SARS-CoV-2 Detection Through Chest CT Screening Using Deep Learning

ABHISHEK NAGRECHA[1], RAJESH AYTHA[2] AND DR. THIAGO E. ALVES DE OLIVEIRA[3], (Assistant Professor (PhD), Lakehead University)

[1]Student, Lakehead University, Thunder Bay, Ontario, Canada P7B 5E1 (email: nagrechaa@lakeheadu.ca)
[2]Student, Lakehead University, Thunder Bay, Ontario, Canada P7B 5E1 (email: raytha@lakeheadu.ca)
[3]Supervisor, Lakehead University, Thunder Bay, Ontario, Canada P7B5E1 (email: talvesd@lakeheadu.ca)

This work was supported in part by the Department of Computer Science, Lakehead University, Thunder Bay, Ontario, Canada P7B 5E1.

ABSTRACT The communicable disease caused by the extreme acute respiratory syndrome is severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). As of April 2021, more than 140 million people around the globe have tested positive for the disease and in access to 3 million people have lost their lives. To this end, CT imaging has been suggested as one of the primary screening methods which can be adopted as a substitute to the testing of RT-PCR. Compelled by this, we have accurately classified CT scans of patients with computing-time accuracy and reliability by implementing convolutional neural network architecture. The novelty of research lies in the fact that by performing crucial pre-processing steps and using a simple CNN architecture with a small number of parameters, we successfully distinguished scans of COVID-19 patients from normal patients. In addition, COVID-CTset is the dataset we have utilised. which contains 48260 CT scan images from 282 normal patients and 15589 CT scan images from 95 patients with COVID-19 infections. In contrast with other state-of-the-art standards, our proposed approach is less hardware-intensive and shows encouraging performance in terms of classification accuracy of 95%, F1-score of 96%, 98% of recall score, and a precision score of 94%. we are convinced that our proposed system will aid clinicians by providing interpretable predictions to validate their assessment towards COVID-19.

INDEX TERMS Covid-CT Scan, Computer Vision, Deep Learning, Image classification, Thresholding

I. INTRODUCTION

The coronavirus infection (COVID-19) is a global pandemic that first emerged in December 2019, in Wuhan, China, the capital city of mainland China's Hubei province [1]. When individuals are in close vicinity, the transmission of COVID-19 is quicker, and the disease has been declared a pandemic by the World Health Organisation (WHO). Studies have shown that more than 60 % of patients lose their lives as soon as they advance to the stage of serious or critical illness [2]. Since December 2020, multiple variants of the virus have been found, however, a variant now known as B.1.1.7, quickly became the most common version of the coronavirus in the United Kingdom, accounting for about 60% of new COVID-19 cases in December. It is being speculated that some of these mutations may enable the virus to spread faster and more infections can result in more people getting very sick. However, on a positive note, as of March 2021, more then 7 different vaccines were made available to protect the people from the virus. The very first national immunisation campaign started in early December 2020, and as of February 15, 2021, 175.3 million antigen doses have been delivered successfully. Most of the federal governments around the globe, had imposed travel restrictions and sealed their borders to stop any further spread. To add to that, a set of rules have been established for travellers entering a new territory under the Quarantine Act. The most prominent illness signs are fever and cough. There may be other effects, including chest pain, growth of sputum, and a stuffy nose that appears to have a significant effect on patients and health services around the world. There is indeed an urgent need for quick and reliable screening methods to classify patients afflicted with COVID-19 to ensure prompt isolation and care in the battle against this unusual disease.However, the fast spread of COVID-19 is one of the major threats, with an average of 1-3 individuals getting infected by the disease upon contact with an infected person [3]. This advocates that they are more likely to infect 15-35 other individuals if 10 individuals are COVID-19 positive.But, unless intervention steps are enforced, COVID-19 could infect a very large number of ethnic groups in the coming days since transmission from individual to individual has been clearly identified [4]. Besides, a new gene close to the virus causing SARS and MERS is

caused by COVID-19. While the spread of COVID-19 has a lower mortality rate than the SARS and MERS outbreaks [5], but it spreads more quickly and has infected larger inhabitants than the SARS and MERS outbreaks [5]. The reverse transcription polymerase chain reaction (RT-PCR) research method [1] is currently the main diagnostic technique for the primary means of screening for COVID-19, a laboratory protocol that interacts with other ribonucleic (RNA) and deoxyribonucleic acids (DNA) to determine the volume of specific ribonucleic acids using fluorescence. While RT-PCR can identify the strain of SARS-CoV-2 that causes COVID-19, in some instances it gave negative testing results even though sufferers showed improvement on follow-up chest computed tomography (CT) scans [5] and its susceptibility is also varied, depending on the method of sampling and the period after symptom onset [6] [7].

