A Robust CNN Model for Diagnosis of COVID-19 Based on CT Scan Images and DL Techniques

Ahmed H. Eldeeb, Mohammed Nagah Amr, Amin S. Ibrahim, Hesham Kamel, and Sara Fouad

Abstract—**The 2019 Coronavirus (COVID-19) virus has caused damage on people's respiratory systems over the world. Computed Tomography (CT) is a faster complement for RT-PCR during peak virus spread times. Nowadays, Deep Learning (DL) with CT provides more robust and reliable methods for classifying patterns in medical pictures. In this paper, we proposed a simple low training proposed customized Convolutional Neural Networks (CNN) customized model based on CNN architecture that layers which are optionals may be included such as the layer of batch normalization to reduce time taken for training and a layer with a dropout to deal with overfitting. We employed a huge dataset of chest CT slices images from diverse sources COVIDx-CT, which consists of a 16,146-image dataset with 810 patients of various nationalities. The proposed customized model's classification results compared to the VGG-16, Alex Net, and ResNet50 Deep Learning models. The proposed CNN model shows robustness by achieving an overall accuracy of 93% compared to 88%, 89%, and 95% for the VGG-16, Alex Net, and ResNet50 DL models for the classification of 3 classes. When this relates to binary classification, the classification accuracy of the proposed model and the VGG-16 models were identical (almost 100% accurate), with 0.17% of misclassification in the class of Non-Covid-19, the Alex Net model achieved almost 100% classification accuracy with 0.33% misclassification in the class of Non-Covid-19. Finally, ResNet50 achieved 95% classification accuracy with 5% misclassification in the Non-Covid-19 class.**

Keywords—**Deep learning; Covid-19; Artificial Intelligence; Computed Tomography; Convolutional Neural Networks**

I. INTRODUCTION

THE 2019 Coronavirus (COVID-19) epidemic became a global health crisis, infecting millions of people and wreaking devastation on the global economy. COVID-19 damages the lungs in the most severe cases, causing Pneumonia. The limitation of testing capacity in addition to, the lack of face masks and mechanical ventilators worldwide [1,2]. The image of the lungs on a chest radiograph, including a chest X-Ray (CX-Ray), Computed Tomography (CT) scan, ultrasound, and others [3, 4], can be used to diagnose If the virus or bacterium or COVID-19 epidemic infects lungs. COVID-19 infection, on the other hand, can be detected immediately using imaging scans at the inside of the lungs. This may give rise to the sick person being isolated early on, preventing the virus from spreading.

COVID-19 can also be diagnosed using traditional laboratory techniques, such as a set of reverse transcript Polymerase Chain Reaction Tests (RT-PCR). The test of RT-PCR takes longer and can result in false negatives. [5]. COVID-19 infection, on the other hand, can be detected immediately using imaging scans at the inside of the lungs. This may give rise to the sick person being isolated early on, preventing the virus from spreading.

CT is already suggested as a part of the most crucial detection ways that can be used as an adjunct to the test by RT-PCR, especially in cases where patients are subjected to routine procedures. The abnormal chest CT image scans is the characteristic step in determining whether or not patients have CoVID-19. However, these anomalies can be challenging to classify, considering other lung diseases. [6]. In this research, we identify Corona-Virus Disease Convolutional Neural Network (COVID-19 CNN). COVID-19 CNN is an adjusted with machine- driven design exploration approach to classify the case of COVID-19 from chest CT images.

Machine Learning (ML) and Deep Learning (DL), two important sections of Artificial Intelligence (AI), have been thoroughly applied in recognizing COVID-19 positive cases. The goal of DL approaches is to learn hierarchical features from data. DL, a part of ML, utilizes algorithms to handle data and imitate the thinking operation or to improve abstractions. The structure of DL is based on stages of calculation layers to deal with data, recognize speech language, and externally perceive things. Each layer receives data, and the output of the previous layer feeds into the next layer. The input layer is the first layer in a network, and the output layer is the last. The layers in between are referred to as hidden (covered-up) layers. Every layer is regularly a basic, uniform algorithm having one type of activation function [7,8].

DL algorithms have been shown to be effective at analyzing and defining infectious regions in radiological images quickly. CNN is the most common DL-used model. It is commonly used with two dimensions' data with a grid pattern as pictures. CNN is mathematically structured from three layers. The first two layers, convolutional and max-pooling, extract the features from the images, while working on a completely connected layer, retrieving features to be mapped to the final output [7,9].

In [10] the author calculates the effects of fine-tuning by using three types of pre-trained CNN models in the classification of COVID-19 using TL (Transfer learning):

Ahmed H. Eldeeb, Mohammed Nagah Amr and Hesham Kamel are with Electronics and Communications Department, School of Engineering, Canadian Higher Engineering Institute, Giza, Egypt (ahmed_eldeib@ieee.org, moh_nagah@cic-cairo.com, Hesham.kamel@ gmail.com).

Amin S. Ibrahim is with Electronics and Communications Engineering Department, Thebes Higher Institute for Engineering, Cairo, Egypt (email: engamin_s84@yahoo.com).

Sara Fouad is with Electronics and Communications Engineering Department, The Higher Institute of Engineering, Modern Academy, Cairo, Egypt (email: sarafouad2012.s.m.a@gmail.com).

SqueezeNet, AlexNet, and Google Net. Three legitimate COVID-19 CX-Ray databases or Kaggle were used to compile the publicly available dataset. According to this research, these pre-trained CNN models based on fine-tuning between them could result in superior performance in classification that doesn't need any preprocessing steps such as data augmentation. Additionally, a fine-tuning model achieved a lower training time when compared to a variety of other pre-trained models. Six datasets were created to evaluate the binary class and three class performances by applying the different proportions of training and testing sets. On two datasets for binary classification with 50% training and testing sets, the three TL-based fine-tuned models demonstrated the best performance metrics. The proposed fine-tuned models based on three TL showed the greatest performance metrics when applied on two datasets in the case of binary class with 50% training and testing sets.

