ERURJ 2024, accpeted

Article

COVID-19 Classification Based Deep Convolutional Neural Network Using CT Scans

Mahmoud Khaled Abd-Ellah [1]*, Heba Hamdy Ali [2], Asmaa Mohamed [1], Amira Mofreh Ibraheem [1]

[1] *Faculty of Artificial Intelligence, Egyptian Russian University, Cairo 11829, Egypt*
[2] *Faculty of computers and Artificial Intelligence, Beni-suef university, Beni-suef, Egypt*
*Corresponding author(s): Mahmoud Khaled Abdellah, E-mail: mahmmoud-khaled@eru.edu.eg

Received 19th July 2023, Revised 24th January 2024, Accepted 8th February 2024

DOI:10.21608/erurj.2024.221776.1059

ABSTRACT

In the year 2020, a pandemic appeared threatening the whole world called COVID-19, the number of deaths due to this dreaded virus is constantly increasing over time. Therefore, many researchers, scientists, and professionals are seeking a solution to this problem. Diagnosis and confirmation of the presence of the virus in a person is done through CT scans. Because of the increased number of infected people, which is estimated in millions, there must be a computer system to help doctors with diagnosis to save time and effort and to help patients in the speed of treatment to preserve their lives and reduce the number of deaths. Thus, we suggested a smart computer method to automatically detect Coronavirus. A modified convolutional neural network (CNN) has been developed for automatic COVID-19 detection. The proposed technique contains three phases. In the first phase, the images are resized, data augmented, over-sampled, and normalized. In the second phase, the CNN extracts features from the images. In the classification phase, the features are used to classify the images as either COVID-19 or non-COVID. The method was evaluated on a database of 1230 non-COVID CT images and 1252 COVID CT images. The method achieved an accuracy of 93.81%, which is outperforming the other methods in terms of accuracy, sensitivity, and specificity.

Keywords: COVID-19, CNN, CT scans, Image Classification

1-Introduction

COVID-19 is the virus detected from coronaviruses. The infection virus is transmitted between people through the small droplets secreted by the person infected with COVID-19. This virus began spreading in December 2019 until this epidemic spread to the whole world. Among the most important symptoms that appear in the affected person are respiratory diseases, the severity of which varies from person to person, and leads to difficulty breathing [1]. So doctors recommend the patient to perform an RT-PCR test and if positive or negative results are obtained from this analysis, the patient performs CT scans for the lung as a result of the low sensitivity of RT-PCR at 60% -70% [2, 3].

We do not deny the wonderful effort by doctors to overcome this epidemic, but with the increase in numbers in huge numbers in a short time, it is necessary to enter modern technological means to help doctors in the early diagnosis of the virus. CNN can be assistive to obtain an accurate diagnosis. Recently, many researchers in various fields have sought to find quick, cheap, accurate, and safe solutions to eradicate this epidemic as shown in **Table 1**. In [1], an nCOVnet method has been proposed to detect the COVID- 19. The nCOVnet network consists of 24 layers which use an input layer and 18 layers of VGG16 network as feature extraction model. Then, a transfer learning model consisting of 5 different layers has been applied. The highest accuracy is 88%.

Tulin Ozturk et al. [2] used the DarkNet model for COVID-19 detection which consists of 24 layers with 5 Maxpool and 19 convolutional. Their model achieved a classification accuracy of 87.02%. Turker Tuncer et al. COVID-19 detection technique has four stages. In the preprocessing stage, they resized and converted the input X-ray images to grayscale. In the feature extraction stage, they used the ResExLBP method to extract features. In the feature selection stage, they used a method called iterative ReliefF (IRF) to select the most discriminative features. In the classification stage, they used five different classifiers [4]. M. Toġaçar et al. [5] introduce a COVID-19 detection technique using the pre-trained model MobileNetV2, SqueezeNet in the feature extraction stage, and the Social Mimic optimization method in the feature selection stage and SVM in the classification stage.

A classification technique is applied to COVID-19 identification through an X-ray dataset by R.M. Pereira et al. using a pre-trained CNN to extract features provided to multi-class classification algorithms. The presented method achieved a 0.89 F1-Score for the hierarchical COVID-19 classification scenario[6]. The CNN transfer learning was adopted for COVID-19 detection by Ioannis and Tzani. The best specificity, sensitivity, and accuracy obtained are 96.46%, 98.66%, and 96.78% respectively [7]. ABDUL WAHEED et al. [8] present a technique called CovidGAN that uses ACGAN to generate synthetic X-ray images which enhance the CNN performance an accuracy of 85%.