Several studies including [6] have advocated the use of chest computed tomography (CT) scans rather than RT-PCR, because its available in all major public health facilities, including hospital emergency rooms (ERs), and even at rural clinics, and its sensitivity is also high, more importantly considering the limitations of RT-PCR and the time required to obtain data. The method of CT scanning has a quicker processing time than the RT-PCR test and can offer more detailed pathology-related details and explain the extent of lung involvement [8]. Additionally, the exposure of COVID-19 symptoms in the lower parts of the lungs has a higher accuracy when using CT scans than that when using RT-PCR [9]. Likewise, radiological imaging is also considered an important screening method for COVID-19 diagnosis [8]. Ai et al. [10] demonstrated the consistency of the radiological history of COVID-19-associated pneumonia with the clinical nature of the condition.Just about all COVID-19 patients have shown similar characteristics on CT scans, including ground glass opacities (GGO) in the preliminary phase and pulmonary consolidation in the final phases. Also, deep learning models can be employed to initially evaluate a COVID-19 patient as an alternative solution to traditional approaches that are time-consuming and labour intensive. However, they cannot exclusively address the whole problem due to the high volume of re-examinations of infected people who wish to know the progression of their illness.

Therefore, this paper aims to propose a deep learning-based system that appends relevant pre-processing steps to automatically detect COVID-19 from CT scan images. In the proposed system, CNN is used for feature extraction and classification of COVID-19 based on those features. In fully connected networks, the layers are fully connected and the nodes between layers are connectionless and process only one input. Hence, the 2-D CNN layout feature combination improves classification greatly.

The paper is organized as follows: A review of recent scholarly works related to this study is described in Section 2. Problem description and formulation, is presented in Section 3. The proposed solution including dataset collection and description is presented in Section 4. Detailed preprocessing steps can be found in Section 5. Our convolutional neural network description along with 5 and 10-fold cross validation can be found in Section 6. The different scenerios and experimental conditions are described in Section 7. The results and comparative performance of the proposed deep learning system are provided in Section 8. The discussion is given in Section 9. Section 10 concludes the paper and the future work is described in Section 11.

II. LITERATURE REVIEW

In [11] they propose a method to diagnose COVID-19 from CT scan samples of the individuals. They enforced their method using the ResNet50V2 network and a modified feature pyramid network (FPN). one thing that was questionable about their work was their usage of FPN for feature extraction, which is generally helpful for object detection. After running two phases, their system was able to determine the condition of the patient using a manually selected threshold. Another thing which we found surprising was that they had not done much pre-processing on the CT scans and actually fed the un-processed scans to the networks for classification. Several papers, including [12], have recapitulated chest computed tomography (CT) as a useful tool in the assessment of suspected SARS-CoV-2 infection patients.

We also studied another similar research [13] which used the same data and discovered some discrepancies. we felt that there was a need for doing additional analysis of the explainability results for coming to an acceptable outcome. Identification of key patterns in the CT images was also missing which would have aided clinicians in manual screening.They implemented Light Gradient Boosting Machine (LightGBM) and Cox proportional-hazards (CoxPH) regression models for prognosis prediction. According to [14], the lack of polished radiologists risks the availability and adequacy of COVID-19 screening facilities in affected countries. Suspicious patients anywhere, especially in developed countries, have fair access to the correct diagnosis, appropriate treatment, and isolation through the implementation of AI diagnostic algorithms. With 10,250 CT scans from three centers in China and three publicly accessible databases, they developed clinically representative large-scale datasets. They developed both CT-based and CXR based diagnostic systems and tested them using paired data to explain the relative efficiency of CT and CXR for the identification of COVID-19 [15].