The article [11] developed an automatic COVID-19 detection framework by evaluating the classification performance of five pre-trained existing CNN models. ResNet-50, ResNet-152, ResNet-101, Inception-ResNet-V2, and Inception-V3 are the names of these models. The framework used four separate classes to accomplish three binary classes (COVID-19 vs. normal, COVID-19 vs. viral Pneumonia, and COVID-19 vs. bacterial Pneumonia) (bacterial Pneumonia, viral Pneumonia, COVID, and normal). On all three datasets, the ResNet-50 outperformed the others [12], according to Dr. Joseph's open-source GitHub repository.

In the COVID-19 epidemic, a trained model based on datasets of images of chest X-rays and CT scans is presented by Jaiswal et al. [13]. The COVIDPEN model is a type of TL method that employs the Pruned Efficient Net, and it has been examined on CT imaging scans and radiographic pictures. Aside from the classification job, the authors have presented a way for interpreting prediction utilizing the local interpretable model-agnostic explanation, which is an image segmentation methodology for representing local and regional features. Wang L and Wong A [14] developed the COVID-Net, a deep CNN model based on the COVIDx dataset. The large model based on million trained parameters is named COVID-Net, which is trained using five datasets to recognize COVID, non-COVID in a binary class, and pneumonia patients into 3 classes.

In this study [15], an automatic detection method for COVID-19 using CX-Rays was achieved using a pre-trained applied for inception-V3 DCNN model with Transfer Learning (TL). Pneumonia and normal CX-Rays were used to diagnose and screen COVID-19 patients. The dataset was unbalanced, with samples distributed as follows: Normal = 1341 , Pneumonia = 1345, and COVID-19 = 864 CX-Ray's .Dr. Joseph's GitHub repository [12], SIRM [16], and the Database of COVID-19 Radiography [17] were used to compile the data. The model was tested for accuracy and cross-entropy loss, and it produced the best results when employing TL. It was also simple to use Markus and colleagues. An automatic approach for quantifying COVID-19 infected chest regions on thick CT sections was established by Li et al. [18].

This study [19] introduces C-COVIDNet based on the model named CNN and image processing, which is used as a basis for the COVID-19 detection model that has been trained by scanning pictures of a chest X-ray dataset divided into three classes: Pneumonia, normal, and COVID-19 individuals. By identifying the Region of Interest (ROI), an effective preprocessing pipeline for images is built to provide adjusted and accurate data about segments of the lung. Inverted binary thresholding, Otsu's thresholding, Gaussian blurring, histogram equalization, and removal of the noise of background are all included in the picture preparation package.

The best accuracy of the training and F1 scores were 97.5 percent and 97.91 percent, respectively. The model was checked versus two free-to-access datasets, Corona Hack-Chest X-Ray Dataset and COVID-19 Patients Lungs X-Ray Images 10000, with 97.7% and 98.9% testing accuracy, respectively.

CNN is the core of this work as we have used it to classify types of diseases in lungs from CT scan images by training the model to extract features effectively from a huge training dataset using based on well-known models and the proposed customized model, which has achieved a better precision regarding the covid-19 Class than AlexNet model [20] and Resnet50 model [21] and approximately similar precision to VGG-16 model [22].

This study is based on implementing a simple CNN model to reduce the training time. As the CNN architecture may include optional layers such as the batch normalization layer to reduce the time taken for training and a layer with a dropout to deal with overfitting. We used Google Collaborator and Python programming language with Keras as Application Programming Interfaces (API) to Tensor flow in order to construct the proposed CNN model [23,24]. Some of the study's significant conclusions include the following:

i. We employed a huge dataset of chest CT slices images from diverse sources COVIDx-CT.
ii. Data augmentation decreases the tendency of deep learning models to overfit the training data by applying geometric manipulations on training data to augment and expand the dataset.
iii. The performance of the suggested CNN model has been shown to be statistically significant in comparison to the performance of other deep learning models.
iv. CT scanning procedure has a faster turnaround time.

The following is how the paper is structured: The dataset utilized in this study, as well as the distribution of classes within it, are shown in Section II. The description of preprocessing of the dataset shown in Section III is used to prepare the data for DL processing. The suggested model architecture and implementation technique are discussed in Section IV. We present the results of the proposed model compared to the other DL models as shown in section V. Finally, the conclusion brings current and future efforts

II.   DATASET AND DISTRIBUTION

A robust model's training process is heavily reliant on the dataset. As a result, we employed a huge dataset of chest CT slices images from diverse sources COVIDx-CT [25]. Figure 1 shows an open-access dataset of 194,922 CT slices from 3,745 patients from the Kaggle benchmark dataset. We utilized a python library Matplotlib [26] for data visualization and graph plotting to visualize certain insights about the data such as patient genders and findings in pie charts and patient ages in relation to the number of cases in the histogram presented in Figures 2 and 3 receptively.
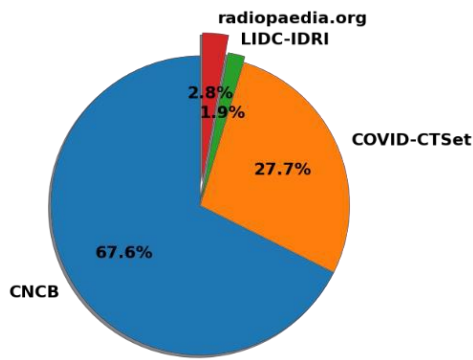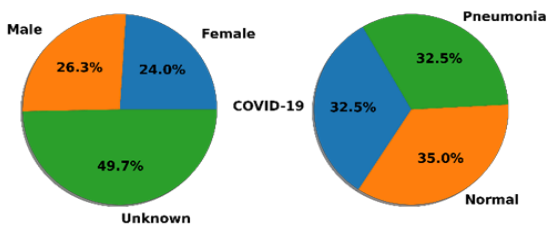
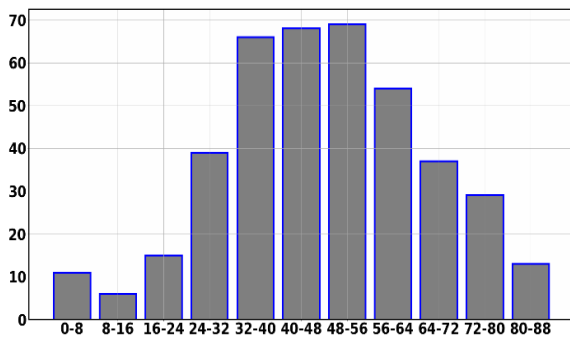Fig. 1. COVIDx-CT Data sources

