Deep Learning Based Automated COVID-19 Classification from Computed Tomography Images

Journal:	*Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*
Manuscript ID	TCIV-2023-0058
Manuscript Type:	Research Article
Date Submitted by the Author:	09-Feb-2023
Complete List of Authors:	MORANI, KENAN; Izmir Demokrasi Universitesi, Eelctrical and Electronics Engineering Unay, Devrim; Izmir Demokrasi Universitesi, Electrical-Electronics Engineering
Keywords:	Medical Image Classification, Computed Tomography CT, Convolutional Neural Networks CNNs, COVID-19 diagnosis, Macro F1 Score, Computer Aided Diagnosis, Surgery, Therapy and Treatment

Deep Learning Based Automated COVID-19 Classification from Computed Tomography Images

Kenan Morani[a] * and D. Unay[b]

Electrical and Electronics Engineering Department, Izmir Democracy University, Izmir, Turkey

[a]kenan.morani@gmail.com, [a]0000-0002-4383-5732

[b]devrim.unay@idu.edu.tr, [b]0000-0003-3478-7318

Deep Learning Based Automated COVID-19 Classification from Computed Tomography Images

This paper proposes a Convolutional Neural Networks (CNN) based method with image pre-processing and hyperparameters tuning for image classification. The aim is to achieve high performance for COVID-19 diagnosis with a less complex methodology using a rigorously annotated COV19-CT database. The CNN model comprises of four similar convolutional layers followed by a flattening and two dense layers. The model is a light solution based on simply classifying 2D-slices of Computed Tomography (CT) scans. The slices were processed via anatomy-relevant masking, followed by removal of non-representative slices from the CT volume. To achieve that, a fixed-sized rectangular area was used for cropping an anatomy-relevant region-of-interest in the images, and a threshold based on the number of bright pixels in binarized slices was employed to remove non-representative slices from the 3D-CT scans. Using slice processing techniques, the proposed methodology shows improved quantitative results in classifying slices. In addition, class wight balancing and slice flipping as augmentation techniques together with a learning rate scheduler were deployed to make a diagnosis at slices level. To take patient level diagnosis from CT scan images, a majority voting method applied on the slices of each CT scan was proposed. The macro F1 score of the proposed method well-exceeded the baseline approach and the other alternatives on the validation set as well as on a test partition of previously unseen images of the database.

Keywords: Medical Image Classification; Computed Tomography CT; Convolutional Neural Networks CNNs; COVID-19 diagnosis; Macro F1 Score.

1. INTRODUCTION

The COVID-19 virus, or the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2), is believed to have initially originated from the species of bats and transmitted to human beings in December 2019. The virus spread rapidly all around the world, affecting lots of people and claiming lives as reported by (Hassanin et al. 2021). COVID-19-infected individuals have experienced fever on the onset, generalized fatigue, dry coughing, diarrhea among other possible symptoms as in (Mizrahi et al. 2020). Early

detection and isolation are vitally important to successfully handle the COVID-19 pandemic. Studies have shown the importance of lung imaging for that cause (Rosenthal 2020). The traditional method of detecting COVID-19 is a PCR test and not CT scans. In fact, a CT scan involves the use of radiation on the human body, therefore it can be an invasive test. The PCR test, despite being an invasive test of the human body, is not as aggressive as a CT test. However, A CT scan is needed when the infection is so severe that it spreads to the lungs and detection by nose or oral swabs may not be possible. As reported by (Republic of Turkey, Ministry of Health 2020), a CT scan is advised in either or both of the following two cases :

1. History of fever in child patient or body temperature at or above 38 degrees Celsius.
2. Existence of any findings during auscultation

Furthermore, a CT scan may also be conducted depending on the statue of the patient, whose respiratory findings cannot be examined with X-ray image modality, or who deteriorates clinically.

In general, the traditional PCR testing suffers from low sensitivity, consequently an RT-PCR test might have to be repeated several times, particularly when clinical suspicion is high, but the test results are negative. To compensate potential risk of false-negative in initial screening RT-PCR, chest CT should be applied for clinically suspected COVID-19 patients with negative initial RT-PCR (He et al. 2019).

That being mentioned, we can see the importance and necessity of applying CT imaging modality for accurate diagnosis of COVID-19 in certain cases. In our work, we chose to develop a method using CT scan image modalities.

Automated solutions were proposed by (Rahmani et al. 2022) for COVID-19 detection from images of different modalities using various algorithmic methods. The

proposed methods would report classification performance scores in different matrices including accuracy, precision, recall, specificity, and F1 scores as in an example reported in (Islam et al. 2021). However, in case of an unequal number of observations in the classes (unbalanced data), accuracy of the solutions is important, but it might be misleading. If this is the case, then the model can be assessed in terms of its "Precision" and "Recall". If the former is high, then that means the model gives more relevant results than irrelevant ones. On the other hand, if the latter is high then that means the model gives most of the relevant results (whether irrelevant ones are also returned). Therefore, for unbalanced classification problems, the weighted average of the two scores or the macro F1 score can be used to evaluate the classification performance of a model in a more reliable manner as discussed in papers such as in (Waleed Salehi et al. 2020).

In this paper, the macro F1 score was used to compare the performances of different deep learning models and methods validated on the same dataset. The comparison was made at two levels: slice-level and patient-level. Our deep learning model resulted in state-of-the-art macro F1 score at slice level. Results at patient level was obtained by combining the deep learning model with two main processing techniques. The processing techniques are referred to as slice processing and hyperparameter tuning. The final method achieved a sufficiently high macro F1 score at patient level that it well exceeds the baseline score and many other alternatives on the COV19-CT-DB database.

The design of those models/methods is aimed at finding an automated solution for COVID-19 diagnosis via CT-scan images. The proposed classification solution in this paper progresses from a deep learning model consisting of four similar 2D convolutional layers followed by a flattening layer and two dense layers to a method that is then used to make diagnosis predictions at patient's level using different thresholds via class probabilities and voting from the slices.

The main contributions of this work can be listed as follows:

- We propose a less complex methodology to achieve COVID-19 diagnosis from CT images.
- We show that processing CT images with a Region of Interest (ROI) dedicated to the lung region improves diagnostic performance.
- We propose a mechanism to take patient-level diagnosis from slice-level in each CT scan.
- We evaluate the performance of the proposed solution on a relatively large, and rigorously annotated dataset designed solely for COVID-19 diagnosis.

2. RELATED WORK

Recently, deep Transfer learning and Customized deep learning-based decision support systems are proposed for COVID-19 diagnosis using either CT or X-ray modalities (Fang et al. 2022; Wynants et al. 2020; Huang al. 2020). Some of these systems are developed based on pre-trained models with transfer learning, such as in (Panwar et al. 2020; El Asnaoui et al. 2021), while a few others are introduced using customized networks trained from scratch, such as in (Mary Shyni et al. 2022; Fan et al. 2020; Nayak et al. 2021).

One approach by (Fan et al. 2020) proposed a novel COVID-19 lung CT infection segmentation network, named Inf-Net. The work utilized implicit reverse attention and explicit edge-attention aiming at identification of infected regions in CT images. The work also introduced a semi-supervised solution, Semi-Inf-Net, aiming at alleviating shortage of high-quality labelled data. The proposed method was designed to be effective in case of low contrast regions between infections and normal tissues.

Another approach, by (Nayak et al. 2021), used a deep learning based automated method validated using chest X-ray images collected from different sources. Different

pre-trained CNN models were compared, and the impact of several hyperparameters was analysed in this work to eventually obtain the best performing model. In this work, ResNet-34 model outperformed other competitive networks and thus development of effective deep CNN models (using residual connections) proved to give more accurate diagnoses of COVID-19 infection.

Recent work, by (Ali Ahmed et al. 2022), introduced an ensembled deep neural network (IST-CovNet), providing evaluation of different 2D and 3D approaches on two different datasets and discussing the effects of pre-processing, segmentation, and classifier combination steps on the performance of the approach. The final model combined the use of a novel attention mechanism with slice level combination using LSTMs (Long Short-Term Memory) and an extended architecture for 3D data. This approach proven to increase accuracies in both 2D, and 3D models validated on the public dataset "MosMedData", introduced in (Morozov et al. 2020), achieving state-of-the-art performance. Furthermore, the authors introduced a large, collective dataset referred to as "IST-C", which was made public to contribute to the literature. Their approach also proven to give high performance on their introduced dataset.

While there are multitude of studies aiming at COVID-19 diagnosis using different dataset or combinations of those, we focused on a recent, heterogeneous database of CT scan images, called "COV19-CT-DB". Our work was employed on this particular database for the advantages that come from its large size, challenging nature, and its rigorous and accurate annotation process. The reasons and the advantages are further explained in the 'DATASET' section. The database was shared via an international competition about mid of the year 2021 and was used by several international teams for COVID-19 diagnosis. At the time of conducting this research such number of annotated CT scans was not possible to come by to the best of our knowledge. Secondly, we believe

that reaching high performance on the COV19-CT-DB database will confirm the robustness to our methodology for other similar CT scan problems, given the challenging nature of the database.

Using the COV19-CT-DB series of images, introduced in (Kollias et al. 2021), a baseline approach introduced a deep neural network, based on CNN-Recurrent Neural Network (RNN) architecture. The CNN part of the model extracts features from the images while the following RNN part takes the final diagnostic decision as published in (Kollias et al. 2018, 2020, 2021).

Another study by (Anwar 2021), which used the same database (COV19-CT-DB) for validation, introduced a different method. In this study 2D deep CNN models were trained on individual slices of the database. Performances of the following pre-trained models were compared -VGG, ResNet, MobileNet, and DenseNet. Evaluation of the models was reported both at slice level (2D) as well as at patient/volumetric level (3D) using different thresholding values for voting at the patient level for the latter. The best results were achieved using the ResNet14 architecture (referred to as AutoML model) via 2D images.

In another study, conducted by (Tan 2021), a 3D CNN-based network with BERT was used to classify slices of CT scans. Only part of the images from the COV19-CT-DB database was used in this work. Firstly, the training and validation set of images were passed through a lung segmentation process to filter out images of closed lungs and to remove background. Secondly, a resampling method was used to select a set of a fixed number of slices for training and validation. Finally, A 3D CNN-based model was used followed by a second level MLP classifier to capture all the slices' information from 3D-volumetric images. The final model architecture achieved improved accuracy and macro F1 score on the validation set.