DeepPneumonia was developed to help doctors diagnose the pneumonia-causing COVID-19 and localise the key lesions. Three key measures were been developed for their fully automatic lung CT diagnostic method. Although, after going through their research we found their pre-processing steps highly debatable, specifically their selection of libraries. Also, the data which was used for this experiment was relatively small, having only 88 COVID-19 patients. To extract the top-K information in the CT images and corral the image level predictions, an Information Relation Extraction neural network (DRE-Net) was developed. Both 2D local and 3D

global model characteristics were being collected. As the backbone, COVNet structure consisted of RestNet50, which took as input a set of CT slices and created functionality for the corresponding slices [16]. Nevertheless, there were few inconsistencies in this research, such as the random choice of CAP from Aug'16 to Feb'20, and the analysis relied only on whether one sample was of COVID-19 or not, but the classification of the virus into various levels of severity were not discussed. Also, we have seen a variety of papers which have put-to-use well-known convolutional neural networks to separate COVID-19 infection from Non COVID-19 classes, including [17] and [18].In [19] features were extracted from each CT lung image by QDE and CNN methods.Results were achieved by combining QDE–DF features and LSTM was used for classification. we checked several papers for some detailed studies on CT image pre-processing and found the methodology of [20] and [21] useful. Besides, the submitted cadre was contrasted with commonly accepted FCN [22].

We were impressed by the pre-processing steps proposed by [20], since they were aligning with our methodology and hence we have included them in our research.Histogram thresholds were used to separate the the backdrop of the CT scan by thresholding the sensitivity values by the average value of each CT slice independently.In [23] they applied several random on-the-fly data augmentation strategies during training, including cropping of square patches at the center of the input frames, random rotation and horizontal reflection etc.But one impediment of this research would be the way they handled their classification task, that is, they failed to distinguish CAP from COVID-19. They had described a fully convolutional network (FCN) when it comes to [21], trained end-to-end, pixels-to-pixels on semantic segmentation. Since their redefinition of classification networks, generated output maps for inputs of any size, the output parameters were generally decreased by subsampling.Subsampled classification networks to retain low philters and fair computing requirements. We found [24], [25], [26] useful for our project framework and hence we reviewed them in detail. [24] recommended the use of an attentive fully convolutional network that could concentrate on contaminated areas of the chest, allowing a more detailed forecast to be made.

They trained the model on a relatively a small dataset comprising of roughly 2000 CT images and also presented a diagram of their prototype attention maps and demonstrated how close they were to the board-certified radiologists' manually identified infected regions, we investigated [25] which discusses the approach and performance of the computer vision CT-based pathological condition diagnosis, having selected some recent representative works to provide an overview of their effectiveness. Based on their role in contagion management, they grouped the approaches listed into 3 categories: computed tomography (CT) scans, X-ray imaging, and control and prevention.Last but not the least, [26] implemented a computer-aided diagnosis (CAD) web site for online identification of COVID-19.In this research, a public chest CT scan database was utilized, including 746 participants. To identify the most effective model for the hybrid system, a range of very well-known deep neural network architectures consisting of ResNet, Inception and MobileNet were inspected. To differentiate between COVID-19 and safety controls, a variation of the tightly connected convolutional network (DenseNet) was chosen to minimise image dimensions.Nevertheless, the limited data set size they have chosen was a substantial drawback because the training phase for deep neural networks demands a significant number of samples and such limited samples trigger over fitting.

III. PROBLEM DESCRIPTION AND FORMULATION

The most significant thing about the current coronavirus, as we have described before, is it's rapid and wide-spreading ability. As a consequence, properly recognizing and quarantining the effects of patients with illness plays an important part in stopping the disease. [5]

There are many approaches, including reverse transcription polymerase chain reaction (RT-PCR), isothermal nucleic amplification test, antibody test, serology test, and medical imaging, for conclusive detection of COVID-19. The primary method of diagnosing COVID-19 and other viral diseases is RT-PCR. [1] [11]. For some of the experiments, however, the methodology is restricted as higher skills and exploration are needed to create new therapeutics. Besides, in most infected regions around the world, the shortage of diagnostic kits is encouraging scholars to come up with innovative and simpler approaches to make an accurate diagnosis. [9] [10].Researchers are analyzing CT scans and X-rays to diagnose COVID-19 thanks to the availability of medical imaging equipment in most healthcare facilities. Infections in the lungs of people with new coronaviruses that may help detect the disease are present in most patients with COVID-19. Pneumonia caused by the new coronavirus was seen in the study of CT scans of COVID-19 patients. [13] [12] Many patients with COVID-19 signs have X-rays and CT scans of their lungs at least four days apart, revealing infections that indicate the presence of a new coronavirus in their body. [5] [14].While medical imaging for the correct diagnosis of COVID-19 is not approved, but it can be used as a secondary diagnostic tool for quarantining the suspect and preventing the virus from being spread to others during the onset of symptoms. [25]. Our primary objective in this research was to employ the deep-learning methods to accurately diagnose the virus from the CT scans due to its ability to derive rich features from multimodal clinical databases. Also, deep learning applied technology has recently seen considerable progress in the field of medical data processing. [23] [18].Deep learning based methods for chest CT data processing and classification have been developed effectively in the ongoing COVID-19 pandemic. [16] [15] [14].