Ferhat and Deniz [9] demonstrate Deep Bayes-SqueezeNet for the COVID-19 diagnosis. The augmented dataset and Fine-tuned hyperparameters increase the diagnosis accuracy of the network. A patch-based CNN with a small trainable parameter number has been proposed by Yujin Oh et al. The method achieved maximum accuracy of 88.9% for COVID-19 diagnosis with Patch size 224×224 With mask [10]. Mohammad and Abolfazl [11] proposed A modified CNN based on the concatenation of ResNet50V2 and Xception to detect COVID-19 with an average accuracy of 91.4%. Table 1 provides a comparison between all the related work concerned on the used dataset, the used techniques and the performance of each work is also mentioned.

Many CNN models have been developed to efficiently classify COVID-19 from CT images. These models use a transfer learning approach, which means that it is pre-trained on a large dataset of images from other sources. This allows the model to learn the basic features of images, which can then be fine-tuned to classify COVID-19 images. The proposed model introduces a new design of CNN which consists of three main stages. In the preprocessing stage, the images are resized, augmented, over-sampled, and normalized. The feature extraction stage uses a new design of CNN to extract features with three parallel paths. The classification stage then uses these features to classify the features into two classes: COVID-19 positive or negative. The model was evaluated on a dataset of CT images from COVID-19 patients. The model achieved a high accuracy of 93.81%.

The paper is organized as follows: Section 2 provides the details of the designed model. Section 3 describes the testing environment, database, and evaluation metrics. Section 4 provides the evaluation results and the discussion. Section 5 provides the conclusion.

Table 1. SUMMARY OF COVID-19 DETECTION ALGORITHMS.

Method	Dataset		Technique Type	Accuracy
	Total	COVID-19		
Harsh Panwar et al. [1]	284	142	nCOVnet	88.09
Tulin Ozturk et al. [2]	1428	224	DarkNet	87.02
Turker Tuncer et al.[4]	321	87	ResExLBP, IRF, and 5 classifier	92.8
M. Toğaçar et al.[5]	458	295	MobileNetV2, SqueezeNet, and SVM	84.56
Ioannis and Tzani [7]	1427	224	VGG19, MobileNet v2, Inception, Xception	86.13
WAHEED et al. [8]	1124	403	CovidGAN, ACGAN, and CNN	85
Ferhat and Deniz [9]	5949	76	Deep Bayes-SqueezeNet	76.37

Yujin Oh et al. [12]	15043	180	The patch-based CNN	88.9
Mohammad and Abolfazl [11]	15085	180	ResNet50V2 and Xception	91.40