Fig. 2. Patient`s genders and findings

Fig.-3. patient ages for the number of cases

Fig. 4. Patient`s Nationalities

Fig. 5. The distribution of each infection data set

Fig. 6. Samples from the used dataset

A molecular diagnostic test conducted in a typical laboratory would take a longer time to complete than a CT scanning process, which can also provide more thorough information about the pathology. As shown in the left pie chart in Figure 4, we worked on an appropriate piece of the COVIDx CT-2 Dataset [25], which consists of a 16,146-image dataset with 810 patients of various nationalities. These images are categorized into normal, pneumonia, and covid-19 images.The images were then separated into 12,249, 1,881, and 2,016 for training, validation, and testing, respectively, as illustrated in Figure-5. The weights of the constructed deep convolutional neural network are trained using the training set. In contrast, the validation set is applied to check the model's generalization potential, and the weights for the model are picked to set aside based on the validation set's loss function value. The model's test results are obtained by running the saved model weights in the test set. The selected part of the data set for each infection type is shown in Figure 6 from a sample of CT images [25].
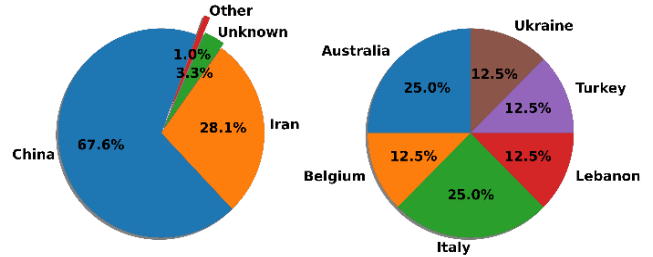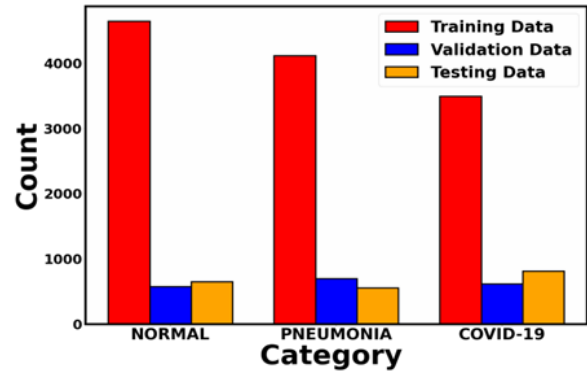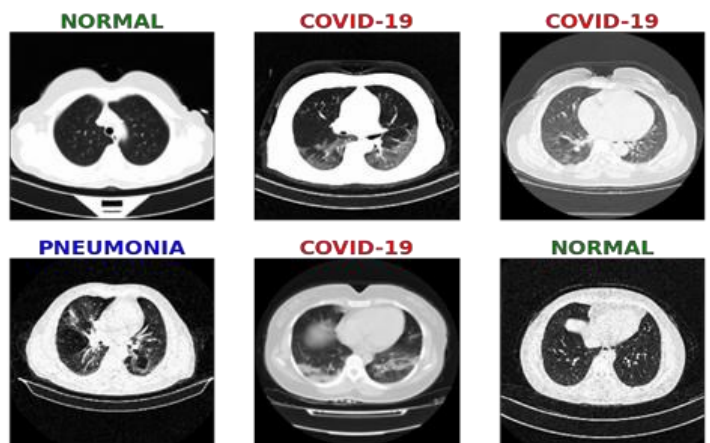
III. DATASET PRE-PROCESSING

In most cases, the pre-processing data stage is utilized to prepare the input data to meet the deep learning model's criteria. Multiple pre-processing processes were used to prepare the input images in our approach. At first, the images are resized to 224x224 pixels; secondly, it has converted into greyscale; thirdly, it is standardized in the range of 0-1 to make the model converge faster. The DL models are usually thought to benefit from data augmentation techniques as they can enhance the model to be robust and skillful. It decreases the tendency of deep learning models to over fit the training data by applying geometric manipulations on training data to augment and expand the dataset. Finally, we have performed random combinations of augmentation methods such as width shift, height shift, rotation, and zoom with ranges of 0.1, 0.1, 0.1, and 5, respectively, as shown in Figure-7.
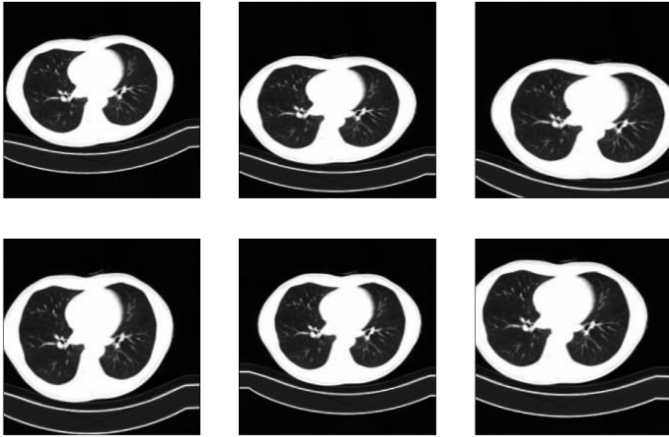
Fig. 7. Data augmentation sample

IV.   THE PROPOSED MODEL

This section will cover implementation, and model training for the suggested technique. The system model is shown in the following Figure 8. Firstly, the convolution layer is the center structure square of a CNN that utilizes convolution activity instead of general grid augmentation. Its boundaries comprise a bunch of learnable Channels, otherwise called portions. The entire undertaking of the convolutional layer is to identify highlights found inside neighbourhood districts of the information picture that are normal all through the dataset then plan their presence on an element map. The convolution activity is given as

$$F(i. j) = (I * K)(i. j)$$
$$= \sum m \sum n \quad I(i + m. j + n)K(m. n) \qquad (1)$$

The image is input matrix labeled as I, K is the filter of Two Dimensional (2D) size (m x n), and F is the 2D feature map output of size (i, j). An activation function Rectified Linear Unit (ReLU) introduces nonlinearity by taking the output of each convolutional layer. In essence, ReLU thresholds the input at zero in order to calculate the activation. If the input is smaller than 0, ReLU outputs 0, and if not, it outputs raw data. it is mathematically given as

$$f(x) = max(0. X) \qquad (2)$$

Padding is a term relevant to CNN as it alludes to the measure of pixels added to a picture when the piece of a CNN is preparing it. For instance, assuming the cushioning in a CNN is set to nothing, every pixel of value-added will be worth zero. Assuming, be that as it may, the zero cushioning is set to one, there will be a one-pixel line added to the picture with a pixel worth of nothing.