In addition, deep learning algorithms for COVID-19 tracking, scanning, and hospital stay prediction has been suggested. [23].The system could also reliably differentiate COVID-19 cases from CAP and NP patients. The precise location of the lesions or inflammations caused by COVID-

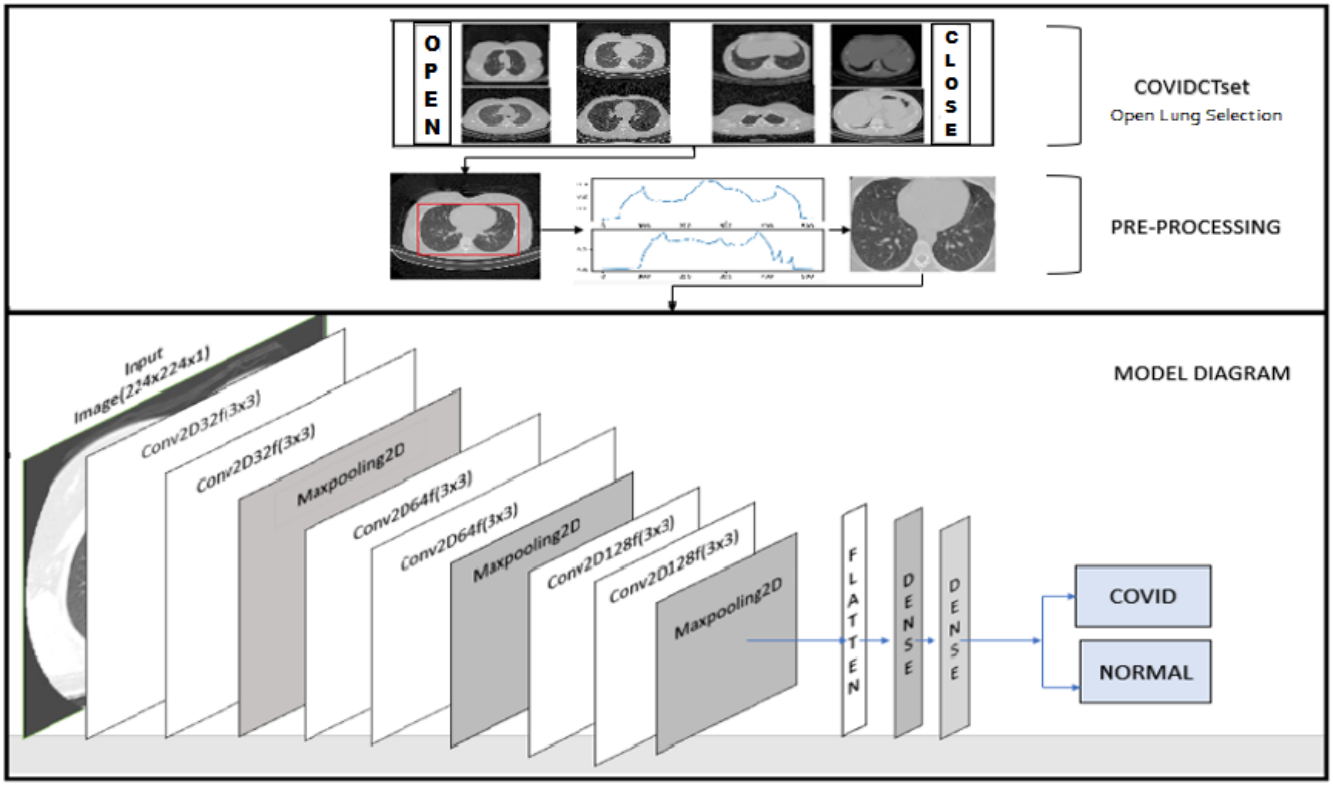

FIGURE 1. System Overview: A deep learning framework based on series of lung CT slices for the classification of COVID-19 and Normal.

19 can now be known, and would also therefore theoretically include guidance on patient severity to direct the triage and care that follows [12].

To further support our approach we have studied numerous papers which demonstrated the use of deep learning in other domains including accurate diagnoses of tumors and infections caused by various diseases.This approach has also been used for numerous medical images, such as neuro and skin lesion segmentation [11], Breast Applications Lesions and pulmonary nodules [3].