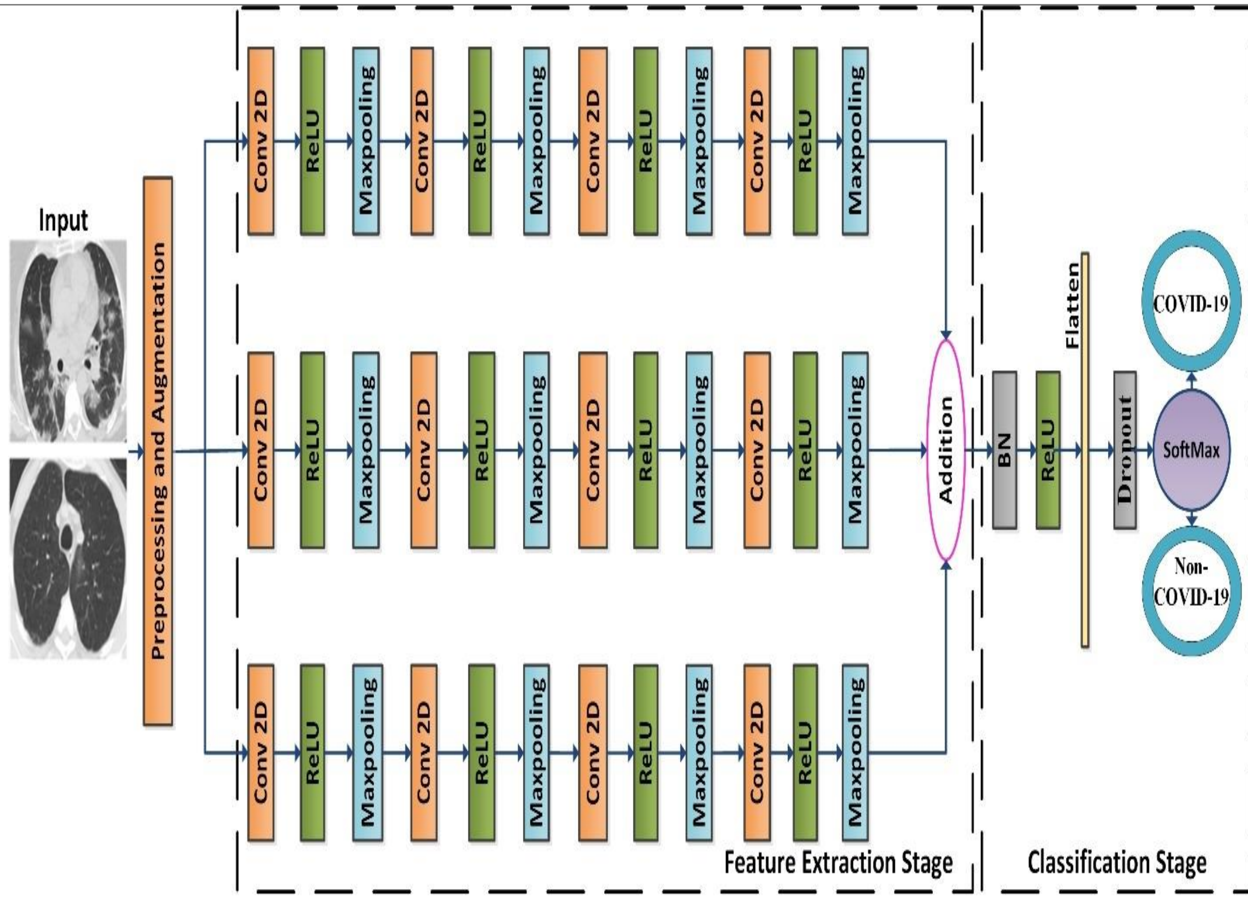

Figure 1. The overall structure of the proposed network.

2. The Proposed Model

Because of their ability to learn hierarchical representations, exploit spatial invariance, use parameter sharing and convolutional filters, learn complex features, facilitate transfer learning, and deliver state-of-the-art performance, Convolutional Neural Networks (CNN) play an important role in image classification [13]. In our suggested models, we are concerned with different affected layers of CNN that can aid in the extraction of more complicated characteristics. First, we will detail the steps of our work that begin with preprocessing and augmentation, then feature extraction, and ultimately classification, as shown in Fig.1.

Algorithm 1 depicts the suggested algorithm's structure. Specifically, by providing a collection of preset CNN building blocks and the image classification dataset, the proposed method begins to operate and eventually presents the optimum CNN architecture to classify the supplied image dataset through a series of evolutionary processes. First, an individual CNN is decoded, and then a classifier is added to this CNN based on the provided image classification dataset. The suggested technique employs a softmax classifier, and the number of classes is specified by the picture dataset. When decoding a CNN, a rectifier activation function is applied to the output of the convolutional layer, followed by maxpooling. A batch normalisation operation, a rectifier activation function, flatten, and dropout are applied before the softmax layer. The CNN is then trained on the training data using the GPU provided, and the classification accuracy is determined using the fitness assessment data. It should be noted that the employment of the softmax classifier and SGD training approach is based on deep learning community practices. When the training phase is completed, the individual's fitness is determined by the best classification accuracy on the fitness evaluation data.

Algorithm 1: Pseudo-code for the Proposed Algorithm

Input: Set Of the pre-defined building block Of CNN, The individual $individual$, the available GPU, the number of training epochs, Maximum iteration, the training data D_{train} and the fitness evaluation data $D_{fitness}$ from the given image classification dataset, and CNN architecture.

Output: best accuracy of CNN architecture to classify the COVID-19 CT image dataset.

Initialization: Construct a CNN with a classifier based on the information encoded in $individual$ and the given image classification dataset; Initialize bounds of hyper-parameter Of CNN i.e filter size, strids, and padding.

$v_{best} \Leftarrow 0$;
foreach epoch in the given training epochs do
    Train the CNN on D_{train} by using the given GPU;
    $v \Leftarrow$ Calculate the classification accuracy on $D_{fitness}$;
    If $v > v_{best}$ then
        $v_{best} \Leftarrow v$
    End
End
Set v_{best} as the fitness of $individual$;
Return

2.1 Preprocessing and Augmentation

The input images are resized to 256x256x3, which means that they are made to be 256 pixels wide, 256 pixels tall, and have 3 channels (red, green, and blue). The images are then normalized, which means that they are scaled so that their values fall within a certain range. To avoid the unbalanced classes problem and overfitting, data augmentation and over-sampling methods are used. Over-sampling increases the number of images in the minority class. In this case, the new images are created by zooming in or out by a factor of two, rotating the images by 90°, and flipping the images horizontally and vertically.

