on ResNet architecture enhancement as a transfer learning model, as well as a system to quickly identify COVID-19. The architecture of the suggested model minimizes ResNet-50 block linkages. This reduces the scale-ability training time and resolves the vanishing gradient problem. The proposed approach is evaluated on two well-known COVID-19 CT datasets with patient-based divides. Resnet18, Google Net, and Alex Net were used to provide a COVID-19 diagnosis. The loss indicates that for SARS-CoV-2 datasets, some architecture combinations can improve detection performance. Accuracy was increased by 3% using three parallel pre-trained networks. The evaluation factors, including accuracy, improved. According to Muhammet Fatih Aslan et.al.2022 [8], CT imaging makes it possible to diagnose COVID-19 quickly and accurately. These two models both use AlexNet. The mAlexNet architecture will first be built and classified. In the second model, which employs 25 mAlexNet features, picture categorization is carried out by an ANN (artificial neural network) that has been tuned using the TSA (tree seed method). Using 3D CT slices, Fozia Mehboob et.al.2022 [9] presented a self-attention transformer-based diagnosis method for COVID-19 diagnosis. The transformer vision approach that has been suggested can forecast the COVID-19 quantification by utilizing the pixel values found in the long-range relation-based maps. The sine-cosine algorithm (SCA) is recommended by Binfeng Xu et.al.2022 [10] to optimize the structure of deep convolutional neural networks (DCNNs) to determine the state of raw CT images. Accuracy and speed are increased through three regular SCA-based upgrades. Initially, IP-based encoding is recommended. A variable-length DCNN is generated by an enfeebled layer. The model is tested using the SARS-CoV-2 and COVID-CT datasets. The following Table 1 shows the summary of related work.

Table 1: Comparison between related work in terms of method, datasets, and result.

Rf.	METHOD	Dataset	Result
(H. A. B. U. Owida, 2022)	ResNet	• COVID-CT (2482 CT images) training (80%) and test (20%) • CoV-2 CT scan (total 812 CT images)	COVID-CT Accuracy= 98.1 Precision= 96.7 Specificity= 97 Sensitivity= 98.3 F1-Score= 97.48 CoV-2 CT scan Accuracy= 90.6 Precision= 88 Specificity= 86.6 Sensitivity= 94 F1-Score= 91.4
(H. A. Owida et al., 2022)	• ResNet-50 • Baseline • Augmentation (rotation, flipping, random zooming, and scaling)	• SARS-CoV-2 • (training (80%) and test (20%) partitions.) • COVID-CT • (validation and test sets)	COVID-CT Accuracy= 98.1 Precision= 96.7 Specificity= 97 Sensitivity= 98.3 F1-Score= 97.48 Accuracy=90.6 Precision= 88 Specificity= 86.6 Sensitivity= 94 F1-Score= 91.4
(Aslan et al., 2022)	• Tree Seed Algorithm (TSA) • Alex Net(features)	• SARS-CoV-2	Model (mAlexNet) Accuracy= 97.92 Precision=0.973 Specificity= 0.9768 Sensitivity= 0.9820 F1-Score=0.977 MCC=0.958 Model (Alex Net + TSA-ANN) Accuracy= 98.54 Precision= 0.9909 Specificity= 0.9923 Sensitivity= 0.9775 F1-Score= 0.9841 MCC= 0.9708
(Mehboob et al., 2022)	Vision transformer (Two models) for two dataset	• Hust19 Dataset • SARS-CoV-2	Hust19 Dataset Accuracy= 98 SARS-CoV-2 Accuracy= 97
(B. Xu et al., 2022)	• DCNN • Sine-cosine algorithm	• COVID-CT • CoV-2 CT scan	CoV-2 CT scan Accuracy= 98.32 Precision= 98.23 Specificity= 96.77 Sensitivity= 97.22 F1-Score= 97.99 COVID-CT Accuracy= 98.01 Precision=98.95 Specificity=96.44 Sensitivity=96.23 F1-Score=97.45

3. Role of Intelligence in the Detection of Covid-19

Artificial intelligence is the simulation of human intelligence processes by machines, primarily computer systems [11]. Specific AI (Artificial intelligence) applications include expert systems, natural language processing, speech recognition, and machine vision [12]. AI systems function by ingesting vast quantities of labeled training data, analyzing the data for correlations and patterns, and employing these patterns to predict future states. A chatbot that receives text exchange examples can learn to simulate natural human conversations. An image recognition tool can learn to identify and describe objects in images by analyzing millions of examples [13]. The ideal characteristic of artificial intelligence is the capacity to reason and take actions with the highest probability of attaining a specific objective. The application of DL (Deeb learning) technology has greatly benefited the field of medical image classification and diagnosis based on X-ray, CT, MRI (Magnetic resonance imaging), and other imaging methods [14].

4. Learning Techniques

Artificial intelligence, which includes learning methods, has revolutionized computer technology. It's mostly ML (Machine learning) and DL. Often, AI works with machine learning [15]. ML is an algorithm-based intelligent artificial system that is self-learning. Machine Learning (ML) refers to systems that demonstrate increasing levels of innovation and intelligence without direct human intervention [16]. The second innovation is in the discipline of deep learning. The application of machine learning techniques to the examination of massive data collections. The advancement of machine learning is guided by the principles of exactness and velocity [17]. Deep Learning (DL) differentiates itself from conventional machine learning (ML) techniques by utilizing multiple layers that enhance abstraction and robust generalization. Using neural networks requires extracting attributes to reduce the network computational burden. For neural networks to integrate features. Essential will be establishing a comprehensive network capable of communicating precise directives [18]. As a consequence, training likelihood of success was minimal or non-existent. Due to advances in computing power and memory capacity, it is now feasible to train large networks effectively. In the same deep-learning network, feature extraction classification and generalization can be executed, yielding significant advantages [19].