IV. PROPOSED SOLUTIONS AND COMPARISON TO THE LITERATURE

For the above-mentioned problems related to the RT-PCR and other testing methods we came up with a solution to work with CT scan images of both patients with Covid related pneumonia and Normal scans. We have used deep learning based CNN architectures to accurately predict the status of the patient through the lung HRCT radiology scans given as the input.The primary reference paper for this research has been [11]. However, we have used an entirely unrelated approach for our research. We have done extensive pre-processing of the CT scans before feeding it to neural network architecture. We observed that the author who originally worked on this dataset, decided to stay away from any pre-processing steps and worked with raw CT scan images. Also, the author validated his results using pre-trained models including ResNet50V2, Xception and Feature pyramid network (FPN). Moreover, the author worked with 11302 scans and included only 450 Covid-19 scans to test the network's performance. On the other hand we have utilized 17548 scans with 8774 Covid-19 images. Also, we have used simple CNN to validate and have achieved finer results then the author in terms of overall accuracy, precision and recall scores. We believe that applying important pre-processing steps has made it possible to achieve these excellent results. You can get more insights about this in the Results section.

A. DATASET COLLECTION AND DESCRIPTION

When it comes to doing research, the most important resource is the availability of good data. one might have path-breaking solutions but until and unless it's been tested on real-time data, it's of no use. The dataset we utilized is named COVID-CTset and was derived from [11] . it is constructed of two main sections. The first section with the name of (TrainingValidation) contains the CT slices for training and validating the networks, and the second section contains the entire data for all the patients. It was retrieved from Negin radiology from March 5 to April 23, 2020, located at Sari in Iran. A SOMATOM Scope model and a Syngo CT VC30-easyIQ software version were used by this medical center to collect and visualize patient's lung HRCT radiology images. The specification of the exported radiology images was a 16-bit DICOM grayscale with a resolution of 512*512 pixels.

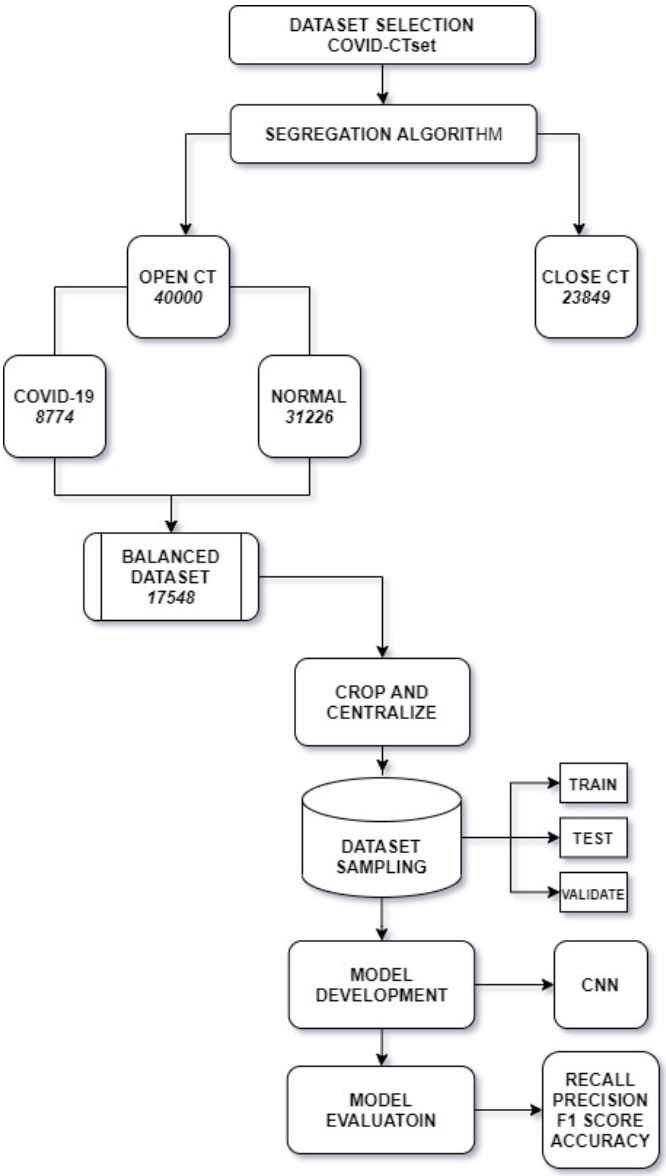

FIGURE 2. Flowchart of our Research Work

The dataset consists of 15589 CT scans belonging to 95 patients infected with COVID-19 and 48260 scans of 282 normal patients. Each patient has three different folders, which includes the CT scans captured from the CT imaging device with three independent thickness types namely SR_2, SR_3, and SR_4. As the patient's CT slices were originally available in DICOM format, it was converted to TIFF format by [11], which retains the very same 16-bit grayscale data but does not infer the private information of the patients.