Another study by (Hsu et al. 2021) introduced 2D and 3D deep learning models to predict COVID-19 cases. The 2D model, named Deep Wilcoxon signed-rank test (DWCC), adopts non-parametric statistics for deep learning, making the predicted result more stable and explainable, finding a series of slices with the most significant symptoms in a CT scan. On the other hand, the 3D model was based on pixel- and slice-level context mining. The model was termed as Convolutional CT scan Aware Transformer (CCAT) and used to further explore the intrinsic features in temporal and spatial dimensions.

More work on the same dataset involved deploying a hybrid deep learning framework named CTNet which combines a CNN and a transformer network together for the detection of COVID-19. The method proposed by (Liang et al. 2021) deployed a CNN feature extractor module with Squeeze-and-Excitation (SE) attention to extract features from the CT scans, together with a transformer model to model the discriminative features of the 3D CT scans. The CTNet provides an effective and efficient method to perform COVID-19 diagnosis via 3D CT scans with data resampling strategy. The method's macro F1 score exceeded the baseline on the test partition of the COV19-CT-DB database.

Additionally, on the COV19-CT-DB, two experimental methods that customized and combined Deep Neural Network to classify the series of 3D CT-scans chest images were deployed. The proposed methods included experimenting with 2 backbones: DenseNet 121 and ResNet 101. The experiments were separated into 2 tasks, one was for 2 backbones combination of ResNet and DenseNet and the other one was for DenseNet backbones combination. Introduced by (Trinh et al. 2021), the method's macro F1 score on the test partition of COV19-CT-DB exceeded the baseline score as can be seen on the leader board of "ICCV 2021 Workshop: MIA-COV19D, introducing AI-enabled Medical Image Analysis and Covid-19 Diagnosis Competition".

The proposed deep learning approaches in the literature summarized above achieved high macro F1 scores on the COV19-CT-DB database. However, these methodologies can be quite complex, implementing a full pipeline of segmentation, slice removal and then classification, or using transfer learning architectures with borrowed model weights to obtain high performance. For example, the baseline methodology (Kollias et al. 2021, 2020) focuses on implementing a rather complex CNN-RNN architecture. In here, we present a computationally less expensive solution. Despite its simplicity, performance of our methodology is comparable to the state-of-the-art methods on the same dataset and thus provides a noteworthy alternative for COVID-19 diagnosis and detection.

3. METHODOLOGY

The methodology used in this work includes slices processing before inputting images into a CNN model, which is tuned to predict the probability of CT slices being COVID. Then we use a majority voting method in each CT scan to take the final diagnosis decision of COVID-19 existence in patients. Fig. 1 shows a flow diagram of the proposed methodology. Section 3 discusses the model architecture and hyperparameters tuning, slice processing techniques for slice removal as well as taking diagnostic decision from slices level to patient level in details.

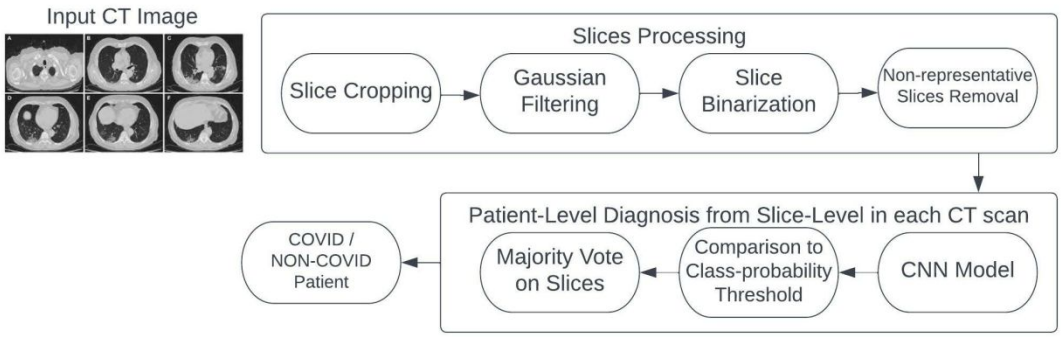

Fig. 1 Block diagram of the proposed approach

3.1 The Model Architecture

The proposed model's architecture consists of four similar sequential 2D convolutional layers followed by a flatten layer, and two dense layers. The input images are designed to have one channel, grayscale.[1]

The number of filters in the convolutional layers are 16, 32, 64, and 128, in order. All filters are 2D, 3x3 in size. Padding was also applied in all four convolutional layers, to match input and output image sizes (Padding="same"). Padding allows for more space for the kernel to cover the image. Adding padding to an image processed by a CNN allows for more accurate analysis of images as explained in (Tang et al. 2019). The four layers had batch normalization and max polling (2,2), and ReLu (Rectified Linear unit) activation function with a binary output for the final diagnosis. Batch Norm is a normalization technique applied between the layers of a Neural Network instead of in the raw data. It is done along mini batches instead of the full data set and serves to speed up training and use higher learning rates, making learning easier as explained in (Chen et al. 2017). Fig. 2 shows the proposed CNN model's architecture.

[1] 512x512 was the size of the original images in COV19-CT-DB database. Cropped images are of size 227x300.

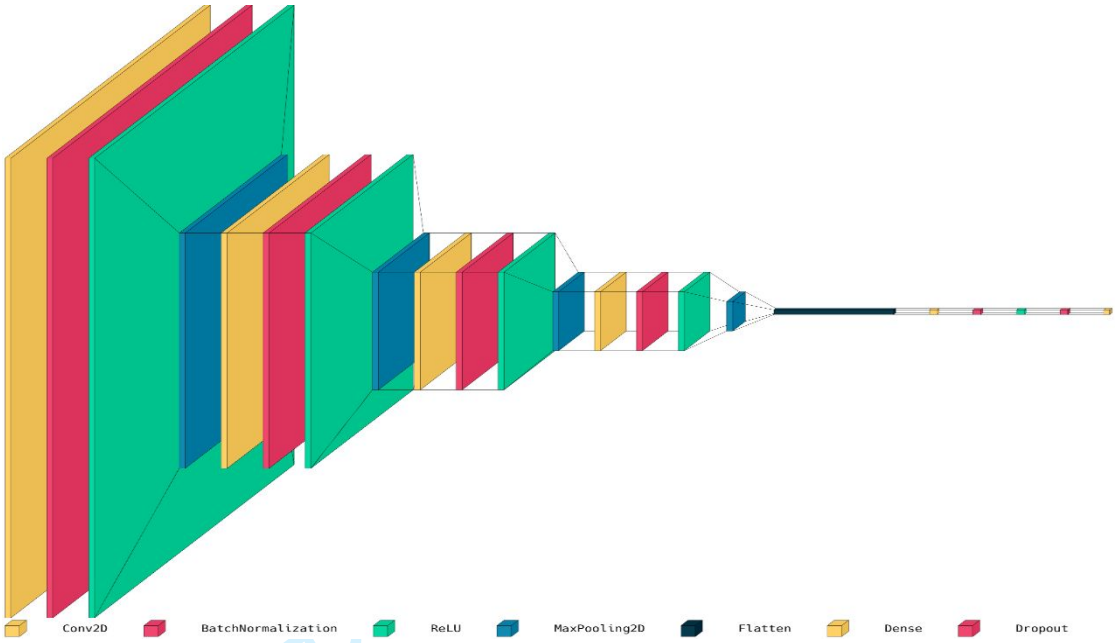

Fig. 2 Visualization of the layers of the proposed CNN model

Following the four convolutional layers was a flattening layer, followed by a dense layer with a dimensionality of 256, batch normalization, ReLu activation function, and a dropout of 0.1. The model then ends with a dense layer using a sigmoid activation function.

The model could be replicated in the following sequence:

- Convolutional layer (512, 512, 16) with padding, Batch Norm, Relu activation
- Max pooling 2D (256, 256, 16),
- Convolutional layer (256, 256, 32) with padding, Batch Norm, Relu activation
- Max pooling 2D (128, 128, 32)
- Convolutional layer (128, 128, 64) with padding, Batch Norm, Relu activation

- Max pooling 2D (64, 64, 64)
- Convolutional layer (64, 64, 128) with padding, Batch Norm, Relu activation
- Max pooling 2D (32, 32, 128)
- Flatten (131072)
- Dense (256)
- Batch Norm (256)
- ReLu activation (256)
- Dropout (256)
- Dense (1)

The output of the final dense layer "Dense (1)" is class1 probability, i.e. the probability of the CNN model predicting class1 corresponding to the Non-COVID class.

The motivation behind the model architecture is adopting symmetric and less complex deep learning network architecture with standard components. For example, using a 3×3 filter was preferred since processing time of 5×5 or larger filters would be almost three times longer or more. Moreover, ReLu activation function, apart from helping with the gradient vanishing problem, is very simple to calculate, as it involves only a comparison between its input and the value '0'. Consequently, its usage helps to prevent the exponential growth in the computation required to operate the neural network compared to non-linear functions such as sigmoid as explained in (Kiliçarslan et al. 2021).

To build an efficient and less hand engineered CNN model, different number of hidden layers were tested against the validation accuracy at slices level. Using three-layer depth model reduced the validation accuracy. Furthermore, using more complex model architecture increased the validation accuracy only trivially. Therefore, consensus was on using a four-layer depth CNN model in our methodology.

3.2 Slice Processing

The activation visualization results of classification on the database show room for improvement in terms of accuracy. Following Grad-Cam visualization in Fig. 9, one can theorize that masking the images with the lung area should improve the performance as the model can better learn to discriminate COVID from Non-COVID. To prove the theory and improve the performance, a fixed-sized rectangular Region of Interest (ROI) was applied to localize the anatomy of interest (lung regions) in the slices. The rectangular area was empirically set to contain both left and right lungs over all slices of every scan in the database. Fig. 3 shows this ROI overlaid on an original slice.