Second, an optional pooling or down-inspecting layer is added after each convolution layer in order to reduce the spatial size of the data and, as a result, the number of network boundaries. A pooling layer sums up a particular location of neurons in the convolution layer by using each feature map output from the convolutional layer as an example. Max Pooling is the most common pooling method and it produces the most valuable results in the information space. L2-norm pooling and average pooling are further options for pooling.
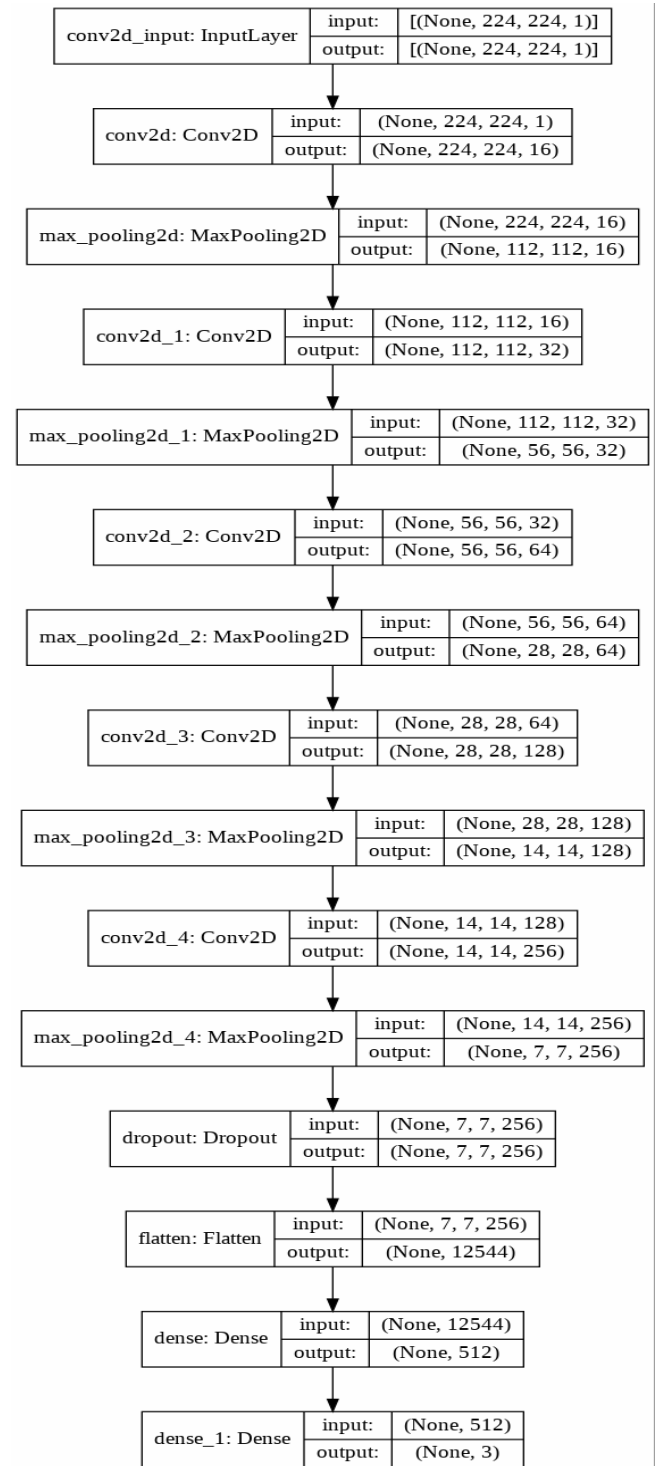

Fig. 8. The proposed system model

Every neuron from the previous layer is coupled to every neuron in the layer above it in a fully connected layer. Each value is determined by anticipating how closely it will fit a particular class. The class scores are then produced using an activation function using the output of the final fully connected layer. The two main classifiers used by CNN are SoftMax and Support Vector Machines (SVM). The following is the SoftMax work that registers the probability distribution of the n yield classes:

$$Z^k = \frac{e^{x^k}}{\sum_{i=1}^{n} e^{x^n}} \qquad (3)$$

Where the input and output vectors, respectively, are x and Z. All outputs added together (Z) equal 1.

To create a complete CNN architecture, all of the layers outlined previously are piled on top of one another. The batch normalization, which shortens training time, and the dropout layer, which addresses overfitting, are examples of optional layers that CNN may contain. The proposed CNN model comprises a total of 6,668,168 parameters, 6,667,496 of which are trainable, and 672 of which are not. The structure of the model regarding the used layers, the output shape, and the number of the parameters at each layer is presented in Table I.

TABLE I
THE PROPOSED MODEL STRUCTURE

Layer (type)	Output Shape	Param #
conv2d_20 (Conv2D)	224 x 224 x 16	160
batch_normalization_12 (Batch Normalization)	224 x 224 x 16	64
max_pooling2d_20 (Max Pooling 2D)	112 x 112 x 16	0
conv2d_21 (Conv2D)	112 x 112 x 32	4640
max_pooling2d_21 (Max Pooling 2D)	56 x 56 x 32	0
conv2d_22 (Conv2D)	56 x 56 x 64	18496
batch_normalization_13 (Batch Normalization)	56 x 56 x 64	256
max_pooling2d_22 (Max Pooling 2D)	28 x 28 x 64	0
conv2d_23 (Conv2D)	28 x 28 x 128	73856
max_pooling2d_23 (Max Pooling 2D)	14 x 14 x 128	0
conv2d_24 (Conv2D)	14 x 14 x 256	295168
max_pooling2d_24 (Max Pooling 2D)	7x 7 x 256	0
batch_normalization_14 (Batch Normalization)	7x 7 x 256	1024
dropout_8 (Dropout)	7x 7 x 256	0
flatten_4 (Flatten)	12544	0
dense_8 (Dense)	500	6272500
dropout_9 (Dropout)	500	0
dense_9 (Dense)	4	2004