5. Pre-Trained Model in Deep Learning

Deep Learning (DL) algorithms [20] are trained on training models. It includes sample output data and input data sets affecting output. The training model runs input data through the algorithm to match processed output with sample output. A model's precision depends on the accuracy of the training or validation dataset. In data science, model training involves applying the optimal weights and biases to a machine learning algorithm to minimize a loss function across the prediction range [21]. The most common pre-trained models:

ResNet: is a computer vision deep learning model. Its CNN (convolutional neural networks) framework supports hundreds or thousands of convolutional layers. Previous CNN designs could not scale to many layers, limiting speed. When adding layers, researchers encountered the vanishing gradient issue. [22].

U-Net: Semantic U-Net segmentation. It contracts and expands. Path reduction resembles convolutional networks. Two 3x3 unpadded convolutions, a ReLU, and a 2x2 maximum pooling algorithm with stride 2 for downsampling are replicated. Downsampling stages multiply feature channels. Every step on the expansive path includes an up-sampling of the feature map, an up-convolution that halves the number of feature channels, a concatenation with the compressed feature map from the contracting path, and two 3x3 convolutions followed by a ReLU. Convolutions require clipping boundary pixels. A 1x1 convolution maps 64-component feature vectors to the desired number of classes at the final layer [23].

AlexNet: Eight-layer convolutional neural network AlexNet. The pre-trained network classifies images into 1000 object groups, including keyboard, mouse, pencil, and many animals. [24].

VGGNet: Karen Simonyan and Andrew Zisserman from Oxford proposed VGGNet in 2014. This study examines how convolutional neural network depth affects accuracy [25].

6. Proposed System

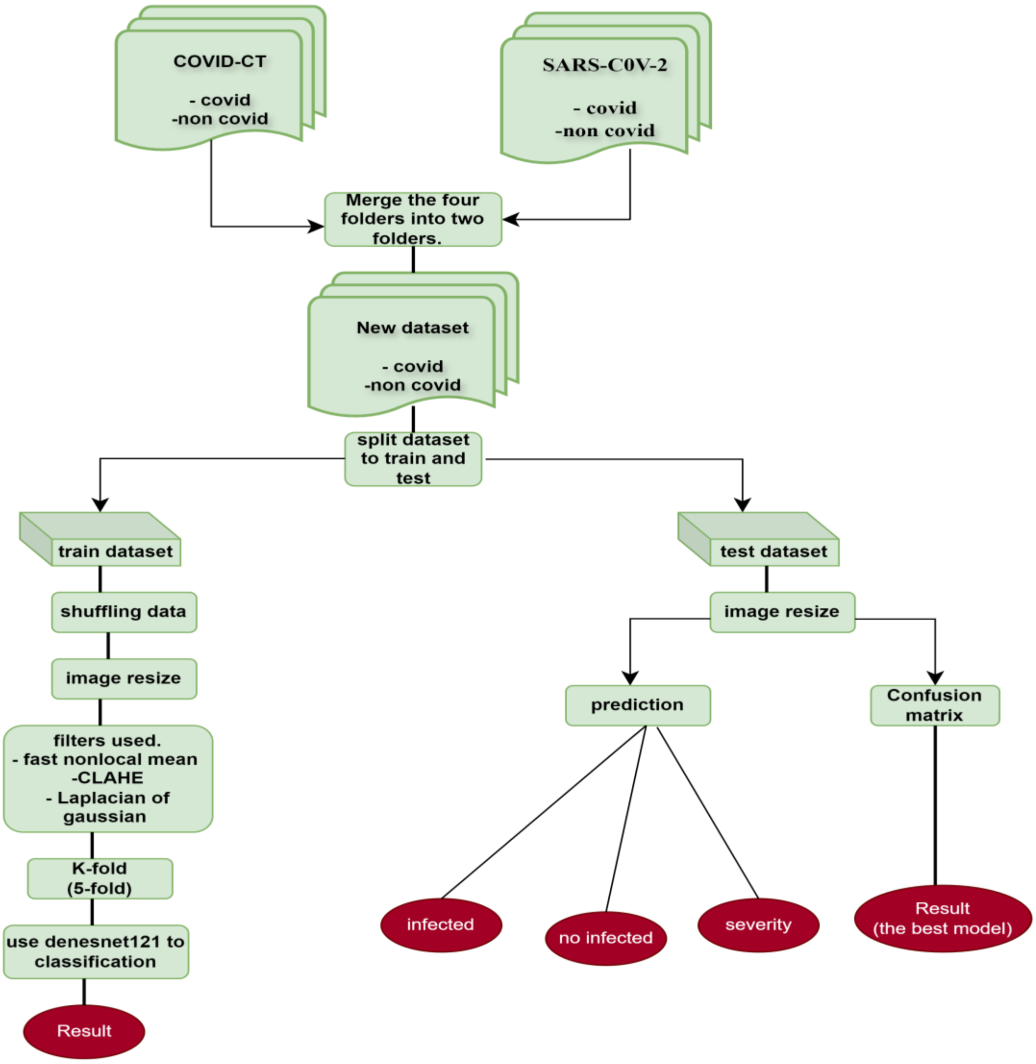

Figure 1: Proposed system design**.**

In the proposed system, two global data sets were used, and these data were merged into a new data set, after which they were divided into training and testing. In the training phase, several steps were taken, including mixing the data to achieve fairness and changing the size of the images to 224 * 224, because the size of the images was different, three filters were applied, to achieve great results. The first filter helped remove noise from the image, and the second unified the colors of the images and improved contrast. The third filter clarified the boundaries of the image. After that, the k-fold was used to divide the training data into validation and training. DenseNet was used to classify the images. As for the data In the test, the size of the images was also standardized and the confusion matrix was applied to them to evaluate the system, and then a prediction was made on the images to know who was infected and who was not infected and the infection rate. Figure 1 shows the block diagram of the system.