2.2 Feature extraction stage

The feature extraction stage contains three parallel paths. Each path contains four convolutional, ReLU layers. A max-pooling layer is used between the convolutional layers to reduce the dimensionality of the features. Different filter size is used in the first convolutional layer in each path. This allows the model to extract different feature levels. The three paths are then merged using an addition layer. The final feature vector is then used for classification as shown in the figure. 1.

Convolutional Layers: The convolutional layer is the main layer in a convolutional neural network (CNN). It is used to extract features from the input data by applying a set of weights (filters) to the local regions of the input. The filter sizes can be different to extract different feature levels. The stride is a hyper-parameter that controls the size of the output. In this case, the stride is set to 1, which means that the filter moves one pixel at a time. The convolutional layers used in this model have 64 filters. This means that 64 different filters are applied to the input data. The ReLU is applied to the convolutional layers output. This function helps to improve the performance of the CNN by introducing non-linearity into the model. The mathematical expression of 2-D convolutional procedure is explained as follows [14]:

$$Y[i,j] = \sum_{f=-\infty}^{\infty} \sum_{g=-\infty}^{\infty} N[f,g]\, X[i-f, j-g] \tag{1}$$

where X is the matrix of input image, N is the filter matrix to generate the output image matrix Y. $[i,j]$ represent image matrices, whereas $[f,g]$ represent kernel matrices. To retain the size of the output extracted feature maps, zeros padding (P) is sometimes used. An input CT image with width W_{in}, height H_{in}, and number of channels C, and an output image with width K_{op} and height Z_{op} with filter's size ($m\ x\ m$) can be represented as

$$K_{op} = \left[\frac{W_{in} - m + 2P}{s}\right] + 1 \tag{2}$$

$$Z_{op} = \left[\frac{H_{in} - m + 2P}{s}\right] + 1 \tag{3}$$

Rectified linear units (ReLUs): Nonlinearities are important in neural networks because they allow the network to learn complex relationships. Without nonlinearities, the network would only be able to learn linear relationships, which are much less powerful. This is why nonlinearities are essential in neural networks, which makes the network more powerful [15]. The ReLU is the most commonly used in CNNs which is represented in equation (1). It has several advantages, including fast learning, better performance, solving the vanishing gradient problem, and ReLUs are very efficient to compute, which can make training CNNs faster [16].

$$ReLU\ (Y) \ = \ max\ (0, Y) \tag{4}$$

Pooling layer: The max-pooling is a down-sampling layer that reduces the feature map size by taking the maximum value in each subregion of the feature map [17]. This has several benefits, including:

- Noise reduction: Max-pooling can help to reduce noise in the feature map by only considering the most important features.
- Receptive field amplification: Max-pooling can amplify the receptive field of the network by taking the maximum value in a larger subregion of the feature map. This can help the network to learn more complex features.
- Dimension reduction: Max-pooling can reduce the dimensionality of the feature map, which can make the network more efficient to train and deploy.

The network uses max-pooling with 3x3 receptive fields and a stride of 2. This means that the max-pooling layer will take the maximum value in each 3x3 subregion of the feature map, and it will move 2 pixels at a time. Eq. (5) describes the maxpooling equation in which Y is the output of pooling operator linked the kth feature map, and X is the element at (p, q) of Rij which is the pooling region that denotes adjacent area around the position (i, j) [18].

$$Y = max_{(p,q)\in Rij}\, X \tag{5}$$

Addition layer: The addition layer is simple but effective. It is a versatile layer that can be used in a variety of applications. The addition layer adds the three parallel path features with the same shape.

2.3 Classification stage

The classification stage of the model classifies the merged features into two classes for four magnification factors. It contains a batch normalization (BN) layer, a ReLU layer, a flattened layer, a dropout layer, and a dense layer with a softmax activation function as the classification layer as shown in the figure. 1. The BN layer normalizes the input features, which can help to improve the stability of the network. The ReLU layer applies the rectified linear unit activation function, which can help to improve the performance of the network. The flattened layer flattens the input features, which can help to improve the efficiency of the network. The dropout layer randomly drops out some of the input features, which can help to prevent overfitting. The dense layer with the softmax activation function classifies the input features into two classes. The classification stage is the final stage of the model, and it is responsible for determining whether the input image is COVID-19 positive or negative.

Batch normalization (BN): Batch normalization (BN) is a regularization technique that can improve the generalization of deep neural networks and accelerate convergence. BN works by normalizing the input features to a standard distribution before they are passed to the next layer. This helps to prevent the network from becoming too sensitive to the initial values of the weights, which can lead to overfitting [15].