Total parameters: 6,668,168

Trainable parameters: 6,667,496

Non-trainable parameters: 672

V. RESULTS AND DISCUSSION

This section shows the proposed customized model's classification results compared to the VGG-16, Alex Net, and ResNet50 Deep Learning models. The proposed model was trained, validated, and tested according to the dataset distribution previously shown in Figure 5 for 3-class classification. In addition, a smaller dataset was prepared by dividing the previous dataset into 3000 samples for COVID-19 and random 3000 samples from other classes (Non-COVID-19 Class) for binary classification. Using the RMSprop optimizer with an initial learning rate of 0.001, a patch size of 32, and 26 epochs, the proposed model has been trained on the prepared dataset. We used Train-Test-Split to assess the effectiveness of our 3-class model with a split percentile of 20% and 80% respectively for training and testing. By applying this procedure, the training time is reduced. The accuracy and loss plots on the training and validation sets over the training epochs were used to report the final performance of the suggested CNN model, as illustrated in Figures 9a and b.
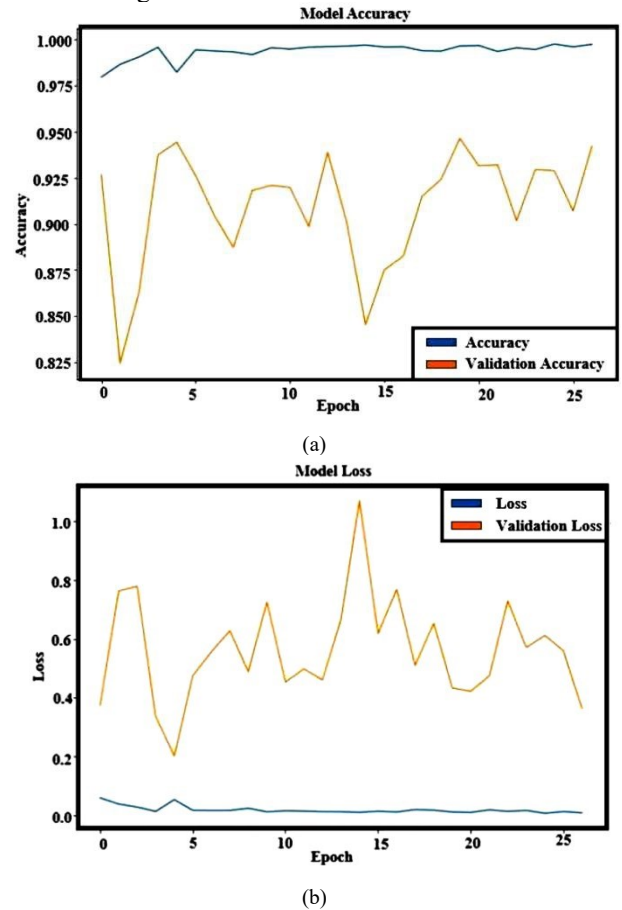

(a)

(b)

Fig. 9. The performance of the proposed model over the training epochs: (a) Training and validation accuracy plots. (b) Training and validation loss plots

A Confusion Matrix is the form to compare the proposed model's classification performance against that of other well-known models (CM). When a point is misclassified by the classifier, it is represented by an off-diagonal element. Diagonal elements are the number of points for which the predicted class is equal to the true class. The confusion matrix's diagonal values should be as high as possible
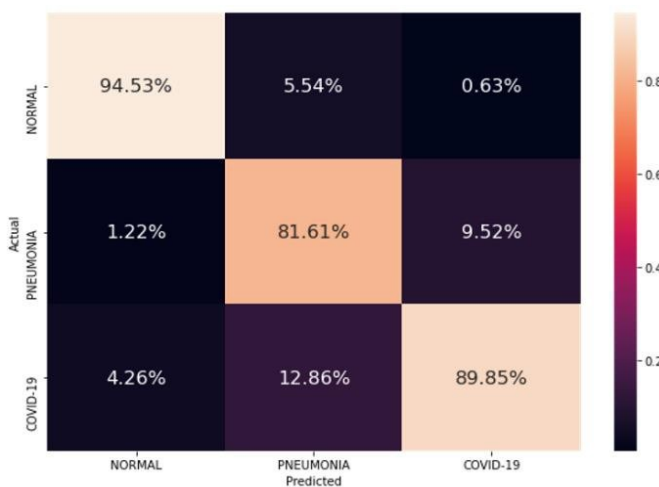
because this indicates several accurate predictions. The suggested CNN model's, VGG-16's, Alex Net's, and ResNet50's respective confusion matrices are shown in Figures 10a, b, c, and d.
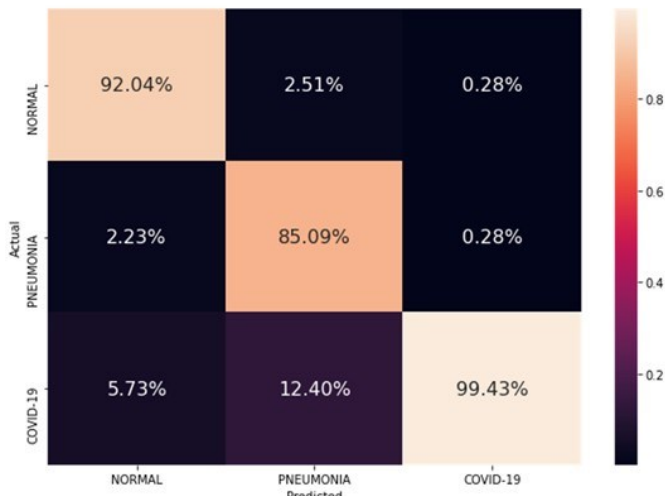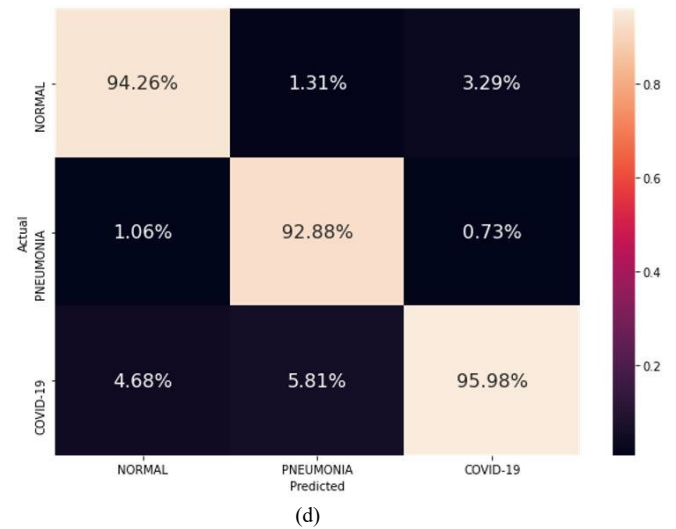