Algorithm of the proposed system

Begin
Step one: Create output dataset folders
Create a new folder named (New Dataset);
Inside New dataset folder;
    create a subfolder named (Covid)
Inside New dataset folder;
    create another subfolder named (Non_covid)
Step two:
Gather Covid Dataset from Dataset One and Dataset Two and put them in a subfolder named (Covid) (created in step one) as described below:
copy ('./Covid/COVID/','./preprocced_images/New dataset/covid/')
copy ('./CT_covid/CT_COVID','./preprocced_images/New dataset/covid/')
Gether non_covid Dataset from Dataset one and Dataset two and put them in a subfolder named (Non_covid) (created in step one) as described below:
copy ('./Covid/non-COVID','./preprocced_images/New dataset/non_covid/')
copy ('./CT_covid/CT_NonCOVID','preprocced_images/New dataset/non_covid/')
Step three: Read dataset image names from the local drive into the train_data array
train_data=[]
disease_type_dataset=[coved , Non_coved]
  for index, sp in enumerate(disease_type_dataset):
    for file in os.listdir(os.path.join(train_dir, sp)):
      train_data.append([sp + "/" + file, index, sp])
Step four: Covert train_data array into data frame form
  dataset_preproccesing = pd.DataFrame(train_data, columns = ['File', 'ID',' Disease Type'])
  dataset_preproccesing = dataset_preproccesing.sample(frac = 1, replace=False, random_state = 40)
Step five: Reset indices (row numbers)
  dataset_preproccesing = dataset_preproccesing.reset_index(drop = True)
Step six: Read images using the open computer vision library (cv2)
  for i, file path in enumerate(dataset_preproccesing['File']. values):
    img = cv2.imread(os.path.join(data_dir, file path))
Step seven: Get the 3 (b,r,g) channels from the original image and rearrange them into (r,g,b)
    b,g,r = cv2.split(img)
    rgb_img = cv2.merge([r,g,b])
Step eight: Resize image into 224x224
      imgclahe=cv2.resize(imgclahe, (224,224), interpolation = cv2.INTER_AREA)
Step nine: Denoising using fastNlMeansDenoisingColored
    DST = cv2.fastNlMeansDenoisingColored(img)
Step ten: Applying CLAHE filter
clahe_model = cv2.createCLAHE(clipLimit=15.0, tileGridSize=(8,8))
    colorimage_b = clahe_model.apply(R_channel)
    colorimage_g = clahe_model.apply(G_channel)
    colorimage_r = clahe_model.apply(B_channel)
    imgclahe = cv2.merge((colorimage_b, colorimage_g, colorimage_r))
Step eleven: Applying sharpened Filter
      sharpen_filter = np. array ([0, -1, 0]

```
                                   [-1,   5,-1 ]
                                   [ 0,  -1, 0 ] )
     //applying kernels to the input image to get the sharpened image
     imgclahe_sharped=cv2.filter2D (imgclahe, -1,sharpen_filter)
Step twelve: Generate k_fold
    k_folds=5
    kf = KFold(n_splits=k_folds, random_state=None, shuffle=True)
    kf.get_n_splits(train)
    for train_index, val_index in kf.split(X):
        trainData = X[train_index]
        val_data = X[val_index]
        trainLabels = y[train_index]
        valLabels = y[val_index]
Step Thirteen: Build CNN Model
    model = build_desnet121(IMAGE_SIZE, 3)
    annealer = ReduceLROnPlateau(
    monitor = 'val_accuracy',
    factor    = 0.70,
    patience = 5,
    verbose = 1,
    min_lr = 1e-4
    checkpoint = ModelCheckpoint(
                 'model.h5',
                 verbose = 1,
                 save_best_only = True)
    EarlyStop=EarlyStopping (
      monitor="val_accuracy,"
      patience=7,
      verbose=1,
      mode="max")
Step fourteen: Train the model
    BATCH_SIZE = 8
    EPOCHS = 100
  model.fit(datagen.flow(train data, trainLabels, batch_size =
              BATCH_SIZE,(
              steps_per_epoch = train data.shape[0],
              epochs = EPOCHS,
              verbose = 1,
              callbacks = [annealer, checkpoint, EarlyStop],
              validation_data = (val_data, valLabels))
  End
```

6.1 Dataset

- SARS-CoV-2 CT-scan Dataset: consists of 2482 CT images from 120 patients, including 1252 CT images from 60 SARSCoV-2-infected patients. 1230 CT images of 60 non-infected patients with SARS-CoV-2 from males (32) and females (28), and 1230 CT images of 60 non-infected patients with SARS-CoV-2 but pneumonia diseases from males (30) and females (30). Hospitals in Sao Paulo, Brazil, were surveyed for data. The images in this dataset are digital photographs of printed CT assessments, and there is no standard for image size (the smallest image in the dataset has dimensions of 104 by 153, while the largest image has dimensions of

484 by 416) [26]. Table 2 and Table 3 summarize the SARS-C0V-2 CT and COVID-CT datasets. Figure (2) shows an example of an image of the SARS-CoV-2 CT-scan dataset.