B. POPULATION CHARACTERISTICS

The plotted Stacked bars in figure 3 depicts the dataset population distribution. its interesting to observe that the data was almost balanced as per the gender of the patients, since there were in total 377 patients, 192 male and 185 females. In total there were 56 males having covid-19 in comparison to 39 females. Also the minimum age of the patient having covid-19 was as low as 17 years(male) and the maximum being 82 years(female).

C. PRE-PROCESSED DATASET

Once we analyzed the original raw dataset, we knew that it would require some extensive preprocessing before the neural networks can be trained. You can read about these steps in the next section. firstly, we re-scaled the scans from 512*512 to 224*224 for less RAM utilization, however, we didn't change image format to prevent any loss of resolution. Once that was done, we filtered out the closed lung CT scans from the whole dataset and decided to work with only the open lungs.

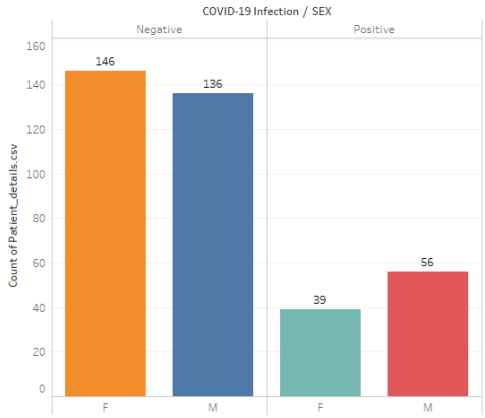

FIGURE 3. Dataset Distribution

Furthermore, after data exploration we found out that the dataset was imbalanced since it only had 15589 CT scans of Covid-19 patients and 48260 CT scans that of normal patients. Next, out of the 15589 CT scans of Covid-19 patients we filtered out 6815 closed lungs, hence we decided to work with 8774 open lungs of Covid-19 patient. Once that was done, we randomly selected other 8774 CT scans from the open normal patient cohort and constructed a dataset containing 17548 CT scans images. Next, we cropped these images in a way that we only trained our network on the open lungs and not on the surrounding region (containing dark pixels) to avoid overfitting. We intend to make this dataset available for other researchers to train their models in the future.

V. PRE-PROCESSING

Below we have identified six crucial pre-processing steps which will speed up the deep Convolutional Neural Networks training process and improve the overall classification performances. However, its noteworthy that we have not utilised DL methods for executing these steps.

1) Generating the CSV file

Since we had around 64k images, it was getting inconvenient to work with them efficiently, so just to get everything

running smooth we loaded all the images into a pandas data frame and converted them into NumPy arrays which took around six hours to complete and saved it on our Google drive.We feel that this file made our overall task much simpler and convenient, since all the pre-processing steps and our deep CNN model with 5 and 10-fold cross validation ahead makes use of this crucial file. The CSV file consists of 63849 records corresponding to the total CT images that we have and 14 unique attributes including index (unique index value starting from 0 to 63848), loc (uniques location of every image on our Google drive), img (NumPy array correnponding to each CT scans), maximat2b (location where Maxima's of each CT scan are achieved- top to bottom for cropping the CT scan), maximal2r (location where Maxima of each CT scan are achieved left to right for cropping the CT scan), colsavg (values of average pixel intensity starting from 0 upto 512 column wise), rowsavg (values of average pixel intensity starting from 0 upto 512 row wise), croppedshape (shape of the cropped CT scan), maxnpfull (the maximum pixel value that we got for an image, this value was required for normalizing the images) , FullAvgPxlDen (average pixel intensity of the whole image), CrpdAvgPxlDen (the average pixel density of the cropped image, we used this to segregate the open and closed lungs). CentAvgPxlDen (average pixel intensity of the central region of the image i.e., 25-75% of the region), OpenClose (classifies image as open or closed), and label (labels the classified image as covid(1) or normal(0)).