(a)

(b)

(c)

(d)

Fig. 10. The Confusion Matrix of: (a) The proposed model. (b) VGG-16. (C) Alex Net. (D) ResNet50

The overall accuracy, precision, recall, and F1 score are furthermore the most crucial measures to assess the effectiveness of a classification algorithm. The following expressions calculate those metrics:

$$Accuracy = \frac{The\ number\ of\ correctly\ classified\ images}{Total\ images\ number} \quad (4)$$

$$Precision = \frac{Sum\ of\ all\ True\ Positives\ (TP)}{Sum\ of\ all\ True\ Positives\ (TP) + All\ False\ Positives\ (FP)} \quad (5)$$

$$Recall = \frac{Sum\ of\ all\ True\ Positives\ (TP)}{Sum\ of\ all\ True\ Positives\ (TP) + All\ False\ Negatives\ (FN)} \quad (6)$$

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (7)$$

The class-wise performance metrics of the proposed CNN model, VGG-16, Alex Net, and ResNet50, the Tables II, III, IV, and V, are respectively, present.

TABLE II
THE PERFORMANCE METRICS OF THE PROPOSED CNN MODEL

Class/ Metric	Precision (%)	Recall (%)	F1-Score (%)
NORMAL	92	97	95
PNEUMONIA	85	97	90
COVID-19	99	86	92
Overall Accuracy (%): 93			

TABLE III
THE PERFORMANCE METRICS OF THE VGG-16 MODEL

Class/ Metric	Precision (%)	Recall (%)	F1-Score (%)
NORMAL	89	95	92
PNEUMONIA	76	96	85
COVID-19	100	78	87
Overall Accuracy (%): 88			

Class/ Metric	Precision (%)	Recall (%)	F1-Score (%)
NORMAL	95	95	95
PNEUMONIA	82	84	83
COVID-19	90	88	89
Overall Accuracy (%): 89			

TABLE V
THE PERFORMANCE METRICS OF THE RESNET50 MODEL.

Class/ Metric	Precision (%)	Recall (%)	F1-Score (%)
NORMAL	94	95	95
PNEUMONIA	93	97	95
COVID-19	96	93	84
Overall Accuracy (%): 95			

The proposed customized model was on a test set consisting of 676 normal, 526 pneumonia, and 814 Covid-19 CT images. The VGG-16 model was on a test set consisting of 669 normal, 529 pneumonia, and 818 Covid-19 CT images. The Alex Net model was on a test set consisting of 658 normal, 541 pneumonia, and 817 Covid-19 CT images. Finally, The ResNet50 model was on a test set consisting of 658 normal, 509 pneumonia, and 849 Covid-19 CT images.s can be noticed, the test set consists of 2016 CT images in total. The proposed CNN model shows robustness by achieving an overall accuracy of 93% compared to 88%, 89%, and 95% for the VGG-16, Alex Net, and ResNet50 DL models, respectively. Nevertheless, the proposed customized CNN model has extremely lower parameters than the other well-known DL models. Table VI shows the number of parameters of the proposed model and the other DL models.
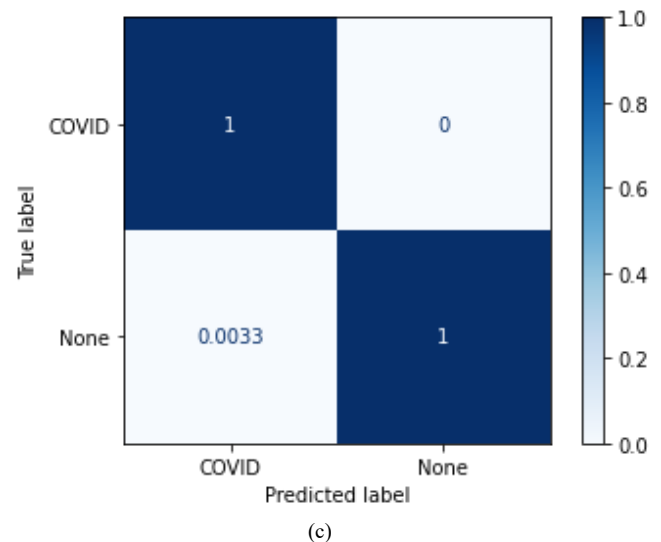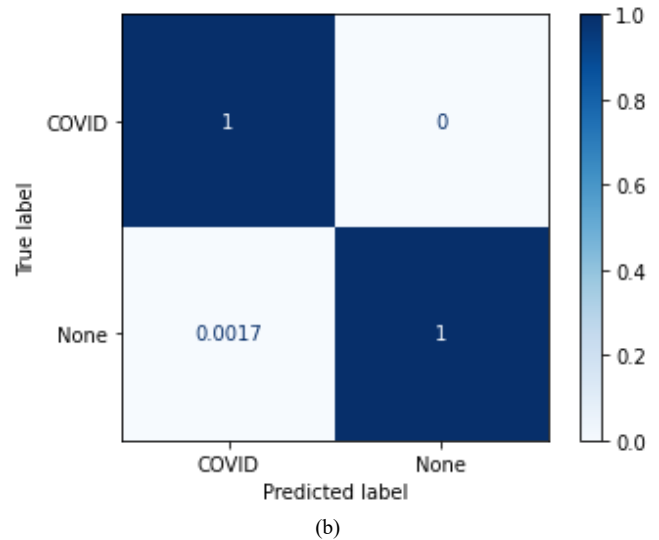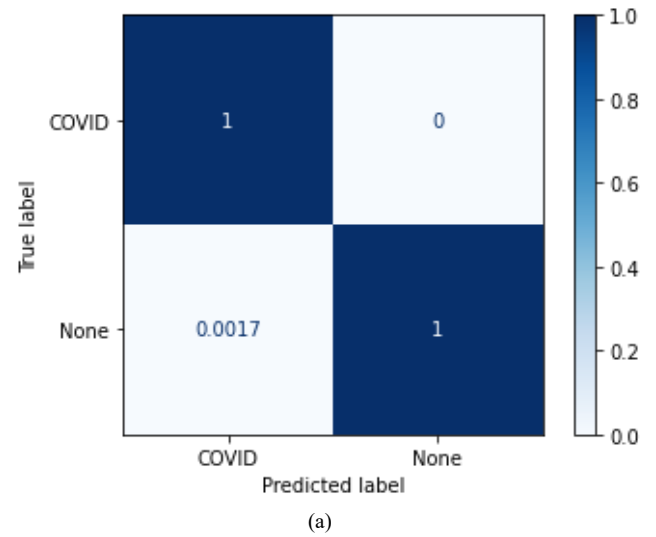
TABLE VI
THE PARAMETERS NUMBER OF DIFFERENT CNN ARCHITECTURES