Table 2: The SARS-C0V-2 CT dataset.

No.Image	No. Patient	Infected Patients	Non-Infected Patients	Image Size
2482 CT Images	120 Patients	-1252 CT Imaging (60 Patients) - Males (32) Females (28)	-1230 CT Images (60 Patients) - Males (30) Females (30)	Smallest= (104*153) Largest= (484* 416)

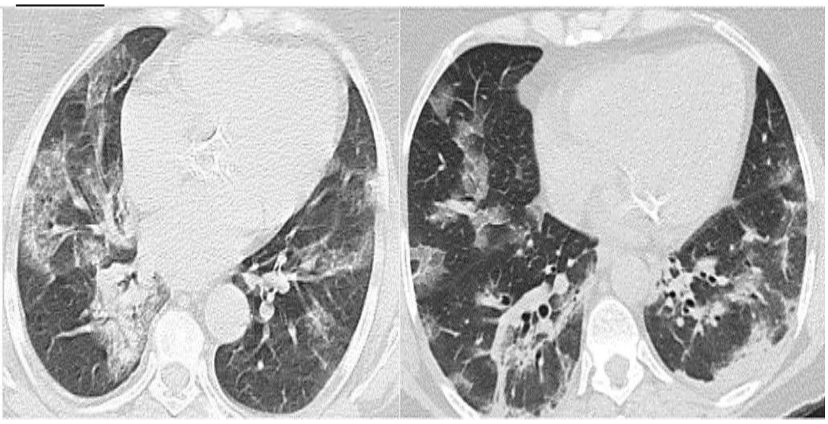

Figure 2: Example of the image in the SARS-C0V-2 CT dataset.

- COVID-CT: Contains 349 COVID-19 CT images from 216 patients in addition to 463 non-COVID-19 CT images. A senior radiologist who has been diagnosing and treating COVID-19 patients since the outbreak of this pandemic confirms the utility of this dataset. Figure (3) shows an example of an image of the COVID-19 CT dataset.

Table 3: Summarize the COVID-19 CT dataset

No.Image	No. Patient	Infected Patients	Non-Infected Patients	Image Size
812 CT images	216 patients	349 CT imaging	463 CT images	The lowest, average, and highest heights are 153, 491, and 1853, respectively. The widths at the lowest, average, and maximum are 124, 383, and 1485, respectively

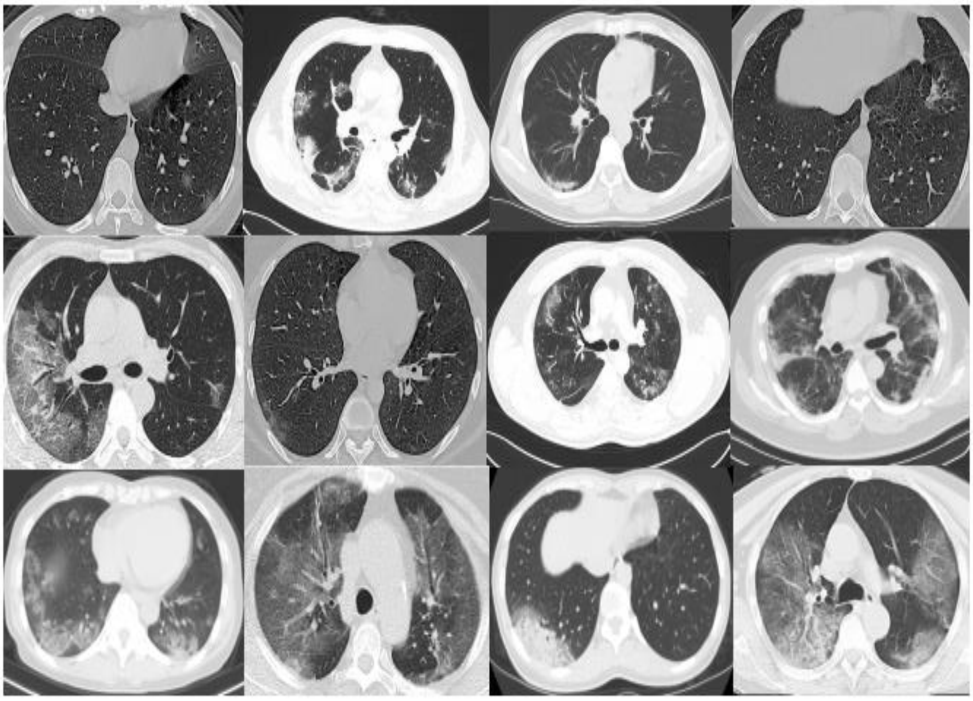

Figure 3: Example of an image in the COVID-19 CT dataset.

6.2 Training and Testing Data

In Figure 4, depending on the size of the dataset, it has been divided into 90% training (2963) and 10% testing (331) by using the Keras library. The reason behind the dataset division is to isolate the test data and not perform any processing on it. After that, the training data was treated as 100% and entered into the k-fold to be divided into the validation part and the training part. The last 5-fold division is the one in which the processing operations will be carried out.