	index	loc	img	maximat2b	maximal2r	colsavg	rowsavg	croppedshape	maxnpfull	FullAvgPxlDen	CrpdAvgPxlDen	CentAvgPxlDen	OpenClose	label
0	0	/content/	[0 0 0 ... 0	(99, 245, 427)	(152, 235, 378)	[	[	(226, 328)	2191	439	622	672	Open	0
1	1	/content/	[0 0 0 ... 0	(101, 244, 424)	(160, 263, 374)	[	[	(214, 323)	3039	475	583	634	Open	0
2	2	/content/	[0 0 0 ... 0	(169, 274, 343)	(151, 249, 375)	[	[	(224, 174)	2212	424	885	895	Close	0
3	3	/content/	[0 0 0 ... 0	(169, 254, 378)	(150, 271, 369)	[	[	(219, 209)	2291	476	1031	1031	Close	0
4	4	/content/	[0 0 0 ... 0	(93, 261, 434)	(148, 218, 379)	[	[	(231, 341)	2246	348	623	688	Open	0
5	5	/content/	[0 0 0 ... 0	(99, 245, 425)	(154, 257, 378)	[	[	(224, 326)	2194	451	606	659	Open	0
6	6	/content/	[0 0 0 ... 0	(166, 272, 344)	(150, 237, 376)	[	[	(226, 178)	2176	405	825	843	Close	0
7	7	/content/	[0 0 0 ... 0	(126, 245, 393)	(169, 224, 352)	[	[	(183, 267)	2564	465	751	808	Close	0
8	8	/content/	[0 0 0 ... 0	(144, 231, 377)	(164, 249, 341)	[	[	(177, 233)	2532	472	866	891	Close	0
9	9	/content/	[0 0 0 ... 0	(94, 261, 434)	(150, 236, 379)	[	[	(229, 340)	2251	339	592	646	Open	0
10	10	/content/	[0 0 0 ... 0	(169, 268, 341)	(169, 259, 374)	[	[	(205, 172)	2422	457	966	978	Close	0
11	11	/content/	[0 0 0 ... 0	(105, 270, 417)	(164, 171, 369)	[	[	(205, 312)	2276	481	570	628	Open	0
12	12	/content/	[0 0 0 ... 0	(96, 258, 428)	(150, 252, 381)	42.25	[	(231, 332)	2226	355	608	663	Open	0
13	13	/content/	[0 0 0 ... 0	(138, 245, 386)	(164, 241, 347)	[	[	(183, 248)	2593	467	800	849	Close	0
14	14	/content/	[0 0 0 ... 0	(160, 170, 355)	(167, 333, 341)	[	[	(174, 195)	2441	473	944	933	Close	0
15	15	/content/	[0 0 0 ... 0	(119, 244, 398)	(164, 201, 358)	[	[	(194, 279)	4095	460	699	754	Close	0
16	16	/content/	[0 0 0 ... 0	(99, 247, 428)	(152, 222, 377)	[	[	(225, 329)	2339	411	618	669	Open	0
17	17	/content/	[0 0 0 ... 0	(166, 275, 343)	(169, 247, 374)	[	[	(205, 177)	2377	443	933	945	Close	0
18	18	/content/	[0 0 0 ... 0	(100, 244, 424)	(155, 269, 378)	[	[	(223, 324)	2105	459	593	642	Open	0
19	19	/content/	[0 0 0 ... 0	(91, 273, 357)	(149, 234, 377)	[	[	(228, 266)	2329	383	746	793	Close	0
20	20	/content/	[0 0 0 ... 0	(103, 277, 418)	(169, 170, 368)	[	[	(199, 315)	2306	481	556	632	Open	0

FIGURE 4. Final.csv file

2) Segregation Algorithm

Since a CT scan of a patient is just like a video, I.e., it contains a sequence of images which are both open and closed, we needed to filter out the closed lungs and only used the open lung CT scans to train our network.This is because, while training, if the network sees these cases (closed lungs); it will result in inaccurate detections, and the model might under-perform. Usually, these images are present at the beginning and end of a CT scans. Regrettably, in our dataset, scans are not even on one scale, and the position of the lung varied for different patients hence it was challenging to get this done. The scan in the area with less dark pixels than the threshold are the one in which the lung seems to be almost closed, and the image with more dark pixels is the one that is visible inside the lung. The below figure 5 shows the three different lung types in our dataset. We decided to only work with CT scans present in the first row of the figure, that are completely open lungs. Our segregation algorithm discards the scans present in the second and the third row, that are partially open and closed lungs.

A) Calculating the Average Pixel Intensity (API)

The solution of the above-mentioned problem was to find out the average pixel intensity (API) from all the CT scans. Once that was found all we needed to do was to compare it with the calculated threshold value and that would give us segregated open and closed lungs.