BN also helps to address the problem of covariate shift, which is a change in the distribution of the input features during training. This can happen, for example, if the training data is not representative of the test data. BN can help to mitigate the effects of covariate shifts by normalizing the input features to a standard distribution. Another benefit of BN is that it can accelerate convergence. This is because BN helps to stabilize the learning process by preventing the weights from becoming too large or too small. This can lead to faster convergence and better performance. For each mini-batch B during training as shown in Eq. (6), the BN parameters μx, σ_x are calculated as the mean and the Standard_deviation of B_x $which$ represents the collection of feature maps Y_x at the input to layer x with weights W_x [19].

$$\mu_x \leftarrow \frac{1}{|B_x|}\sum_{Y_x \in B_x} W_x Y_x \quad , \qquad \sigma_x \leftarrow \sqrt{\frac{1}{|B_x|}\sum_{Y_x \in B_x}(W_x Y_x - \mu_x)^2} \tag{6}$$

Flatten layer: The flatten layer is a neural network layer that converts the input to a flat vector output. This means that the layer takes the input, which may be a multidimensional tensor, and flattens it into a single vector. The size of the batch is not affected by the flatten layer.

Dropout layer: Dropout is a regularization technique that can prevent overfitting by randomly dropping out (or ignoring) neurons during training. This means that the network will not be able to rely on any single neuron, which can help to prevent the network from becoming too sensitive to the specific input data. It is often used in conjunction with other regularization techniques, such as batch normalization and L2 regularization [17].

Softmax: The softmax activation function is a normalization function that is often used in the output layer of CNN. It takes a vector feature as input and outputs a probabilities vector. The probabilities represent the likelihood that the input belongs to each of the possible classes. The classification layer is the final layer, and it returns a neural network classification output [20]. The softmax function is defined as follows:

$$W_j = \frac{\exp(X_j)}{\sum_{i=1}^{n} \exp(a_i)} \tag{7}$$

3-Testing Environment

In this section, we perform some experiments on the COVID-19 database to evaluate the proposed method.

3.1 Dataset

In our work, we used a dataset of 2482 CT scans downloaded from the Kaggle website [21], containing 1252 COVID-19 and 1230 non-COVID-19. As shown in **Figure 2**, we can classify the input dataset into COVID and non-COVID classes. The dataset is divided randomly between training and testing with percentage values of 70% and 30% respectively.

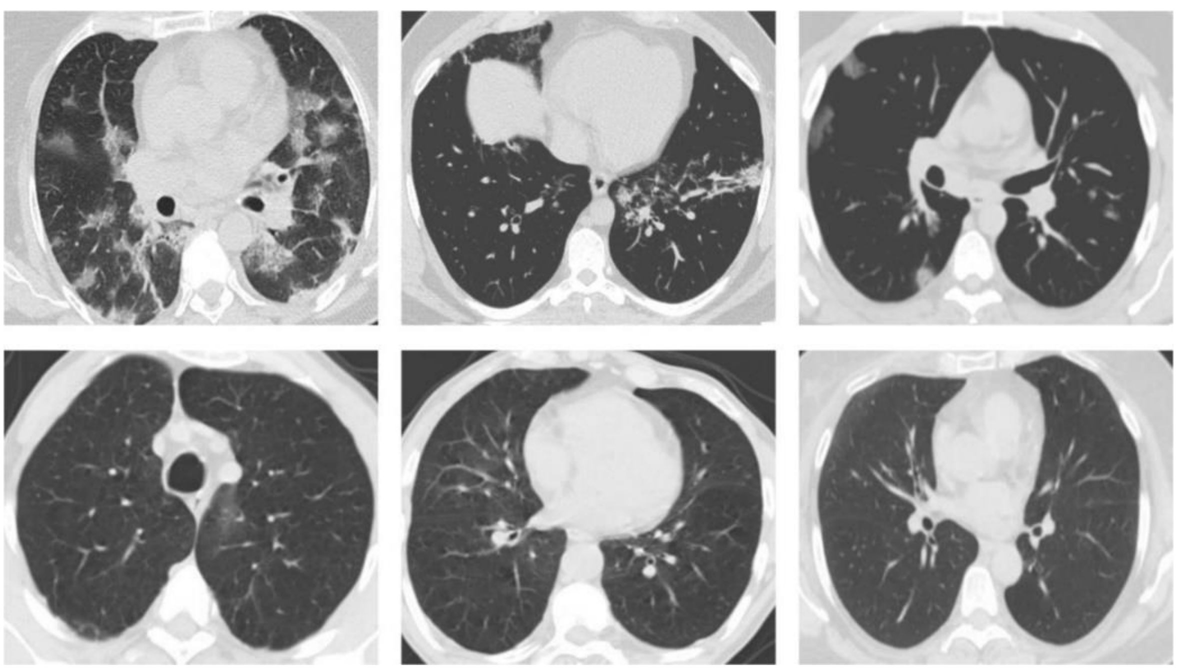

Figure 2. Different CT images with COVID-19 in the first raw and non-COVID-19 in the second raw.