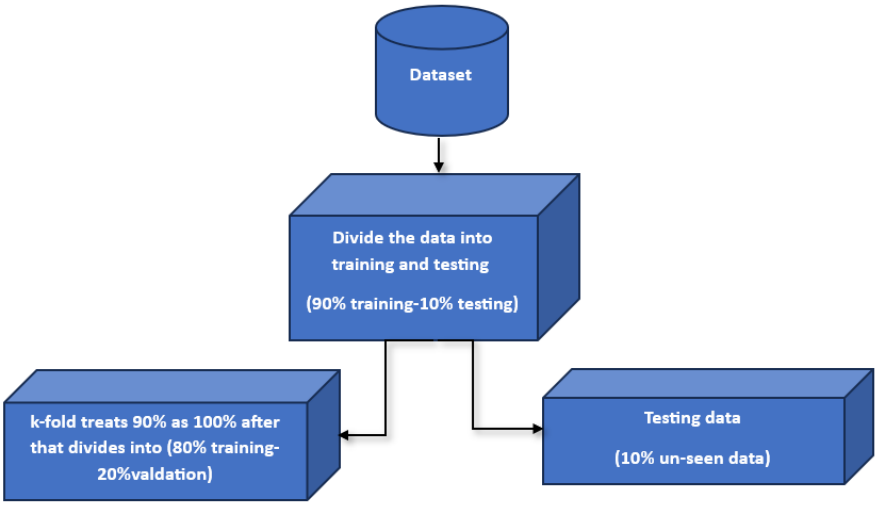

Figure 4: Two stages of dividing the dataset.

6.3 Processing stage based on filters

In the proposed system, three filters were used, which will be applied sequentially, one after the other, on the previously explained data isolated for training. An important note that the images belonging to the data set are colored images consisting of three RGB (Red, Green, Blue) channels, and these images can be directly worked on by filters or need a processing method to deal with the filter, meaning that it is possible to work on the three channels together, or it is possible to work on the three channels separately and then combined like a CLAHE (Contrast Limited Adaptive Histogram Equalization) filter. Three filters were adopted in the proposed system, as shown in Figure 5.

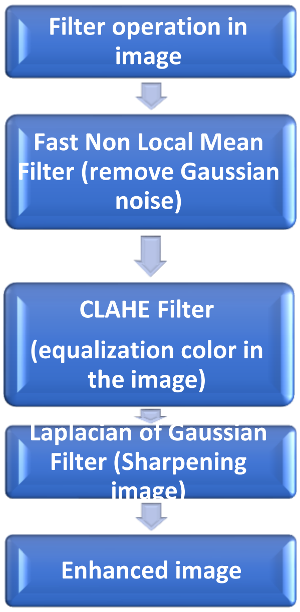

Figure 5: Three steps off filter in the proposed system.

6.4 Stage of Preprocessing

The proposed system goes through several stages of the treatment process, and the following is an explanation of the stages:

- Step one: Denoising filter

In the first step, a filter is used to remove noise in the image, and this filter is distinguished by the fact that it can deal with color images, it deals with the three channels. Figure 6 shows the difference between images after and before using the filter.

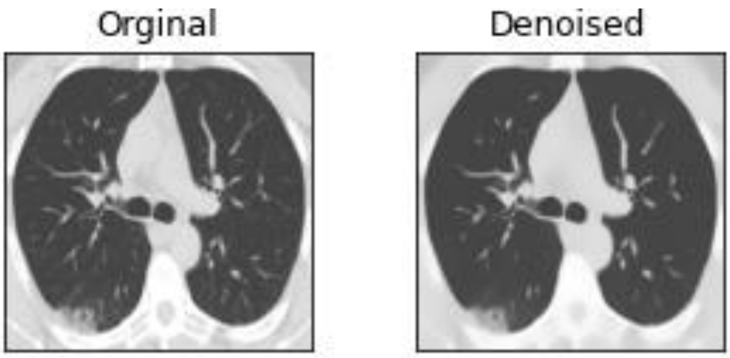

Figure 6: Difference between images after and before using the filter.

- Step Two: CLAHE filter

In this step, the CLAHE filter is used, which naturally works on non-color images; that is, it works on one channel, but in the proposed system, it will work on colored images by dividing it into three channels, each channel separates from the other, taking into account that Python does not work in an RGB sequence, but rather the reverse BGR. Therefore, the division of the image in the proposed system is the result of four images, which are as follows:

CLAHE Filter works on three channels:

- Chanel blue-------first image
- Chanel green-----second image
- Chanel red------ the third image
- RGB-----------------fourth image

The following points summarize the reason for these steps for channel separation and combination:

1. Using the CLAHE filter works on the channels separately, and the step of merging the images is to obtain the processed color images.

2. The benefit of merging the channels in the fourth image is to reverse the order of [BGR to RGB]. Because of what was previously mentioned, Python deals with images in reverse order. Fig. 7 shows that the filter CLAHE is effective in image enhancement.

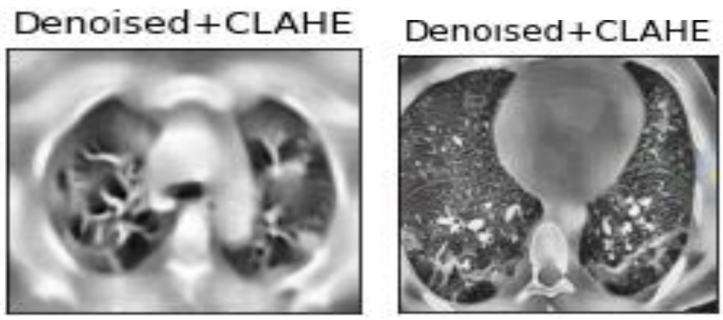

Figure 7: CLAHE filter in the proposed system.