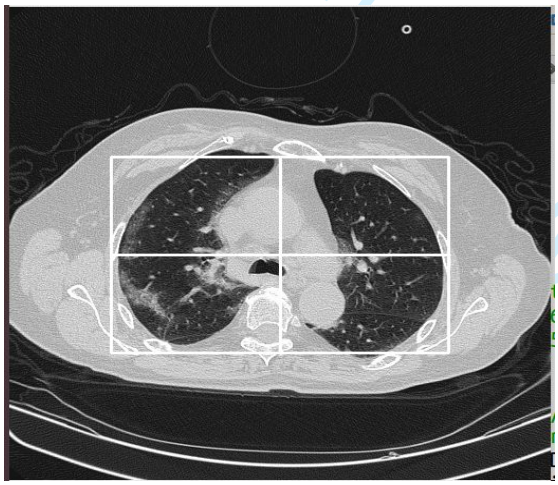

Fig. 3 Static Rectangular Cropping of images

The above-mentioned fixed-sized masking may be affected by the variations in the sizes and shapes of the lung seen at different slices. For example, in the upper slices the lungs appear smaller (relative to the mid-slices), while the lower slices display the lungs

in the shape of a banana. Nevertheless, the fixed-size cropping proposed is an attempt to localize the lung region from the slices in a simple and less complex way. It was conducted by choosing one fully representative slice from a CT scan in the training set and a manually drawn rectangle aimed to capture the lung area. The resulting size of the rectangle is 227×300, which is the size of the input image to the CNN model. In an attempt to account for varying sizes of the lung region in the slices, we work on removing non-representative slices as follows.

After cropping, thresholding was applied to identify and remove uppermost and lowermost slices of the CT scans, corresponding to non-representative slices of the lung volume, aiming to achieve better performance at the patient level diagnosis. Identification of the non-representative slices was realized based on the number of bright pixels in a binarized slice. This procedure is explained below.

Firstly, the cropped images were blurred by using a Gaussian filter to suppress noise and thus enhance large structures in the image. A Gaussian function with a standard deviation of one was convolved with the cropped image's pixel intensity values. The Gaussian function can be expressed in two dimensions as in Equation 1:

$$G(x.y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-x^2-y^2}{2\sigma^2}} \quad (1)$$

Where x and y are the distances from the origin in the horizontal and vertical directions, respectively, and σ is the standard deviation ($\sigma = 1$).

The Gaussian filtering was chosen over other filters because it is less computationally expensive to implement thanks to its filter separability property. The 2D Gaussian filter is separable to two 1D filters and can be expressed as the outer product of the two, which in turn means that the filter can be split into two passes, horizontal and vertical as

explained in (Talbi et al. 2015). With a square image x[k,l] of size N×N and a square filter kernel h[n,m] such as the Laplacian filter of size M×M, the raw 2D convolution to produce the cropped output image of size N×N requires about $N^2 \times M^2$ Multiply–Accumulates (MACs). The raw 2D convolution between x[k,l] and the filter h[n,m] is implemented by using two for loops to range through each output pixel and two additional for loops to perform the 2D convolution at that pixel location. Hence a total of 4 nested for-loops are required resulting in a complexity of $O(N^2 \times M^2)$. With a separable filter h[n,m], such as the Gaussian filter, we can have $h[n,m] = f[n] \times g[m]$, where f[n] and g[m] are the one-dimensional filters. In this case the convolution between the image x[k,l] and the filter h[n,m] can be performed without a raw 2D convolution sum, by the following approach:

- First, perform a 1D convolution between columns of x[k,:] and the 1D filter f[n], which requires about N×M MACs to complete. This operation should be performed for each column of x[k,l] by proceeding along its horizontal, N many, columns. Hence a total of N×M×N MACs will be required to complete the first step to produce the intermediate image w[o,p].
- Then, apply the similar algorithm, with the filter g[m] and rows of the intermediate image w[:,p], which will require similar number of MACs as N×M×N.

Hence in total, only about $2 \times N^2 \times M$ MACs will be needed for the separable implementation of the 2D convolution. The actual number depends on the cropping type applied. Thus, a complexity of only about $O(N^2 \times M)$ is attained. In conclusion, for the implementation of the separable convolution algorithm, only 3 nested for loops are required. This filtering method helps to keep the approach drastically less time and memory consuming when training or testing our method.

Secondly, a histogram based binarization was applied to the resulting blurred images. By looking at the slice's histograms, an estimated threshold for histogram-based image binarization was empirically chosen to be 0.45. This fixed threshold was chosen after applying scale ([0,1]) normalization to the voxel intensities of each scan. Fig. 3 illustrates an exemplary histogram of one of the Gaussian blurred images in the database and the corresponding resulting binarized image.

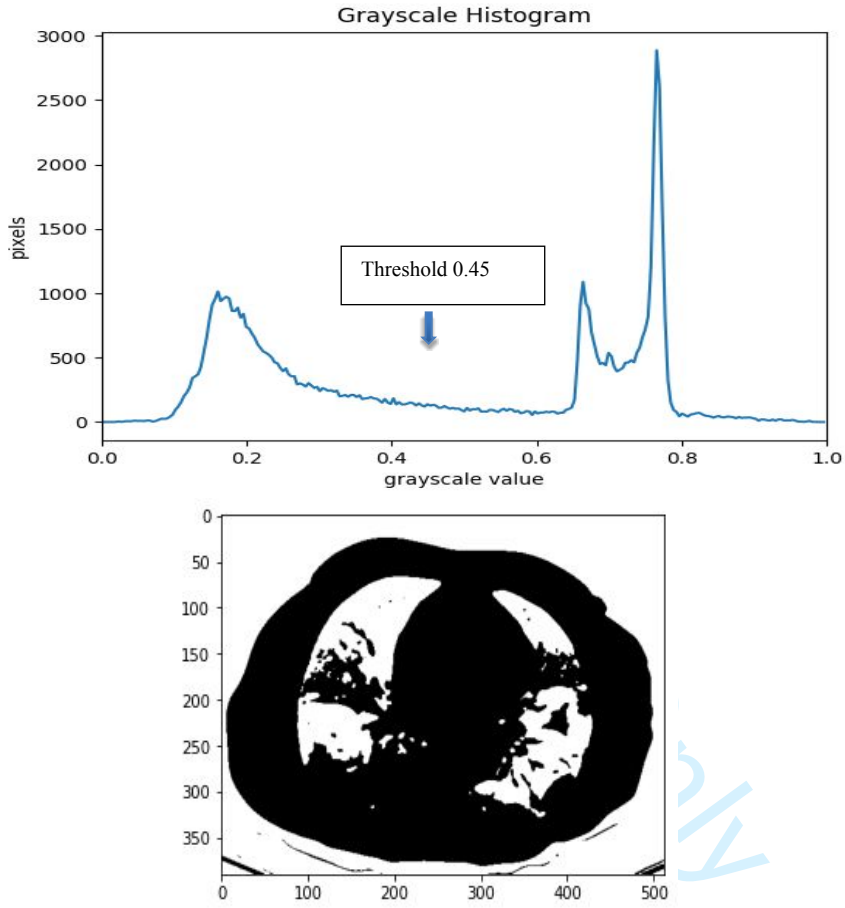

Fig. 4 Histogram of cropped and blurred image (top) and the resulting binary image (bottom)

Finally, the binarized image's pixels were used to find a threshold to remove non-representative slices of the CT volume. To choose the threshold, four candidate CT scans were arbitrarily selected from the training set (CT scans 5, 6, 7, and 8) and random slices from them were processed as explained above. To indicate the importance of the slices in the CT volume, labels from one to three were visually assigned by careful examination;

three being the most representative slice and one being the least representative; a representative slice means a slice that shows a large area of the lung. Similarly, less representative slices are those that display little to no lung area. Fig. 5 shows the results of four candidate CT scan slices.

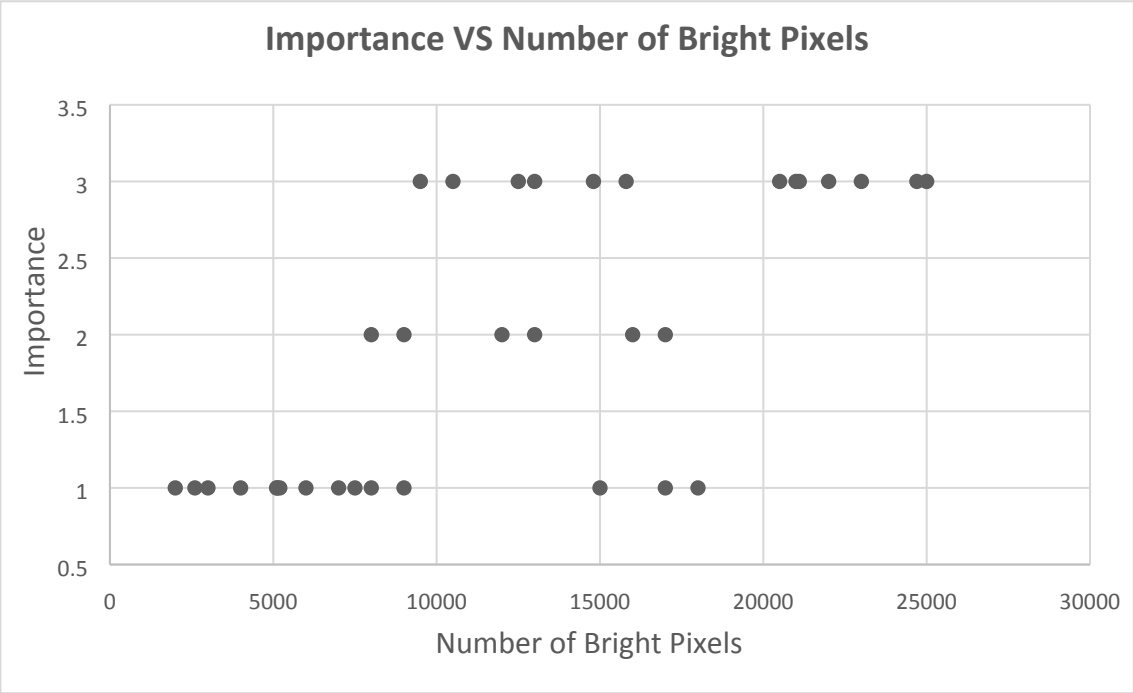

Fig. 5 Number of bright pixels in the candidate CT volume's slices against the importance of the slice in the CT volume

The chosen filtering threshold ratio for the number of bright pixels was 0.066 (corresponding to 4500 out of a total of 68100 pixels in a 227×300 sized slice). Consequently, if the resulting binarized image has a bigger number of bright pixels than the threshold, this indicates that the slice corresponding to the binarized image will be kept in the CT scan, otherwise it will be removed. The threshold was chosen carefully so that at least one slice in each CT scan volume is not removed which will allow to take the final diagnosis decision for all CT scans or all patients. Table 1 shows the error percentage, where errors indicate CT scan images left fully empty (zero slices) after our selected filtering threshold and slightly higher one on all three partitions. The error is then calculated by dividing the number of CT scans left empty after slice removal divided by

all the CT scan in the partition. The results indicate that a filtering threshold in the range of [3400,3500] is optimum for the given data partitions to capture all the CT scans.