3.2 Evaluation metrics

Reliable benchmarks such as accuracy, Sensitivity, and Specificity are used to assess the performance of the proposed system and assure its efficacy. The outcomes of the trials were averaged across patch levels rather than patient levels [22].

$$Accuracy\ (ACC) = \frac{\alpha + \beta}{\alpha + \beta + \Omega + \mu} \tag{8}$$

Sensitivity (Sen) is considered as the rate of correctly classified positive samples in the positive class dataset,

$$\text{Sensitivity (SEN)} = \frac{\alpha}{\alpha + \mu} \tag{9}$$

Specificity (SPC) [23] is the measure of true negatives that are correctly classified by the model

$$\text{Specificity (SPC)} = \frac{\beta}{\beta + \Omega} \tag{10}$$

where α the True Positive, β the True Negative, Ω the False Positive, and μ the False Negative values.

4- Experimental Results and Discussion

The researchers first tested several different designs of convolutional neural networks (CNNs) to find the one that achieved the best accuracy, sensitivity, and specificity in classifying COVID and non-COVID CT images. The confusion matrices of the proposed method and two other literatures are shown in Table 2 which illustrate that the proposed method achieved very good results on both COVID and non-COVID images. To ensure that the results were credible, the researchers compared their method to other popular methods that have been reported in the literature. The results of this comparison are shown in Table 3. The proposed method outperformed other methods in terms of accuracy, sensitivity, and specificity.

Table 2. The proposed model Confusion matrices

Method		**COVID**	**non-COVID**
Harsh Panwar et al. [1]	COVID	41	1
	non-COVID	9	33
Turker Tuncer et al.[4]	COVID	74	10
	non-COVID	13	224
Proposed Model	COVID	354	25
	non-COVID	21	344

Table 3. Comparison of the proposed detection model against literature.

Ref	Year	Method	Dataset	Evaluation metrics (%)		
				ACC	SEN	SPC
Harsh Panwar et al. [1]	2020	nCOVnet	284	88.09	82	97.06
Tulin Ozturk et al. [2]	2020	DarkNet	1428	87.02	85.35	92.18
Turker Tuncer et al.[4]	2020	SVM	321	92.8	85.1	95.7
M. Toġaçar et al.[5]	2020	SqueezeNet	458	84.56	82.75	85.04
Ioannis and Tzani [7]	2020	Inception	1427	86.13	12.94	99.70
WAHEED et al. [8]	2020	CNN-AD	1124	85	69	95
Ferhat and Deniz [9]	2020	SqueezeNet	5949	76.37	69.21	79.93
Yujin Oh et al. [12]	2020	PATCH-BASED	15043	88.9	85.9	96.4
Mohammad and Abolfazl [11]	2020	Concatenated	15085	91.40	87.35	94.32
J. Skibinska et al. **[24]**	2021	k-NN	54	78	77	80
S. Wang et al. **[25]**	2021	M-inception	1065	89.5	87	88
M. Emin Sahin **[26]**	2022	ResNet50	13,824	91.54	-	-
Saha et al. **[27]**	2022	Parallel CNN	1500	93.44	88.33	95.23
Chowdhury et al. **[28]**	2023	EfficientNetB4	1823	78.63	76.85	-
Khan et al. **[29]**	2023	TL-ShuffleNet	2684	92.26	90.96	93.53
Proposed Model	**2023**	**TPCNN**	**2482**	**93.81**	**94.4**	**93.22**

Time is a critical factor in healthcare, especially in emergencies. Delays in diagnosis and treatment can have serious consequences for patients. For example, delays in COVID-19 test results can lead to delayed treatment and increased spread of the disease. Imaging tests, such as CT scans, can also take a significant amount of time to perform and interpret. This can delay diagnosis and treatment, which can have negative consequences for patients. However, technological advancements have led to the development of faster and more efficient diagnostic and imaging methods. These methods can help to reduce time consumption in healthcare and improve patient outcomes.