- Step Three: Sharpening Filter

This filter is the last stage in the filtering process, adopted to clarify the borders of the images more accurately. The kernel type of the three types was adopted based on experience because this type serves the images in the proposed system very efficiently, and this filter is in the form of a matrix (3*3).

0	-1	0
-1	5	-1
0	-1	0

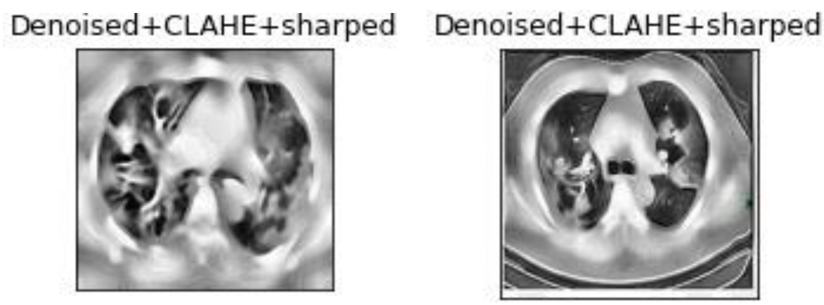

Figure 8: Sharping filter in the proposed system.

This matrix is specific to this filter, and Figure 8 shows the importance of using this filter and its impact on the image resolution in a very distinctive way. Figure 9 shows the three steps that discussed in the above steps:

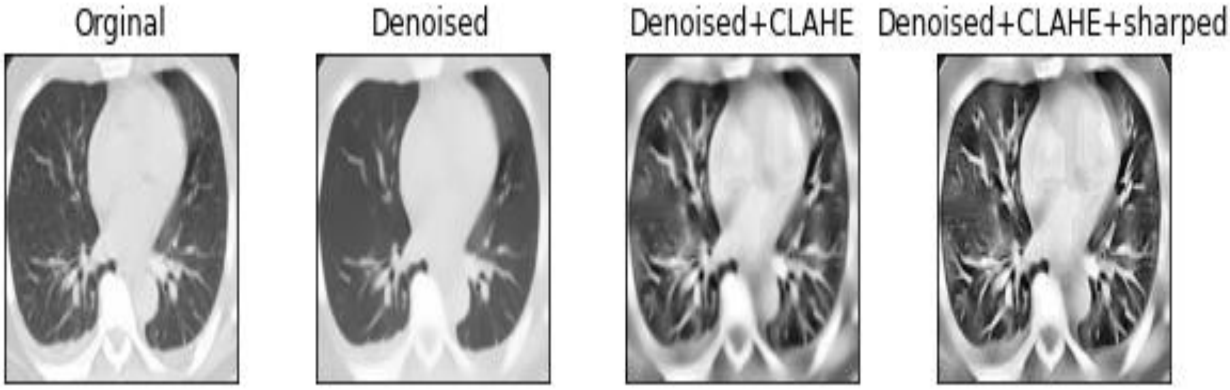

Figure 9: Three steps of the filter.

- Step four: resizing the image.

The final preparation stage of the training process is reuniting the image to size (224 * 224). This size was adopted by experience and its effect on the system accuracy and implementation time. The benefit of this step is to standardize the sizes of the different images to avoid fluctuations in the training process that could cause overfitting. Also, as a step interrelated with it, normalization was made, as the benefit of it is to unify the reading of the values of the collected images.

6.5 Classification by using Densenet-121

In the classification process. Dense net-121 was used, and the main reason is the pre-trained model. The selection of this type of model is based on the experience of some of the pre-trained models that have been approved by previous researchers who were mentioned in previous studies [27]. As this model consists of 121 layers, the idea of these layers is to train in-depth and also to get rid of the degradation condition, as the structure of dense net-121 follows the idea of crossover, linking between input and convolutional each time. It can be summarized the idea of having the max pooling layer is to extract the most vital features that exist. Still, to avoid the degradation that causes it, input is called every time with the features, and this method has been adopted in this pre-trained model. Table 4 shows the order of the layers in the pre-trained model adopted in the proposed system.

Table 4: The layer in the proposed model**.**

Layer (type)	Output shape	Param #
Input_2 (InputLayer)	[(None, 224, 224, 3)]	0
Conv2d (conv2D)	(None, 224, 224, 3)]	84
Densenet121 (functional)	(None, None, None, 1024)	7037504
Global_average_pooling2d lobalAveragePooling2D)	(G (None, 1024)	0
Dense (dense)	(None, 64)	65600
Dense_1 (dense)	(None, 2)	130

Table 5: details of train and testing in the proposed system**.**

X_train	Y_train	X_test	Y_test
(2963, 224, 224, 3)	(2963, 2)	(331, 224, 224, 3)	(331, 2)