Table 1 Percentage errors of different filtering thresholds on all dataset partitions (no error is observed on the Validation COVID [165 CTs], Validation Non-Covid [209 CTs], or Training Covid [690 CTs] sets.)

Data Partition /Threshold	3400 (Selected Threshold)	3500
Training Non-Covid [870 CTs]	0% (No Empty CT)	0.090% (1 Empty CT)
Test [3455 CTs]	0% (No Empty CT)	0.095% (5 Empty CTs)

The slice processing methodology reduces the number of slices in the dataset by including only the representative slices. Accordingly, the number of training and validation slices were reduced to 280462 (corresponding to 16% reduction) and 63559 (15.3% reduction), respectively. Please note that the original number of the slices are as shown in the 'DATASET' section.

The motivation behind slice processing is to localize the lung region of the images in a simple and efficient way. It also aims at removing non-representative slices from each CT scan in a way that will keep at least one slice for each scan image to take the final diagnosis for each patient in all sets od the COV19-CT-DB.

3.3 Hyperparameters Tuning

Hyperparameters were tuned while training the CNN model by changing learning rate, class wight balance, and using augmentation techniques.

Adam optimizer was used with initial learning rate of 0.1. The learning rate decreased exponentially via a learning rate scheduler. The decay of the learning is calculated as in Equation 2:

$$Learning\ Rate = Initial\ Learnig\ Rate \times Decay\ Rate^{\frac{Optimizer\ step}{Decay\ step}} \quad (2)$$

The decay rate was set to 0.96. The value of steps divided by decay steps is an integer division, i.e. the decayed learning rate follows a staircase function.

The optimizer's steps were defined using floor divisions as in Equation 3:

$$train_{step} = \left\lfloor \frac{number\ of\ training\ slices}{batch\ size} \right\rfloor$$

$$validation_{step} = \left\lfloor \frac{number\ of\ validation\ \ slices}{batch\ size} \right\rfloor \quad (3)$$

Decay steps were set every 100000 steps.

Furthermore, class weights were used to balance out varying numbers of input images in the classes. Number of training samples after using non-representative slices removal is reported in Section 3.2. The class weight is calculated using the formula in Equation 4:

$$class\ weight\ = \frac{number\ of\ slices\ for\ a\ class\ in\ the\ training\ set}{number\ of\ all\ slices\ in\ the\ training\ set} \quad (4)$$

Finally, horizontal and vertical flipping were used as augmentation techniques. These image flipping techniques aimed at improving the accuracy via smoothing the effects of the content variations present in the slice as explained in (Hussain et al. 2017).

3.4 Patient Level Decision

At the patient level, different class probability thresholds were tried and compared using class prediction probability to achieve the highest diagnosis accuracy. The class probability thresholds were based on the probability of prediction of class 1 (Non-COVID); If the output probability for class 1 is greater than the chosen threshold, then the slice would be predicted as Non-COVID. Otherwise, the slice would be predicted as

COVID. In that, if number of COVID slices is equal to the number of Non-COVID slices in any one of the CT volume, then the decision is that the patient is a Non-COVID. This slice level decision can be expressed as follows:

if Class1 probability > class probability threshold:

Predict slice as Non-COVID

else:

Predict slice as COVID

After slice level predictions are obtained, a patient is diagnosed based on the presence/absence of COVID slices in his/her CT: if patient CT data contains more Non-COVID predicted slices than COVID predicted slices, the patient is diagnosed as Non-COVID else the patient is diagnosed as COVID (majority voting method).

The clinical relevance of the patient level diagnosis approach we presented above can be explained as follows. Assuming that a patient has lung damage due to COVID seen in 30% of its slices. So, our network classifies around 30% of the slices as COVID and the rest as Non-COVID, and thus the final result will be Non-COVID (in line with majority voting).

While even a minor anomaly seen in a single slice may be attributed to a disease, we speculate that in the Covid case a reasonable amount of involvement is necessary for the diagnostic decision to be taken, and our deep learning model is highly sensitive to even the smallest anomalies observed in the slices.

Please note that we also tried the “all-or-nothing approach” where COVID diagnosis decision is taken even a single slice is predicted as COVID, but that approach yielded less accurate results – as elaborated in the results section – supporting our above observations.”.

3.5 Performance Evaluation

At slices level, validation accuracy was used to evaluate the method's performance. The accuracy is calculated as in Equation 5:

$$Accuracy = \frac{number\ of\ correct\ predictions}{total\ number\ of\ predictions} \quad (5)$$

Over the training, the maximum validation accuracy reached was used as a model performance measurement.

Furthermore, to report the confidence intervals of the results obtained, the Binomial proportion confidence intervals for macro F1 score are used. The confidence intervals were calculated as discussed in (May 28, 2018 posting Jason Brownlee in statistics on machinelearningmastery; unreferenced, see "confidence-intervals-for-machine-learning"), where the radius of the interval is defined as in Equation 6:

$$radius\ of\ interval = z \times \sqrt{\frac{accuracy \times (1 - accuracy)}{n}} \quad (6)$$

n is the number of samples used.

In the above formulation, z is the number of standard deviations from the Gaussian distribution, which is taken as z=1.96 for a significance level of 95%.

At patient level, the proposed model was evaluated via the COV19-CT-DB database. As explained in (Takahashi 2022; Opitz and Burst 2019), the macro F1 score was calculated after averaging precision and recall matrices (the arithmetic mean) at patient level as in as in Equation 7:

$$macro\ F1 = \frac{2 \times average\ precision \times average\ recall}{average\ precision + average\ recall} \quad (7)$$

Performance evaluation of our method is conducted at slice-level and at patient-level, where the former corresponds to considering 2D slices individually in any quantitative or

qualitative analysis. Whereas, in patient level results CT volumes are considered as a whole, and thus the prediction is emphasizing 3D-CT prediction value or patient level rather than each 2D slice's predicted value.

4. DATASET

COV19-CT-DB is the dataset used for validating the methodology proposed in this paper. The CT images in the database were manually annotated by experts and distributed for academic research purposes via the "AI-enabled Medical Image Analysis Workshop and Covid-19 Diagnosis Competition".

The database consists of about 5000 3D chest CT scans acquired from more than 1000 patients. The training set contains 1560 scans in total with 690 of the cases - COVID while the rest (870) belong to the Non-COVID class. The validation set contains, in total, 374, where 165 are COVID cases and 209 are Non-COVID cases. Fig. 6 shows distribution of the CT images with respect to the classes in the training and validation sets. The CT scans in the database contain largely varying slice numbers, ranging from 50 to 700. Please note that the COV19-CT-DB database includes 3 different sets/partitions: a training set, a validation set, and a test set.

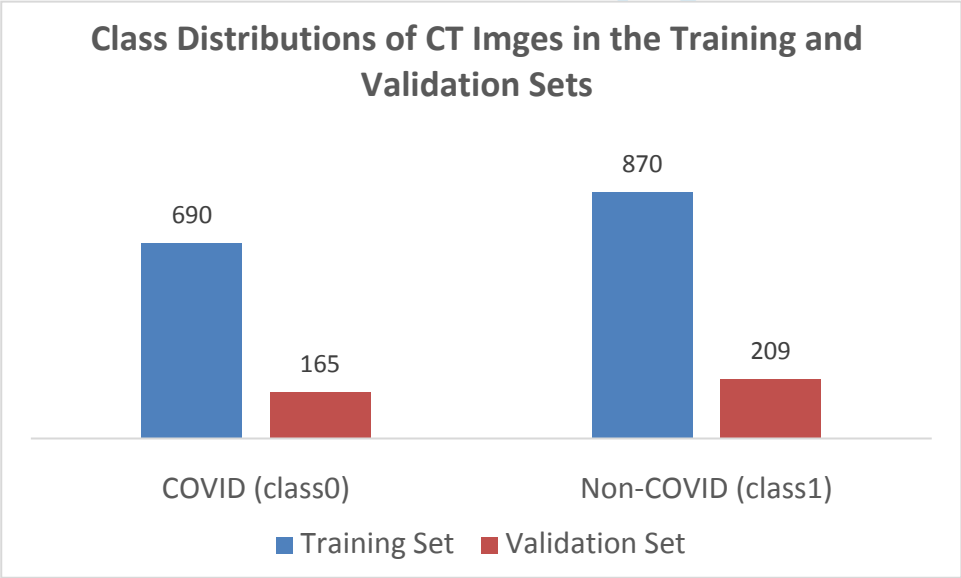

Fig. 6 Class distributions of the CT images in the dataset

The data is unbalanced in terms of the number of 2D slices for both COVID and Non-COVID classes. The images, which are input to the model, were received mainly in loosely compression format; Joint Photographic Experts Group (JPEG) format, grayscale images, with 8-bit depth. The images were all resized to an original size of 512x512 and processed as such.

The dataset used in our paper is preferred for its rigorous and accurate annotation process and its large size. The COV19-CT-DB database was annotated with the help of two radiologists and two pulmonologists with more than 20 years of medical experience in the field. Annotation of the dataset had been realized via the results of a PCR test and a consensus agreement among all medical experts involved, where the agreement reached 98%.Furthermore, COV19-CT-DB is a large dataset, and we believe its use will ensure robustness of our methodology and will make it generalizable on other smaller datasets.

The dataset was also preferred for its challenging nature. In the provided COV19-CT-DB, the slice thickness, the number of slices, and the observed anatomy in the 3D field-of-view in each CT-scan varies – as elaborated in the dataset section. This variability further increases the challenging nature of the problem and potentially make an accurate predictive method pervasive and robust. The model reaching high performance on the COV19-CT-DB is expected to perform well on more consistent datasets.