In our methodology, we used a model that achieved the best accuracy value among all models. However, this model took 18.35 seconds to train. The average testing time per image was 0.62 seconds. Overall, time consumption is a critical factor in healthcare. Efforts to reduce time consumption can lead to improved patient outcomes and overall healthcare efficiency.

In conclusion, the proposed method is a very effective way to classify COVID-19 and non-COVID-19 CT images. It achieved state-of-the-art results on a variety of datasets, and it is more fast, accurate, sensitive, and specific than any other method that has been reported in the literature. The proposed model outperforming the other models due to the following reasons:

1) Building a new CNN network consists of three parallel paths
2) Selecting the best values of convolutional layer filters that help on extracting more features from different three level.
3) Extract the most relevant features which fit the highest accuracy.
4) Selecting the best hyperparameters of filters, padding, stride.
5) Extract more local and global features.
6) Compare the performance results of the proposed model with literatures for COVID-19 classification and outperformed other methods in terms of accuracy, sensitivity, and specificity as shown in Table 3.

5. Conclusion

In 2020, a pandemic called COVID-19 emerged and caused a large number of deaths. To diagnose the virus, doctors use CT scans. However, with millions of people infected, there is a need for a computer system to help doctors diagnose COVID-19 more quickly and easily. We propose a smart computer-based method for automatically detecting COVID-19. Our method consists of three phases. We use a modified convolutional neural network to classify images. Our method was trained on a dataset of 1230 non-COVID-19 CT images and 1252 COVID-19 CT images. It achieved an accuracy of 93.81%, which is comparable to the accuracy of other state-of-the-art methods. Our method has the potential to help doctors diagnose COVID-19 more quickly and easily. This could help to save lives and reduce the number of deaths from the virus.

- **Conflict of Interest**

The authors declare no conflict of interest.

5. References

[1] H. Panwar, P. K. Gupta, M. K. Siddiqui, R. Morales-Menendez, and V. Singh, "Application of deep learning for fast detection of COVID-19 in X-Rays using nCOVnet," *Chaos, Solitons & Fractals,* vol. 138, p. 109944, 2020/09/01/ 2020.

[2] T. Ozturk, M. Talo, E. A. Yildirim, U. B. Baloglu, O. Yildirim, and U. Rajendra Acharya, "Automated detection of COVID-19 cases using deep neural networks with X-ray images," *Computers in Biology and Medicine,* vol. 121, p. 103792, 2020/06/01/ 2020.

[3] J. P. Kanne, B. P. Little, J. H. Chung, B. M. Elicker, and L. H. Ketai, "Essentials for Radiologists on COVID-19: An Update—Radiology Scientific Expert Panel," *Radiology,* vol. 296, no. 2, pp. E113-E114, 2020/08/01 2020.