CNN Architecture	No. of Parameters
VGG-16 [13]	138 million
Alex Net [13]	62.3 million
ResNet50 [13]	25 million
Proposed	6.67 million

As shown in Table VI, the proposed model may be considered a simple CNN architecture with robust performance contrast to the other state-of-art CNN architectures. For the binary classification problem, the normalized confusion matrix of the proposed CNN model, VGG-16, Alex Net, and ResNet50 is presented in Figures 11a, b, c, and d, respectively.

As depicted in Figures 11a, and b, the classification accuracy of the proposed model and the VGG-16 model are identical (almost 100% accurate), with 0.17% of misclassification in the Non-Covid-19 class. In Figure 11c, the Alex Net model achieved almost 100% classification accuracy with 0.33% misclassification in the Non-Covid-19 class. Finally, ResNet50 achieved 95% classification accuracy with 5% misclassification in the Non-Covid-19 class, shown in Figure 11d. Therefore, When compared to other models, the proposed model performs robustly. However, the proposed model has a considerably lower parameter number than other

models. Tables VII and VIII present the training and testing time for the proposed architecture and the other DL models in the 3-classes and binary classification problems.

(a)

(b)

(c)
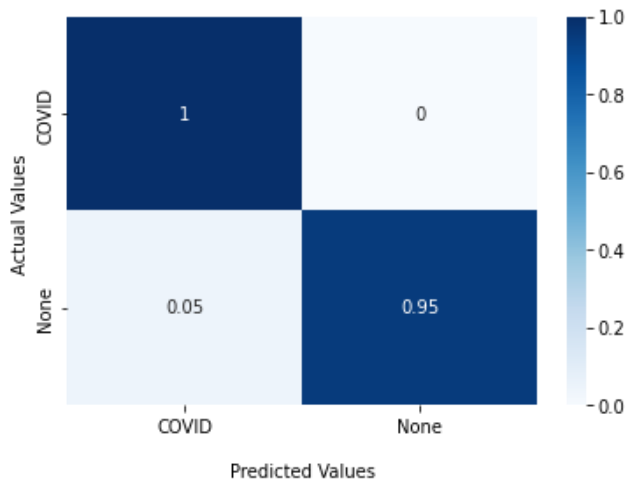
(d)

Fig. 11. The Binary Confusion Matrix of
(a) The proposed model. (b) VGG-16. (C) Alex Net. (D)ResNet50.

TABLE VII
THE TRAINING AND TESTING TIME OF DIFFERENT CNN
ARCHITECTURES FOR THE 3-CLASSES CLASSIFICATION
PROBLEM.

CNN Architecture	Training Time	Testing Time
VGG-16 [13]	12h 0m 34s	324.14s
Alex Net [13]	34m 43s	11.69s
ResNet50 [13]	4h 43m 18s	91.496s
Proposed	30m 41s	10.74s

TABLE VIII
THE TRAINING AND TESTING TIME OF DIFFERENT CNN
ARCHITECTURES FOR THE BINARY CLASSIFICATION PROBLEM.

CNN Architecture	Training Time	Testing Time
VGG-16 [13]	4h 31m 59s	127.98s
Alex Net [13]	25m 31s	5.72s
ResNet50 [13]	2h 31m 8s	84.197s
Proposed	12m 29s	4.9s

As shown in Tables VII and VIII, the proposed CNN architecture has a lower training and testing time than the other architectures. Concretely, the low parameter number in the proposed model plays a vital role in reducing the processing (learning) time (training and testing time). Hence, the proposed model could be a practical choice for clinical practitioners and radiologists to diagnose COVID-19 from CT images. Our proposal is a light- weight CNN model that enables doctors to diagnose patients easily and faster than other traditional techniques during five second processing time. Figure 12a shows a software program (CNN Model For Detecting COVID-19) that was installed on a website or any mobile application to serve a popular medical society. As shown in Figure 12b, the program enables doctors to select among diverse CT scan images either form Kaggle website or other sites and press Upload for preprocessing, training, and testing a selected one. In Figure 12c, the uploaded image was analyzed onto three main probabilities based 3- classes: COVID-19, PNEUOMNIA, and NORMAL.

CONCLUSION

COVID-19 has an enormous impact on people's daily life. Therefore, COVID-19 diagnosis and early detection become significantly important. A lot of Deep Learning solutions have been investigated for this purpose. This paper introduces a new CNN architecture with less complexity and faster training and testing time. The proposed model performance was tested and evaluated on different dataset sizes with 3-class and binary classification problems to ensure the robustness of the model. The proposed model achieved a robust performance compared to the other well-known DL models such as VGG-16, Alex Net, and Res Net 50 in both classification problems. Nevertheless, the proposed CNN architecture consists of significantly fewer parameters than the other models. Hence, the training and testing time of the proposed model is apparently less than the other models. The proposed model can be very useful for radiologists and health experts to develop a better understanding of COVID-19 cases detection and classification. CNN Model For Detecting COVID-19 program was validated in this work to prove the concept of our proposed technique.