5. RESULTS

5.1 Slice Level Results

Training the CNN model using a batch size of 128 took about two and half days over a workstation using GNU/Linux operating system on 62GiB System memory with Intel XI(R) W-2223 CPU @ 3.60GHz processor.

Our method achieved an accuracy of 84.5% on the validation set at the slices level. Fig. 7 shows the evolution of validation and training accuracies when using slice processing techniques. Early stopping was used during training to halt the training if the validation accuracy does not show improvement after 7 epochs.

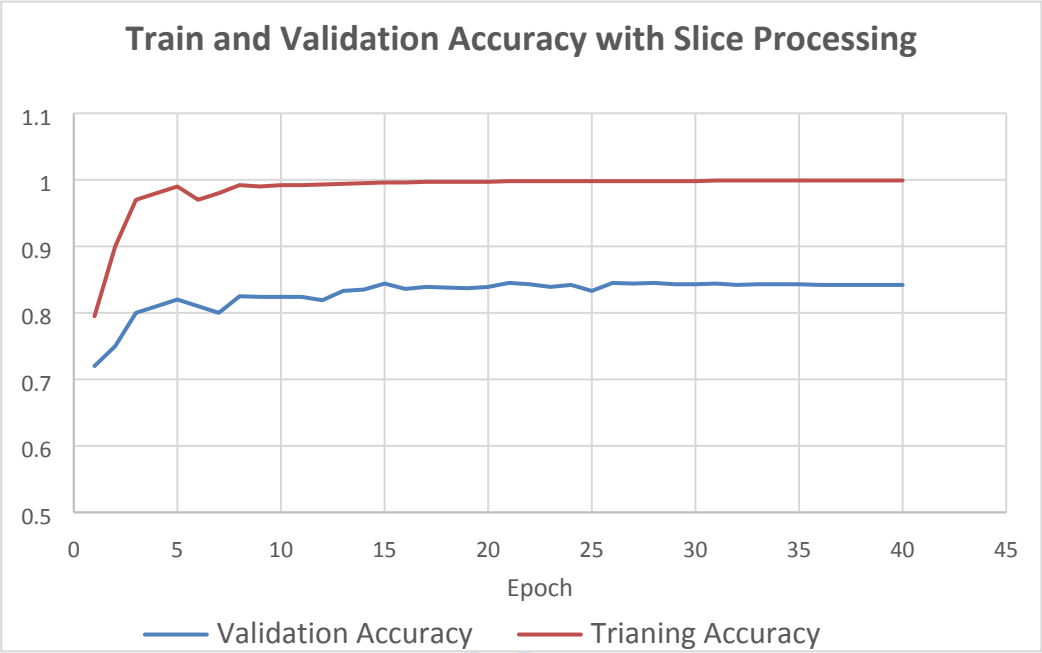

Fig. 7 Evolution of validation and training accuracy with slice processing

The interval of the validation accuracy score with 95% significance level is calculated as in Equation 8:

$$interval = 1.96 \times \sqrt{\frac{0.845\ (1 - 0.845)}{75532}} \approx 0.0013 \quad (8)$$

The results show a narrow deviation from our reported validation accuracy.

In order to evaluate the improved results achieved by using the slice processing technique before training, we use validation accuracy at slice level as a performance metric. The proposed CNN model reaches a maximum of 80.8% accuracy on the validation set when the original sized images are used as input, with similar hyperparameters tuning. That concludes that processing the slices as described above is

a major factor to improve the validation accuracy at slice level. Fig. 8 shows the evolution of training and validation accuracies when slice processing is not employed.

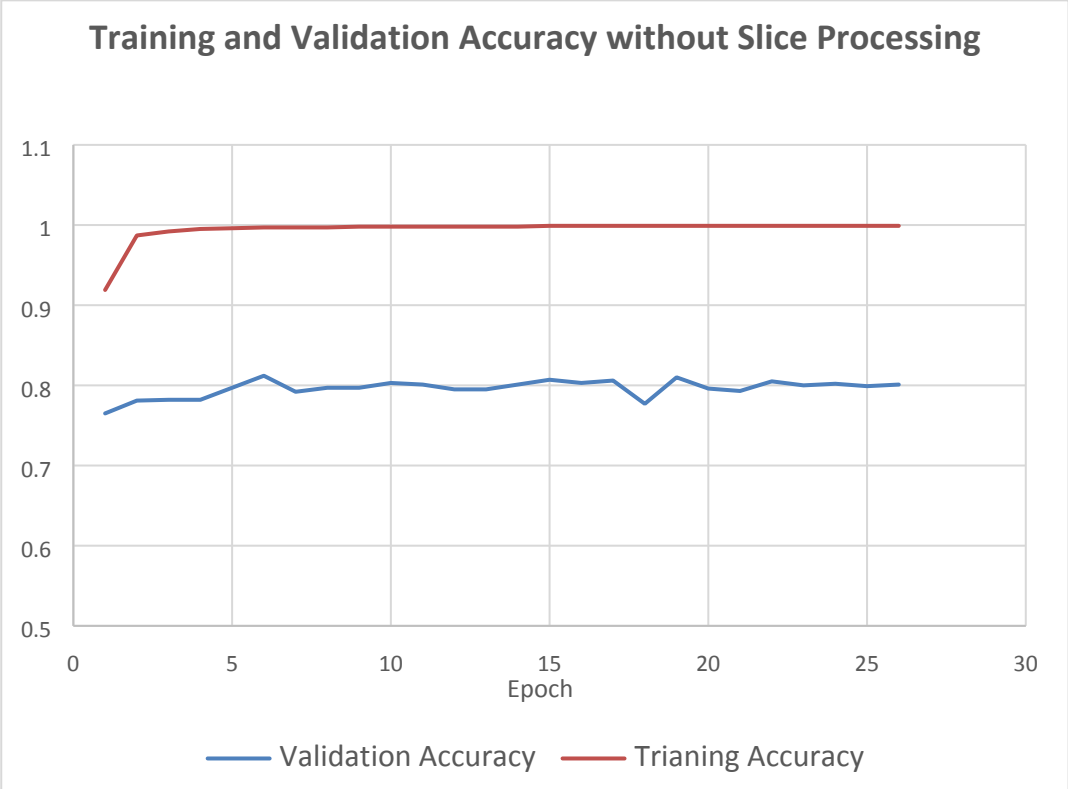

Fig. 8 Evolution of training and validation accuracy without slice processing

In terms of model over/underfitting, we realize a trend where the validation accuracy keeps increasing but the validation loss starts to increase instead of decreasing during training. We understand that accuracy and loss are not necessarily exactly (inversely) correlated, as loss measures a difference between the raw prediction (float) and the class label (0 or 1), while accuracy measures the difference between the threshold prediction (0 or 1) and the class label. So, if raw predictions change, loss changes but accuracy is more "resilient" as predictions need to go over/under a threshold to actually effect the accuracy. In here, we present our analysis to this issue and our CNN model. Two phenomena can be discussed here:

1. Some images with borderline predictions get predicted better and so their output class changes (for example, a covid image whose prediction was 0.4 becomes 0.6). This explains the regular "loss decreases while accuracy increases" behaviour that we expect.

2. Some images with very bad predictions keep getting worse (for example, a Covid image whose prediction was 0.2 becomes 0.1). This leads to a less expected "loss increases while accuracy stays the same" behaviour. Note that cross entropy loss measures the calibration of a model and when it is used for classification as we did in our methodology, bad predictions are penalized much more strongly than good predictions are rewarded. For a Covid image, the loss is $log(1 - prediction)$, so even if many covid images are correctly predicted (low loss), a single misclassified covid image will have a high loss, hence "blowing up" the average loss.

The network is starting to learn the patterns that are only relevant for the training set, which hinders its generalization capability, leading to phenomenon 2. Some images from the validation set get predicted increasing incorrectly . However, the model is at the same time still learning some patterns, which are useful for generalization (phenomenon 1, "good learning") as more and more images are being correctly classified. All that being said, our resulting model was chosen so long as it was learning and thus the model was trained for full 40 epochs. Fig. 9 shows the validation loss relative to the training loss, proving that the model is doing less better on validation, but eventually it is still learning.

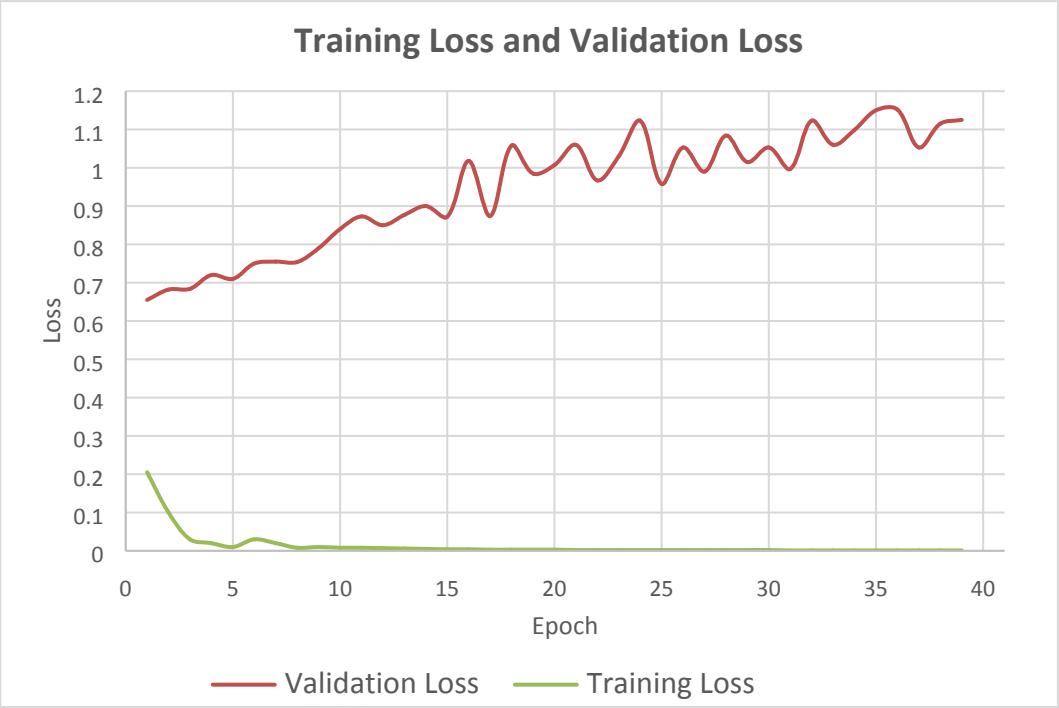

Fig. 9 Evolution of training and validation losses with slice processing techniques

The effect of increasing and decreasing the batch size on the diagnostic performance has been explored as well. In that, using a batch size of 64 allowed the model to reach a slice level validation accuracy of 83.3% with slice processing techniques. A batch size of 128 increased the resulting validation accuracy to the number reported above. With that, we can consent to using 128 batch size.