[4] T. Tuncer, S. Dogan, and F. Ozyurt, "An automated Residual Exemplar Local Binary Pattern and iterative ReliefF based corona detection method using lung X-ray image," *Chemometrics and Intelligent Laboratory Systems,* vol. 203, p. 104054, 2020/08/15/ 2020.
[5] M. Toğaçar, B. Ergen, and Z. Cömert, "COVID-19 detection using deep learning models to exploit Social Mimic Optimization and structured chest X-ray images using fuzzy color and stacking approaches," *Computers in Biology and Medicine,* vol. 121, p. 103805, 2020/06/01/ 2020.
[6] R. M. Pereira, D. Bertolini, L. O. Teixeira, C. N. Silla, and Y. M. G. Costa, "COVID-19 identification in chest X-ray images on flat and hierarchical classification scenarios," *Computer Methods and Programs in Biomedicine,* vol. 194, p. 105532, 2020/10/01/ 2020.
[7] I. D. Apostolopoulos and T. A. Mpesiana, "Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks," *Physical and Engineering Sciences in Medicine,* vol. 43, no. 2, pp. 635-640, 2020/06/01 2020.
[8] A. Waheed, M. Goyal, D. Gupta, A. Khanna, F. Al-Turjman, and P. R. Pinheiro, "CovidGAN: Data Augmentation Using Auxiliary Classifier GAN for Improved Covid-19 Detection," *IEEE Access,* vol. 8, pp. 91916-91923, 2020.
[9] F. Ucar and D. Korkmaz, "COVIDiagnosis-Net: Deep Bayes-SqueezeNet based diagnosis of the coronavirus disease 2019 (COVID-19) from X-ray images," *Medical Hypotheses,* vol. 140, p. 109761, 2020/07/01/ 2020.
[10] Y. Oh, S. Park, and J. C. Ye, "Deep Learning COVID-19 Features on CXR Using Limited Training Data Sets," *IEEE Transactions on Medical Imaging,* vol. 39, no. 8, pp. 2688-2700, 2020.
[11] M. Rahimzadeh and A. Attar, "A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet50V2," *Informatics in Medicine Unlocked,* vol. 19, p. 100360, 2020/01/01/ 2020.
[12] Y. Oh, S. Park, and J. C. Ye, "Deep Learning COVID-19 Features on CXR using Limited Training Data Sets," *IEEE Transactions on Medical Imaging,* pp. 1-1, 2020.
[13] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging,* vol. 9, no. 4, pp. 611-629, 2018/08/01 2018.
[14] A. Dhillon and G. K. Verma, "Convolutional neural network: a review of models, methodologies and applications to object detection," *Progress in Artificial Intelligence,* vol. 9, no. 2, pp. 85-112, 2020.
[15] M. Abd-Ellah, A. I. Awad, A. A. M. Khalaf, and H. Hamed, "Deep Convolutional Neural Networks: Foundations and Applications in Medical Imaging," 2020, pp. 233-260.
[16] M. K. Abd-Ellah, A. I. Awad, A. A. M. Khalaf, and H. F. A. Hamed, "Two-phase multi-model automatic brain tumour diagnosis system from magnetic resonance images using convolutional neural networks," *EURASIP Journal on Image and Video Processing,* vol. 2018, no. 1, p. 97, 2018/09/30 2018.
[17] M. K. Abd-Ellah, A. I. Awad, H. F. A. Hamed, and A. A. M. Khalaf, "Parallel Deep CNN Structure for Glioma Detection and Classification via Brain MRI Images," in *2019 31st International Conference on Microelectronics (ICM)*, 2019, pp. 304-307.
[18] R. Nirthika, S. Manivannan, A. Ramanan, and R. Wang, "Pooling in convolutional neural networks for medical image analysis: a survey and an empirical study," *Neural Computing and Applications,* vol. 34, no. 7, pp. 5321-5347, 2022/04/01 2022.

[19] R. Balestriero and R. Baraniuk, *Batch Normalization Explained*. 2022.
[20] M. K. Abd-Ellah, A. A. M. Khalaf, A. I. Awad, and H. F. A. Hamed, "TPUAR-Net: Two Parallel U-Net with Asymmetric Residual-Based Deep Convolutional Neural Network for Brain Tumor Segmentation," in *Image Analysis and Recognition*, Cham, 2019, pp. 106-116: Springer International Publishing.
[21] 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*, p. 2020.04.24.20078584, 2020.
[22] A. M. Ibraheem, K. H. Rahouma, and H. F. A. Hamed, "Automatic MRI Breast tumor Detection using Discrete Wavelet Transform and Support Vector Machines," in *2019 Novel Intelligent and Leading Emerging Sciences Conference (NILES)*, 2019, vol. 1, pp. 88-91.
[23] M. Abd-Ellah, A. I. Awad, A. A. M. Khalaf, and H. Hamed, "Two-phase multi-model automatic brain tumour diagnosis system from magnetic resonance images using convolutional neural networks," *EURASIP Journal on Image and Video Processing*, vol. 2018, p. 97, 09/30 2018.
[24] J. Skibinska, R. Burget, A. Channa, N. Popescu, and Y. Koucheryavy, "COVID-19 Diagnosis at Early Stage Based on Smartwatches and Machine Learning Techniques," *IEEE Access*, vol. 9, pp. 119476-119491, 2021.
[25] S. Wang *et al.*, "A deep learning algorithm using CT images to screen for Corona virus disease (COVID-19)," (in eng), *European radiology*, vol. 31, no. 8, pp. 6096-6104, 2021.
[26] M. Emin Sahin, "Deep learning-based approach for detecting COVID-19 in chest X-rays," *Biomedical Signal Processing and Control*, vol. 78, p. 103977, 2022/09/01/ 2022.
[27] P. Saha and S. Neogy, "Concat_CNN: A Model to Detect COVID-19 from Chest X-ray Images with Deep Learning," *SN Computer Science*, vol. 3, no. 4, p. 305, 2022/05/23 2022.
[28] D. Chowdhury *et al.*, "CoviDetector: A transfer learning-based semi supervised approach to detect Covid-19 using CXR images," *BenchCouncil Transactions on Benchmarks, Standards and Evaluations*, vol. 3, no. 2, p. 100119, 2023/06/01/ 2023.
[29] S. H. Khan *et al.*, "COVID-19 detection and analysis from lung CT images using novel channel boosted CNNs," *Expert Systems with Applications*, vol. 229, p. 120477, 2023/11/01/ 2023.