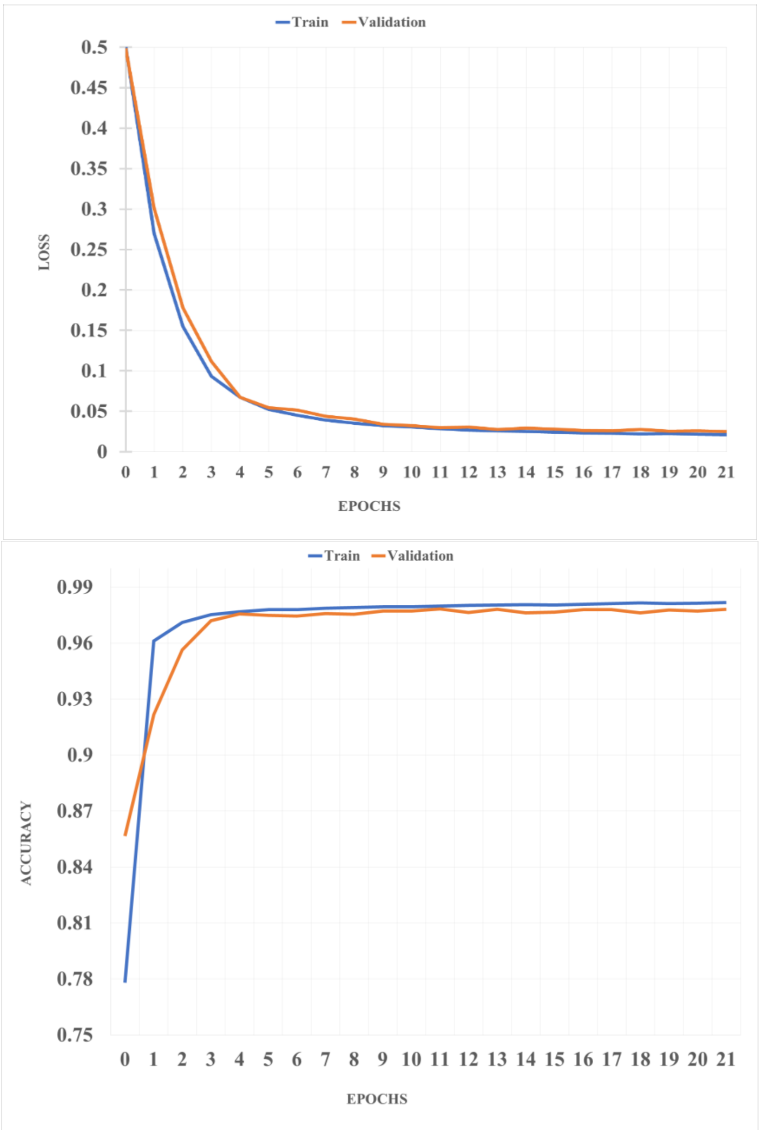

Figure 10: Result of accuracy and loss in the proposed system.

6.6 GUI and prediction

The proposed system is supported by user interfaces that enable users to use it easily, whether they have computer experience. The destinations also contain all the details of the proposed system, including training, testing, and the confusion matrix.

7. Conclusions

The coronavirus pandemic was a fundamental and influential event in the world, as it worked to change the features of the world by imposing new preventive habits, new treatment methods, and new detection methods since its inception in 2019. The alarm bell rang, and to this day, this epidemic was caused by a virus that affects the bronchi, lungs, and breathing, which is the cause of many deaths and is still in the process of transformation and development, but with fewer media exposures. From this point of view, a medical system was proposed to detect the presence of the Coronavirus in the lungs based on CT scans. He made simulated processing steps for this data, and then a classification was made based on the presence of infection from the absence of infection and its percentage. Choosing deep learning was a good step for the proposed system results. Prepared as future works, this is an experiment on other pre-trained models and an evaluation of data from hospitals in a realistic way. The proposed system obtained highaccuracy results of up to 99% in real terms and provided the system with interfaces that support the user to facilitate work on it.

References

[1] A. A. Mohsen, H. F. AL-Husseiny, K. Hattaf and B. Boulfoul, “A mathematical Model for the Dynamics of COVID-19 Pandemy Involving,” *Iraqi Journal of Science,* vol. 62, no. 1, pp. 295-307, 2021.

[2] A. Hafeez, S. Ahmad, S. A. Siddqui, M. Ahmad and S. Mishra, “A Review of COVID-19 (Coronavirus Disease-2019) Diagnosis, Treatments and Prevention,” *EJMO,* vol. 4, no. 2, p. 116–125, 2020.

[3] M. M. ISLAM, F. KARRAY, R. ALHAJJ and J. Zeng, “A Review on Deep Learning Techniques for the Diagnosis of Novel Coronavirus (COVID-19),” *IEEE Access,* vol. 9, pp. 30551-30572, 2021.

[4] G. Herpe, M. Lederlin, M. Naudin, M. Ohana, K. Chaumoitre, J. Gregory, V. Vilgrain, C. A. Freitag, C. De Margerie-Mellon, V. Flory, M. Ludwig, L. Mondot, I. Fitton, A. R. R. Jacquier, P. Ardilouze, I. Petit, A. Gervaise, O. Bayle, A. Crombe, M. M. Sokeng, C. Thomas, G. Henry, V. Bliah, T. Le Tat, M.-S. Guillot, P. Gendrin, M. Garetier, E. Bertolle, C. Montagne, B. Langlet, A. Kalaaji, H. Kayayan, F. Desmots, B. Dhaene, P.-J. Saulnier, R. Guillevin, J.-M. Bartoli, J.-P. Beregi and J. P. Tasu, “Efficacy of Chest CT for COVID-19 Pneumonia Diagnosis in France,” *Radiology,* vol. 298, no. 2, pp. E81-E87, 2021.

[5] M. M. Mijwil, “Deep Convolutional Neural Network Architecture to Detect COVID-19 from Chest X-Ray Images,” *Iraqi Journal of Science,* vol. 64, no. 5, pp. 2561-2574, 2023.

[6] A. Kovács, P. Palásti, D. Veréb, B. Bozsik, A. Palkó and Z. T. Kincses, “The sensitivity and specificity of chest CT in the diagnosis of COVID-19,” *European Radiology,* vol. 31, no. 5, pp. 2819-2824, 2021.