The effect of varying the number of hidden layers was also explored. The results show slice-level validation accuracy of 81.7% when a three-layers deep CNN model was used. To achieve that, the last hidden layer with 128 filters was eliminated. The validation accuracy is sufficiently less than the number reported above as compared to a four-layers deep model, when slice processing techniques are used. On the other hand, using more layers in the CNN model was tried starting by adding one layer. As we move forward in adding the layers, the detected patterns get more complex; hence there are larger combinations of patterns to capture. That is why we increase the filter size in subsequent layers to capture as many combinations as possible as explained by (December 31, 2018 posting Adrian Rosebrock on pyimagesearch; unreferenced). Therefore, we multiply the

number of filters in the fifth layer to have 256 filters, leaving all other convolutional layer settings the same. Using a five-layers deep CNN model increased the validation accuracy at slice level only trivially to 84.8%.

To assess model's complexity, number of trainable parameters were used a measuring factor for different CNN models with varying depths. The number of parameters were compared by eliminating the fully connected layer and the output layer. As observed in Table 2 validation accuracy improves as complexity increases.

Table 2 Model complexity analysis

Network's Depth	No. of Model Parameters	Increase rate in Complexity	Validation Accuracy	Improvement rate in accuracy
3 Hidden Layers	23744	-	81.7%	-
4 Hidden Layers (standard)	98122	x 4.132	84.5%	x 0.03
5 Hidden Layers	394304	x 4.018	84.8%	x 0.003

The complexity seems to increase about 4 times as we add one more convolutional layer to the CNN model. However, the accuracy only increases by a factor of 0.003 by adding a fifth hidden layer. Looking at the complexity results of different layer-depths and considering the improvement achieved on validation accuracy at slice level, consensus was on using a four-layers deep CNN model as our standard model architecture.

On the other hand, to understand how our proposed model performs the classification, Guided Grad-cam class activation visualization was used at the last convolutional layer of the model - the layer followed by a (256) flatten layer (Zhang et al. 2020). Fig. 10 shows the Grad-cam visualization for a slice in the validation set. The slice belongs to a COVID case and was correctly classified by the model. The outputs for the correct and incorrect classifications are adapted to the input image. They show that the model pays attention to:

- the lung area, and
- the posterior and anterior walls (with the anterior walls getting very strong attention values).

The results, however, show that the model also pays attention to areas outside of the lung, mainly to the patient's sitting table. For further understanding of the model's prediction mechanism, correctly and incorrectly classified slices were checked, and we can observe a similar attention distribution on the COVID cases incorrectly classified and Non-COVID cases (correctly and incorrectly classified slices) as well.

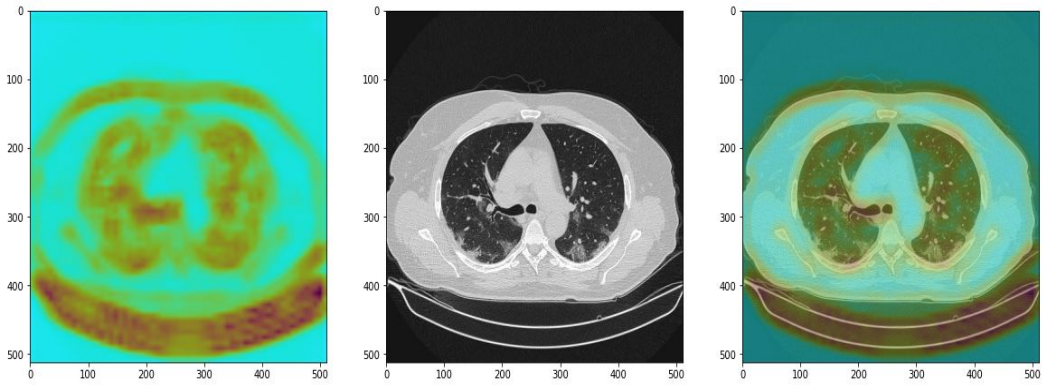

Fig. 10 Guided Grad-Cam (Selvaraju et al. 2017) visualization of the obtained results; (left) of a correctly predicted COVID slice (middle). The image on the right pane displays an overlaid version of the two

As for the slice level decision, the proposed model can sometimes incorrectly predict the uppermost and the lowermost slices as Non-COVID (specifically, 20 out of 24 extreme slices in the validation partition are misclassified). These extreme slices correspond to the anatomical regions where COVID involvement is not seen, and therefore can be considered the least representative slices for the diagnosis of the illness. Fig. 11 shows exemplary slices that are correctly classified by our proposed model, while Fig. 12 depicts exemplary slices that are incorrectly classified where the extreme slices can be observed.

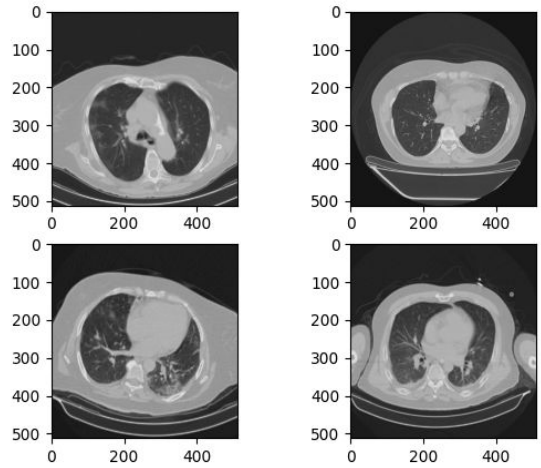

Fig. 11 Examples of correctly classified slices from COVID (right) and Non-COVID (left) cases

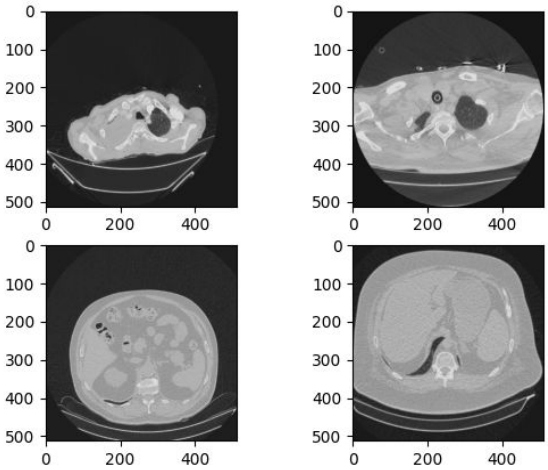

Fig. 12 Examples of incorrectly classified slices from COVID (right) and Non-COVID (left) cases

5.2 Patient Level Results

In order to obtain patient level diagnosis from slice level decisions different class probability thresholds (for slices prediction) varying in the range of [0,1] were tried as explained in Section 3.5, and the corresponding macro F1 scores were compared. Majority voting was used at patient level (for CT prediction). As observed in Fig. 13, the model achieves the highest macro F1 score with a class probability threshold of 0.40, followed by class probability threshold of 0.15. The validation accuracies at patient level when using the mentioned thresholds are 88.5% and 87.7%, respectively. With that, the

results demonstrate that at patient level a class probability threshold of 0.40 gives the best performance when used with majority voting in terms of macro F1 score. The patient level macro F1 score achieved using that class probability threshold with the proposed methodology reaches 0.882 on the validation set. The resulting macro F1 score of the proposed model comfortably exceeds that of the baseline model on the validation set. The baseline score on the COV19-CT-DB validation set is 0.70, as reported in (Kollias et al. 2021).

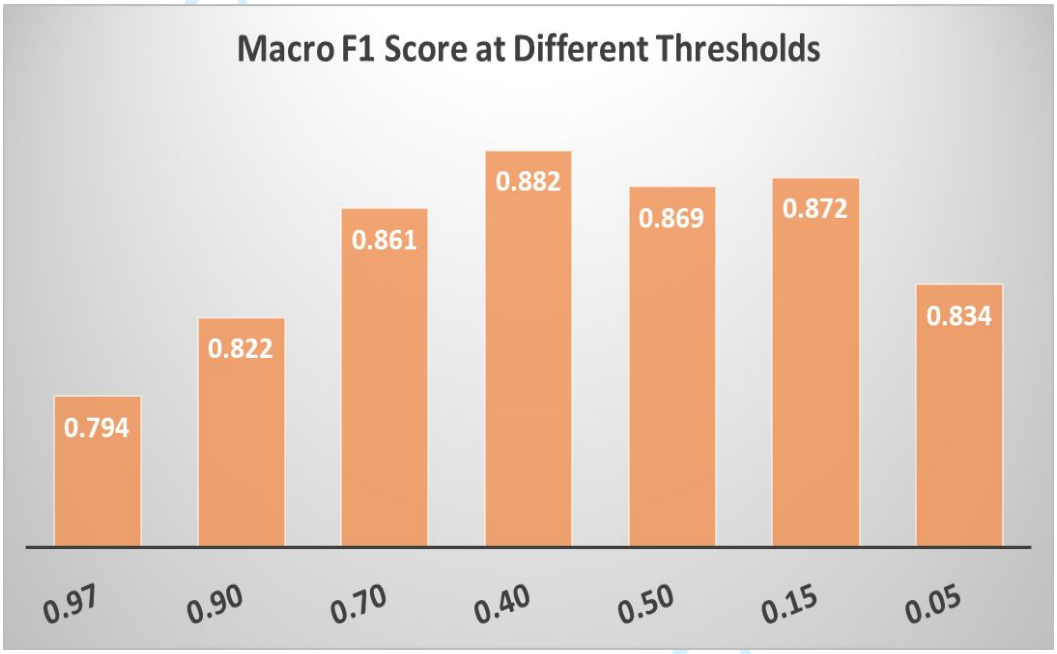

Fig. 13 Different class probability thresholds (horizontal axis) for making predictions at patient level and the corresponding macro F1 scores

In general, the model misclassifies 13 Non-COVID cases out of 209, and 30 COVID ones out of 165. Class-specific macro F1 scores of the proposed method are 0.86 for the COVID class and 0.90 for the Non-COVID. Table 3 shows the confusion matrix of the proposed method at the patient level for the best threshold value.