REFERENCES

[1] Marco Ciotti, Massimo Ciccozzi, Alessandro Terrinoni, Wen-Can Jiang, Cheng-Bin Wang & Sergio Bernardini (2020) The COVID-19 pandemic, Critical Reviews in Clinical Laboratory Sciences, 57:6, 365-388, https://doi.org/10.1080/10408363.2020.1783198

[2] Skegg, D., Gluckman, P., Boulton, G., Hackmann, H., Karim, S. S., Piot, P., & Woopen, C. (2021). Future scenarios for the COVID-19 pandemic. The Lancet, 397(10276), 777–778. https://doi.org/10.1016/s0140-6736(21)00424-4

[3] E. Mathieu et al., "A global database of COVID-19 vaccinations," Nat. Hum. Behav., vol. 5, no. 7, pp. 947–953, Jul. 2021, https://doi.org/10.1038/S41562-021- 01122-8 .

[4] I. Katsamenis, E. Protopapadakis, A. Voulodimos, A. Doulamis, and N. Doulamis, "Transfer Learning for COVID-19 Pneumonia Detection and Classification in Chest X-ray Images," in 24th Pan-Hellenic Conference on Informatics, Nov. 2020, pp. 170–174, https://doi.org/10.1145/3437120.3437300

[5] F. Shi et al., "Review of Artificial Intelligence Techniques in Imaging Data Acquisition, Segmentation, and Diagnosis for COVID-19," IEEE Reviews in Biomedical Engineering, vol. 14. pp. 4–15, 2021, https://doi.org/10.1109/RBME.2020.2987975

[6] Shah, V., Keniya, R., Shridharani, A., Punjabi, M., Shah, J., & Mehendale, N. (2021). Diagnosis of COVID-19 using CT scan images and deep learning techniques. Emergency Radiology, 28(3), 497–505. https://doi.org/10.1007/s10140-020-01886-y

[7] [Book] Mahrishi, M., Hiran, K. K., Meena, G., & Sharma, P. (2020). Machine learning and deep learning in real-time applications. IGI Global.

[8] [Book] Deng, L., & Yu, D. (2014). Deep learning: Methods and applications. now Publishers Inc.

[9] [Book] Gulli, A., & Pal, S. (2017). Deep learning with Keras: Implementing deep learning models and neural networks with the power of Python. Packt Publishing.

[10] T. D. Pham, "Classification of COVID-19 chest X-rays with deep learning: new models or fine-tuning?" Heal. Inf. Sci. Syst., vol. 9, no. 1, p. 2, Dec. 2021, https://doi.org/10.1007/s13755-020-00135-3

[11] A. Narin, C. Kaya, and Z. Pamuk, "Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks," Pattern Anal. Appl., vol. 24, no. 3, pp. 1207–1220, Aug. 2021, https://doi.org/10.1007/s10044-021-00984-y

[12] J. P. Cohen, P. Morrison, L. Dao, K. Roth, T. Q. Duong, and M. Ghassemi,

"COVID-19 Image Data Collection: Prospective Predictions Are the Future," Jun. 2020, [Online]. Available: http://arxiv.org/abs/2006.11988

[13] Jaiswal, A.K.; Tiwari, P.;Rathi,V.K., Qian, J.; Pandey, H.M.;Albuquerque, V.H.C.: Covidpen: a novel COVID-19 detection model using chest X-rays and CT scans. medrxiv (2020).

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

[15] S. Asif, Y. Wenhui, H. Jin, and S. Jinhai, "Classification of COVID-19 from Chest X-ray images using Deep Convolutional Neural Network," in 2020 IEEE 6th International Conference on Computer and Communications (ICCC), Dec. 2020, pp. 426–433, https://doi.org/10.1109/ICCC51575.2020.9344870

[16] I. D. Apostolopoulos, S. I. Aznaouridis, and M. A. Tzani, "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., vol. 40, no. 3, pp. 462–469, Jun. 2020, https://doi.org/10.1007/s40846-020-00529-4

[17] "Kaggle: Your Home for Data Science" https://www.kaggle.com/tawsifurrahman/covid19-radiography-database (accessed Mar. 14, 2022).

[18] Z. Li et al., "From community-acquired pneumonia to COVID-19: a deep learning–based method for quantitative analysis of COVID-19 on thick-section CT scans," Eur. Radiol., vol. 30, no. 12, pp. 6828–6837, Dec. 2020,

[19] Neha Rajawat, Bharat Singh Hada, Mayank Meghawat, and Soniya Lalwani, Rajesh Kumar4, "C-COVIDNet: A CNN Model for COVID-19 Detection Using Image Processing," Arabian Journal for Science and Engineering,30 April 2022.

[20] Alom, Md. Zahangir & Taha, Tarek & Yakopcic, Christopher & Westberg, Stefan & Hasan, Mahmudul & Esesn, Brian & Awwal, Abdul & Asari, Vijayan. (2018). The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches.

[21] Mascarenhas and M. Agarwal, "A comparison between VGG16, VGG19 and ResNet50 architecture frameworks for Image Classification," 2021 International Conference on Disruptive Technologies for Multi- Disciplinary Research and Applications (CENTCON), 2021, pp. 96-99,

[22] Guan, Qing & Wang, Yunjun & Ping, Bo & Li, Duanshu & Du, Jiajun & Yu, Qin & Lu, Hongtao & Wan, Xiaochun & Xiang, Jun. (2019). Deep convolutional neural network VGG-16 model for differential diagnosing of papillary thyroid carcinomas in cytological images: A pilot study. Journal of Cancer.

[23] Api documentation: Tensorflow core v2.5.0. TensorFlow. (n.d.). https://www.tensorflow.org/api_docs

[24] Team, K. (n.d.). Keras documentation: Keras API reference. Keras. https://keras.io/api/

[25] Covidx ct. Kaggle. (n.d.). https://www.kaggle.com/hgunraj/covidxct.

[26] Matplotlib. Overview - Matplotlib 3.4.3 documentation. (n.d.). https://matplotlib.org/stable/contents.html