[7] H. Abu Owida, H. S. Migdadi, O. S. M. Hemied, N. F. F. Alshdaifat, S. F. A. Abuowaida and R. S. Alkhawaldeh, “Deep learning algorithms to improve COVID-19 classification based on CT images,” *Bulletin of Electrical Engineering and Informatics,* vol. 11, no. 5, pp. 2876-2885, 2022.

[8] M. F. Aslan, K. Sabanci and E. Ropelewska, “A New Approach to COVID-19 Detection: An ANN Proposal Optimized through Tree-Seed Algorithm,” *Symmetry,* vol. 14, no. 7, pp. 96-100, 2022.

[9] F. Mehboob, A. Rauf, R. Jiang, A. K. J. Saudagar, K. M. Malik, M. B. Khan, M. H. Abdul Hasnat, A. AlTameem and M. AlKhathami, “Towards robust diagnosis of COVID-19 using vision self-attention transformer,” *Scientific Reports,* vol. 12, p. 8922, 2022.

[10] B. Xu, D. Martín, M. Khishe and R. Boostani, “COVID-19 diagnosis using chest CT scans and deep convolutional neural networks evolved by IP-based sine-cosine algorithm,” *Medical & Biological Engineering & Computing,* vol. 60, pp. 2931-2949, 2022.

[11] A. A.-R. Hussien and N. A. Z. Abdullah, “A Review for Arabic Sentiment Analysis Using Deep Learning,” *Iraqi Journal of Science,* vol. 64, no. 12, pp. 6572- 6585, 2023.

[12] N. H. Ali, E. A. Matheel and E. A. Akbas, “Learning evolution: A survey,” *Iraqi Journal of Science,* vol. 62, no. 12, pp. 4978-4987, 2021.

[13] F. M. Shah, S. K. Saha Joy, F. Ahmed, T. Hossain, M. Humaira, A. S. Ami, S. Paul, A. R. K. Jim and S. Ahmed, “A Comprehensive Survey of COVID-19 Detection Using Medical Images,” *SN Computer Science,* vol. 2, no. 6, p. 434, 2021.

[14] H. Majid and K. H. Ali, “Automatic Diagnosis of Coronavirus Using Conditional Generative Adversarial Network (CGAN),” *Iraqi Journal of Science,* vol. 64, no. 7, pp. 3642-3656, 2023.

[15] H. Liu and B. Lang, “Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey,” *Applied Sciences,* vol. 9, no. 20, p. 4396, 2019.

[16] R. F. Abbas and E. A. Matheel, “Residual Network with Attention to Neural Cells Segmentation,” *Iraqi Journal of Science,* vol. 64, no. 4, pp. 2023-2036, 2023.

[17] Q. Bi, K. . E. Goodman, J. Kaminsky and J. Lessler, “What is Machine Learning? A Primer for the Epidemiologist,” *American Journal of Epidemiology,* vol. 188, no. 12, pp. 2222-2239, 2019.

[18] W. K. Yousif and A. A. Ali, "Simulation of Pose to Pose Moving of the Mobile Robot with Specified GPS Points," *Journal of Engineering*, vol. 26, no. 11, pp. 195-208, 2020.

[19] S. Piramuthu, "Financial credit-risk evaluation with neural and neurofuzzy systems," *European Journal of Operational Research*, vol. 112, no. 2, pp. 310-321, 1999.

[20] S. Alkentar, B. Alsahwa, A. Assalem and D. Karakolla, "Practical comparation of the accuracy and speed of YOLO, SSD and Faster RCNN for drone detection," *Journal of Engineering,* vol. 27, no. 8, pp. 19-31, 2021.

[21] W. Jiang, N. Synovic, M. Hyatt, T. R. Schorlemmer, R. Sethi, Y.-H. Lu, G. K. Thiruvathukal and J. C. Davis, "An Empirical Study of Pre-Trained Model Reuse in the Hugging Face Deep Learning Model Registry," *arXiv preprint arXiv:2303.02552*, vol. 1, 2023.

[22] D. Wu, Y. Ying, M. Zhou, J. Pan and D. Cui, "Improved ResNet-50 deep learning algorithm for identifying chicken gender," *Computers and Electronics in Agriculture*, vol. 205, p. 107622, 2023.

[23] J. Zhang, Y. Zhang, Y. Jin, J. Xu and X. Xu, "MDU-Net: multi-scale densely connected U-Net for biomedical image segmentation," *Jiawei Zhang*, vol. 11, no. 1, p. 13, 2023.

[24] N. Alsharabi, T. Shahwar, A. U. Rehman and Y. Alharbi, "Implementing Magnetic Resonance Imaging Brain Disorder Classification via AlexNet–Quantum Learning," *Mathematics*, vol. 11, no. 2, p. 376, 2023.

[25] A. Ijaz, B. Raza, I. Kiran, A. Waheed, A. Raza, H. Shah and S. Aftan, "Modality Specific CBAM-VGGNet Model for the Classification of Breast Histopathology Images via Transfer Learning," *IEEE Access*, vol. 11, pp. 15750-15762, 2023.

[26] E. Soares, P. Angelov, S. Biaso, M. H. Froes and D. K. Abe, "SARS-CoV-2 CT-scan dataset: A large dataset of real patients CT scans for SARS-CoV-2 identification," *medRxiv*, 2020.

[27] F. M. S. Al-Zwainy and N. T. Hadal, "Application Artificial Forecasting Techniques in Cost Management (review)," *Journal of Engineering*, vol. 22, no. 8, pp. 1-15, 2016.