Table 3 Confusion matrix on patient level decision using a voting threshold of 0.40. Columns refer to actual cases while rows display predictions of the proposed model

Actual / **Predicted**	*COVID*	*Non-COVID*
COVID	135	13
Non-COVID	30	196

Despite the fact that misclassification of 13 Non-COVID cases out of 209 is less problematic than misclassification of 30 COVID cases out of 165, the model aimed mainly at increasing the quantitative results. Further, the work here considers automation of the solutions are equal as diagnostic accuracy.

Further, to validate the results the method was tested on the test partition of the COV19-CT-DB database (unseen images). On unseen dataset of images, the method's performance exceeded those of the baseline and other works. Within the context of the MIA-COVID19 competition, the teams were provided with a test partition of images. The model achieved 0.82 macro F1 score, with 0.96 F1 score for Non-COVID and 0.68 F1 score for COVID. This score is above the baseline, which is 0.67 macro F1 score.

Our proposed method did not only exceed the baseline macro F1 score (0.67), but also outperformed other alternatives entered the competition and reported accuracies on COV19-CT-DB's test partition as reported on MIA-COV19D competition's leader board. Table 4 compares our model to other alternatives on the test partition (unseen images) of COV19-CT-DB. Our team is named "IDU-CVLab" and the code was developed using Python.[2] Our proposed solution outperforms almost all state-of-the-art alternatives with an exception of the CCAT and DWCC by (Hsu et al. 2021). The method proposed by (Hsu et al. 2021) is based on ensembled deep learning models; firstly, a proposed Deep

[2]https://github.com/IDU-CVLab/COV19D

Wilcoxon signed-rank test for COVID19 Classification adopts nonparametric statistics for deep learning to find a series of slices with the most significant symptoms in CT scan. Secondly, a Convolutional CT scan Aware to further explore the intrinsic features in temporal and spatial dimensions. Finally, a three-layer perceptron is used as a classifier to take final diagnosis decisions. While our method uses one simple deep neural network, the CCAT and DWCC uses more. This makes our solution much less complex while achieving good performance.

Table 4 Comparison of the proposed method with the baseline and other state-of-the-art models on the COV19-CT-DB test partition (unseen images)

The Method	Macro F1
ResNet50-GRU (Baseline model) (Kollias et al. 2021)	0.67
A hybrid deep learning framework (CTNet) (Liang et al. 2021)	0.78
Custom Deep Neural Network (Trinh et al. 2021)	0.78
Our proposed methodology (IDU-CVLab)	**0.82**
CCAT and DWCC (Hsu et al. 2021)	0.88

6. CONCLUSION AND DISCUSSION

This paper proposes a solution for COVID-19 diagnosis using deep learning and image processing techniques. The adopted CNN model architecture was trained, validated, and tested on the recent carefully annotated COV19-CT database. Proposing simple and an efficient solution was the main aim of this study. CT scan image process techniques aimed at non-representative slices removal. The resulting slices were used as input to train a simple and symmetric CNN model where the validation accuracy at slices level was monitored for several architectural modifications. To achieve that, different number of model layers and different training batch sizes were tested. The proposed CNN method was tested with different class probability thresholds to make diagnosis decision.

Next and to achieve predictions at patient level, majority voting on each CT/for each patient scan was used. The method achieved a macro F1 score exceeding the base line

score and other alternatives on the test partition of the COV19-CT-DB database. More complex modelling techniques do not reach as high macro F1 scores as the CNN model trained. With that, we encourage researchers, programmers, and otherwise to consider a simpler and from scratch deep learning models with appropriate modifications to suit the task.

One gap in this study is that slices removal threshold may require fine-tuning for other CT datasets. The slices removal threshold should guarantee that the CT scan is not fully void of any 2D slice to allow our proposed model to make diagnosis decision at patient level.

Another gap in this work is using majority voting after selective slices removal in each CT scan. Since the removal of the slices is subject to a threshold -biased-, then the majority vote should not be the best method to make patient level diagnosis decisions. This gap could be closed by performing further tuning between the slice removal approach and the patient level voting methods. Some examples of closing such a gap can be to find a synchronised dynamic between the number and the type (uppermost, central, or lower most) of slices left in the CT and the voting method at patient level. Finding such a synchronization should maximize the performance of the proposed method for each diagnostic decision on a CT scan/patient.

Even though the rectangular region selection for slice processing along with other slice processing and hyperparameters tuning techniques improved the method's performance or the accuracy, using a manually fixed-size rectangular shape to crop the slices can still be considered another gap of this presented methodology. The reason is that this slice cropping approach will perform poorly in localizing the lung region in lowermost and uppermost slices in the CT image. Further improvement for limiting the region of interest with the lung volumes instead of processing the whole CT scan will be

a promising approach that can be realized through segmenting lung parenchyma prior to classification. This approach could ultimately improve the diagnostic performance of the proposed method.

One more gap in this research is in focusing on noise suppression of slices. In an attempt to reach an appropriate threshold for non-representative slice removal, the Gaussian filter is used to suppress the noise in the slices. However other filters could have achieved better results in reaching more appropriate image enhancement results and therefore finding better threshold for non-representative slices removing. Further improvement could include comparing different noise compressing filters or techniques to reach better performance.

ACKNOWLEDGEMENT

The authors acknowledge the work of all the medical staff and others who manually annotated the images in the COV19-CT-DB database and shared them in a relatively big dataset.

Funding: This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

There are no relevant financial or non-financial competing interests to report.

REFERENCES

Ali Ahmed SA, Yavuz MC, Şen MU, Gülşen F, Tutar O, Korkmazer B, Samancı C, Şirolu S, Hamid R, Eryürekli AE, et al. 2022. Comparison and ensemble of 2D and 3D approaches for COVID-19 detection in CT images. Neurocomputing. 488:457–469.

Anwar T. 2021. COVID19 Diagnosis using AutoML from 3D CT scans. doi:10.36227/techrxiv.14914851.v1. [accessed 2023 Feb 2].

Chen L, Fei H, Xiao Y, He J, Li H. 2017. Why batch normalization works? a buckling perspective. In: 2017 IEEE International Conference on Information and Automation (ICIA). IEEE. p. 1184–1189.

El Asnaoui K, Chawki Y. 2021. Using X-ray images and deep learning for automated detection of coronavirus disease. J Biomol Struct Dyn. 39(10):3615–3626.

Fan D-P, Zhou T, Ji G-P, Zhou Y, Chen G, Fu H, Shen J, Shao L. 2020. Inf-Net: Automatic COVID-19 lung infection segmentation from CT images. IEEE Trans Med Imaging. 39(8):2626–2637.

Fang R, Guo J, Xian B. 2022. How machine learning applied in covid-19 prevention & control. J Phys Conf Ser. 2386(1):012033.

Hao X, Zhang G, Ma S. 2016. Deep learning. Int J Semant Comput. 10(03):417–439.

Hassanin A, Tu VT, Curaudeau M, Csorba G. 2021. Inferring the ecological niche of bat viruses closely related to SARS-CoV-2 using phylogeographic analyses of Rhinolophus species. Sci Rep. 11(1):14276.

He J-L, Luo L, Luo Z-D, Lyu J-X, Ng M-Y, Shen X-P, Wen Z. 2020. Diagnostic performance between CT and initial real-time RT-PCR for clinically suspected 2019 coronavirus disease (COVID-19) patients outside Wuhan, China. Respir Med. 168(105980):105980.

Hsu C-C, Chen G-L, Wu M-H. 2021. Visual Transformer with statistical test for COVID-19 classification. arXiv [eessIV].

Huang L, Han R, Ai T, Yu P, Kang H, Tao Q, Xia L. 2020. Serial quantitative chest CT assessment of COVID-19: A deep learning approach. Radiol Cardiothorac Imaging. 2(2):e200075.

Hussain Z, Gimenez F, Yi D, Rubin D. 2017. Differential data augmentation techniques for medical imaging classification tasks. AMIA Annu Symp Proc. 2017:979–984.

Islam MM, Karray F, Alhajj R, Zeng J. 2021. A review on deep learning techniques for the diagnosis of novel Coronavirus (COVID-19). IEEE Access. 9:30551–30572.

Kiliçarslan S, Adem K, Çelik M. 2021. An overview of the activation functions used in deep learning algorithms. Journal of New Results in Science. 10(3):75–88.

Kollias D, Arsenos A, Soukissian L, Kollias S. 2021b. MIA-COV19D: COVID-19 detection through 3-D chest CT image analysis. In: 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). IEEE. p. 537–544.

Kollias D, Bouas N, Vlaxos Y, Brillakis V, Seferis M, Kollia I, Sukissian L, Wingate J, Kollias S. 2020. Deep transparent prediction through latent representation analysis. arXiv [csLG].

Kollias D, Tagaris A, Stafylopatis A, Kollias S, Tagaris G. 2018. Deep neural architectures for prediction in healthcare. Complex Intell Syst. 4(2):119–131.

Kollias D., Vlaxos Y, Seferis M, Kollia I, Sukissian L, Wingate J, Kollias S. 2021. Transparent adaptation in deep medical image diagnosis. In: Trustworthy AI -

Integrating Learning, Optimization and Reasoning. Cham: Springer International Publishing. p. 251–267.

Liang S, Zhang W, Gu Y. 2021. A hybrid and fast deep learning framework for Covid-19 detection via 3D Chest CT Images. In: 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). IEEE.

Mary Shyni H, Chitra E. 2022. A comparative study of x-ray and ct images in covid-19 detection using image processing and deep learning techniques. Comput Methods Programs Biomed Update. 2(100054):100054.

Mizrahi B, Shilo S, Rossman H, Kalkstein N, Marcus K, Barer Y, Keshet A, Shamir-Stein N, Shalev V, Zohar AE, et al. 2020. Longitudinal symptom dynamics of COVID-19 infection. Nat Commun. 11(1):6208.

Morozov SP, Andreychenko AE, Pavlov NA, Vladzymyrskyy AV, Ledikhova NV, Gombolevskiy VA, Blokhin IA, Gelezhe PB, Gonchar AV, Chernina VY. 2020. MosMedData: Chest CT scans with COVID-19 related findings dataset. arXiv [csCY].

Nayak SR, Nayak DR, Sinha U, Arora V, Pachori RB. 2021. Application of deep learning techniques for detection of COVID-19 cases using chest X-ray images: A comprehensive study. Biomed Signal Process Control. 64(102365):102365.

Opitz J, Burst S. 2019. Macro F1 and Macro F1. arXiv [csLG].

Panwar H, Gupta PK, Siddiqui MK, Morales-Menendez R, Singh V. 2020. Application of deep learning for fast detection of COVID-19 in X-Rays using nCOVnet. Chaos Solitons Fractals. 138(109944):109944.

Rahmani AM, Azhir E, Naserbakht M, Mohammadi M, Aldalwie AHM, Majeed MK, Taher Karim SH, Hosseinzadeh M. 2022. Automatic COVID-19 detection mechanisms and approaches from medical images: a systematic review. Multimed Tools Appl. 81(20):28779–28798.

Rosenthal PJ. 2020. The importance of diagnostic testing during a viral pandemic: Early lessons from novel Coronavirus disease (COVID-19). Am J Trop Med Hyg. 102(5):915–916.

Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D. 2017. Grad-CAM: Visual explanations from deep networks via gradient-based localization. In: 2017 IEEE International Conference on Computer Vision (ICCV). IEEE. p. 618–626.

Takahashi K, Yamamoto K, Kuchiba A, Koyama T. 2022. Confidence interval for micro-averaged F 1 and macro-averaged F 1 scores. Appl Intell. 52(5):4961–4972.

Talbi F, Alim F, Seddiki S, Mezzah I, Hachemi B. 2015. Separable convolution gaussian smoothing filters on a xilinx FPGA platform. In: Fifth International Conference on the Innovative Computing Technology (INTECH 2015). IEEE. p. 112–117.

Tan W, Liu J. 2021. A 3D CNN network with BERT for automatic COVID-19 diagnosis from CT-scan images. arXiv [eessIV]. [accessed 2023 Feb 2]. http://arxiv.org/abs/2106.14403.

Tang H, Ortis A, Battiato S. 2019. The impact of padding on image classification by using pre-trained convolutional neural networks. In: Lecture Notes in Computer Science. Cham: Springer International Publishing. p. 337–344.

Trinh QH, Van Nguyen M. 2021. Custom Deep Neural Network for 3D covid chest CT-scan classification. arXiv [eessIV].

Waleed Salehi A, Baglat P, Gupta G. 2020. Review on machine and deep learning models for the detection and prediction of Coronavirus. Mater Today. 33:3896–3901.

Wynants L, Van Calster B, Collins GS, Riley RD, Heinze G, Schuit E, Bonten MMJ, Dahly DL, Damen JAA, Debray TPA, et al. 2020. Prediction models for diagnosis and prognosis of covid-19: systematic review and critical appraisal. BMJ. 369:m1328.

Zhang Y, Hong D, McClement D, Oladosu O, Pridham G, Slaney G. 2021. Grad-CAM helps interpret the deep learning models trained to classify multiple sclerosis types using clinical brain magnetic resonance imaging. J Neurosci Methods. 353(109098):109098.

Deep Learning Based Automated COVID-19 Classification from Computed Tomography Images

Revised Manuscript

-Response to Reviewers on “Ready to revise” manuscript 223650179-

The authors would like to thank the reviewers and the Editorial Office for their careful review of our manuscript and their comments. Overall, the paper has been carefully edited to improve clarity and content. Before we proceed to the point-by-point response to each of the comments, we would like to summarize major changes:

1. A flow diagram showing the methodology along with proper explanation was added in the revised manuscript.

2. Model complexity analysis and accuracy test results were added in the revised manuscript.

3. The factors that improve performance as well as the limitations of our methodology through comparative and descriptive studies were added in the revised manuscript.

4. Proper reasoning and explanation behind all parts of our methodology were included in the revised manuscript.

5. References were updated.

Changes are marked with blue font in the revised manuscript. In our response, reviewers’ comments are shown verbatim in **bold**, with our replies shown in blue font. The revisions relative to the reviewer’s comments were added in the same order the email was received. The same indexing scheme is used in the revised manuscript attached to this letter using track changes in MS Word.

==========

Emailed to Kenan Morani

Dear Dr KENAN MORANI,

Your manuscript entitled "Deep Learning Based Automated COVID-19 Classification from Computed Tomography Images" which you submitted to Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, has been reviewed. The reviewer comments are included at the bottom of this letter.

I regret to inform you that the reviewers have raised serious concerns, and therefore your paper cannot be accepted for publication in Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization. However since the reviewers do find some merit in the paper, I would be willing to reconsider if you wish to undertake major revisions and re-submit, addressing the reviewers' concerns.

Please note that resubmitting your manuscript does not guarantee eventual acceptance, and that your resubmission will be subject to re-review before a decision is rendered.

You will be unable to make your revisions on the originally submitted version of your manuscript. Instead, revise your manuscript using a word processing program and save it on your computer.

Please resubmit your revised manuscript via the Taylor & Francis Submission Portal, at the following URL: https://rp.tandfonline.com/submission/create?journalCode=TCIV.

Because we are trying to facilitate timely publication of manuscripts submitted to Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, your revised manuscript should be uploaded by 12-Apr-2023. If it is not possible for you to submit your revision by this date, we will consider your paper as a new submission.

I look forward to a resubmission.

Sincerely,

Professor Joao Tavares

Editor-in-Chief, Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization

tavares@fe.up.pt

http://www.tandfonline.com/tciv

==========

We would like to thank the editor, the editorial office as well as the reviewers for allocating their valuable time and energy to review our manuscript and provide us with the valuable feedback shared. We believe the reviewers have underlined important points in reviewing our manuscript and the concerns they raised are valid in general. Accordingly, we have revised the manuscript

by addressing each comment of the reviewers one-by-one and clearly indicating a response in this letter as well as the part changed in the manuscript. We believe these revisions have significantly improved the manuscript, and we sincerely hope that both the editor and the reviewers will find these changes adequate for publication.

Comments from the Editors and Reviewers:

Reviewer: 1 Comments to the Author

Model implementation and architecture diagram and its explanation with sequence flow diagram.

We thank the reviewer for pointing out this issue. Indeed, in the revised manuscript, an architecture diagram with sequence flow was added and all parts of it were clearly explained.

complexity analysis is missing.

Thank you for pointing out this important issue. In the revised manuscript, we have included model complexity analysis using justification and comparative studies.

Accuracy test is missing.

Thank you for pointing out the issue. The accuracy analyzes in our original manuscript are further elaborated in the revised version.

Reviewer: 2 Comments to the Author

Frankly speaking, once you check the literature and compare the results with the proposed method here and with others, then it can be easily detectable that the article cannot disseminate high novelty. The proposed approach is an extendable version of the previous work.

Thank you for providing this valuable information. In our work, we propose a different method that provides a lightweight solution with unique image processing techniques. With regard to literature on the same dataset, other alternatives to our solution achieve good performance deploying deep learning methods freely. Although that resulted in good performance, the methods are still heavy to implement. Instead, we have provided a well researched and tuned method to provide diagnosis for COVID-19 virus's existence. In our revised manuscript, we included further analysis and accuracy testing to show the process and reasoning for choosing all parts of our methodology.

Reviewer: 3 Comments to the Author

In this paper authors have proposed Convolutional Neural Networks (CNN) based method with image pre-processing and hyperparameters tuning for the image classification.

Some of the suggestions are as follows:

1. Authors have discussed the CNN layer, sequencing and the parameters used. It is suggested to discuss the selection strategy also. It should be justified properly.

Thank you for the comments here. In the revised manuscript, we have added thorough discussion on the sequencing and parameters selection strategy in our methodology.

2. Authors have suggested that "The upper slices have an image of the lung with a small size, while the lower slices have a lung in the shape of a banana. However, the fixed-size cropping on the slices is an attempt to localize the lung region in a simple and less complex way." If it is fixed. Then determine the size and dimensions also. How the varying sizes will be considered. Discuss it.

Thank you for pointing out this issue. The revised manuscript fully tackled this problem and a thorough discussion of how varying sizes can be considered using our methodology is presented.

3. Authors have suggested that "The Gaussian filtering was chosen over other filters because it is less computationally expensive to implement. The 2D Gaussian filter is separable to two one-dimensional filters" Also name those filters along with the justification by citing any source for the same.

Thank you for pointing out this issue. In the revised manuscript, we have theoretically and mathematically explained why and how the Gussian filter is used and preferred in our methodology with comparison to other more complex alternatives.

4. Authors have suggested that "The threshold was chosen carefully so that at least one slice in any CT scan volume not removed which will allow to take the final diagnosis decision for all CT scans or all patients." It is suggested to prove it with error rates.

Thank you for the suggestion of adding the error rates. Indeed, in our revised manuscript, we focused on error rates and we compared two possible thresholds, showing how we decided on the method's standard threshold. Limitations to this part of the methodology were also included in the new manuscript.

5. It is suggested to consider variable epochs and compare the results.

Variable epochs were considered and early stopping based on validation accuracy training improvement was tried and explained in our revised manuscript.

6. Include a block diagram of the complete approach. Proper explanation is also needed

Thank you for raising this important issue. In the revised manuscript, a block diagram showing the methodology was added.

7. The improvement should be specified properly with the reasons, justification and comparative study.

Thank you. The revised manuscript included thorough and clear reasoning of the improvement through justification and comparative study.

8. What are the major factors that increase the performance? Justify and discuss it in detail.

Thank you for pointing out this issue. We have revised our manuscript to show the major factors that increased the performance through comparative and descriptive studies.

9. What are the limitations of this study?

Thank you for pointing out this issue. The revised manuscript discusses the limitations and the gaps including major limitations to implementing our methodology.