Centered square region of interest

In this method we took a sample of 1000 images of both open and closed lungs and calculated the API from these sample and made it as our deciding factor in filtering out the closed lungs. We calculated the API, by just focusing on the central region. To do so, we set a region in the middle of the scans i.e., starting from 25% till 75% of both height and width. In all the scans, this area should be at the core of the lung, so that the variations in this area are shown by the open and closed lungs. For example: for our 512*512 images, we considered a region of 128 that will go up to 384 which is the starting 25 to 75 percent of the region for both the x and y directions.

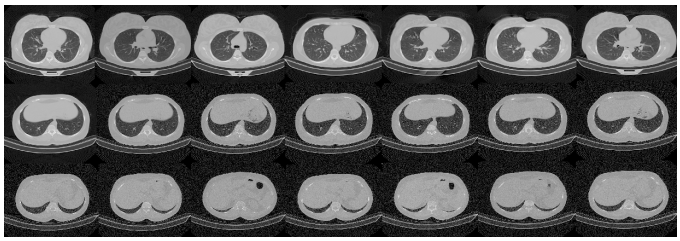

FIGURE 5. top-to-bottom CT slices of open, partially open and closed lungs.

Also, one extremely important aspect of the above proposed algorithm is that it works solely based on the API value provided, which in turn depends upon the set of scans that are taken as an input to calculate the value for both open and the closed lungs. Next, to calculate the API, we first took the location of all the CT scan images from the CSV file, once all the images were fetched, height and width of all scans were measured, and based on those we got the pixel values from individual location. Then these pixel values were added on to the counts value which was later divided by the scans height and width value.

For doing this we created a function 'pixelslist' which took three parameters, the first being, imglist: that is the list of location of images to calculate the API value, the second being Mode: we considered two distinct modes for this, one being full, which considers the whole CT scan image to calculate the API and the other being central, which would only consider the central region where the lungs are located (25-75% of both height and width) and the last parameter being the normalization: that is to visualize the saved segregated CT scans on our standard monitors.

B) Calculating the Threshold Value

Once we obtained the API of the scans, we needed to calculate the threshold value. so,one way to calculate the threshold was to simply take the average of the open and the closed mean counts i.e., *threshold = (openmean+closedmean)/2* where for the openmean value we covered 99% of the dataset for the open lungs with a value of '792' and for the closedmean value we subtracted two standard deviations (hence covering 95% of the dataset for the closed lungs) values from the mean value resulting to a value of '713.84' (901.24- 93.7 – 93.7) hence giving us a threshold value of 752.5 (taking the average) but this method had some limitations and resulted in more impurities in the open lungs data and hence we decided to give it a skip.

The other way which we felt was more appropriate was to first find out the threshold value by using the above-mentioned formula, and then get the maximum value from the openpixels list value. Once we had both these values, all we needed to do was to consider the minimum value out of those, and that would be our proposed threshold value.

For doing this we created a function 'threshold' which had three parameters, the first being openlist: that is the list of folder containing randomly chosen open images, the second being closedlist: that is the list of folder containing randomly chosen closed images, and the third being the normalization parameter which when set to 'True' makes the images visualizable. The main idea behind calculating this threshold was to compare it with the API of each scans and segregate it into open or closed lung. We observed that in case of an open lungs more dark pixels were observed in the central region as compared to that of partially open or closed lung. Based on this analysis we came to a logic that images which had an API value smaller than that of threshold value were open lung CT images, and the ones which had an API values greater than that of the threshold value were the closed CT scans.

Once we obtained both the threshold and the average pixel intensity values, we created a function 'segregate' which took two parameters, the first being 'pixellist': that is the list of API count of all the images and the second being threshold: that is the value which will be considered to bifurcate. This returned us with two lists, open and closed with index values present in each. We got around 40000 Open CT scan images out of 63849 images. Indicating that nearly 37% (23849) of the total dataset images to be partially or completely closed. Since we did not have any assistance from a radiologist, we had to take decision by ourselves to define what would be considered as an open and closed lung for this problem. We discovered several scans in the dataset which had partially open lung and decided to label them as closed since it might have lost some important features where the coronavirus may be present and that would lead to inaccurate predictions.

3) Cropping Function

The main idea behind cropping out the lungs was to reduce the amount of information we will be passing to the convolutional neural networks. So that the model only has perform one task of classifying the scans. For that we summed all the columns of the vectors of the CT scan to get the row which we need to select. This was done for all the images since the lung size differs from person-to-person. We were able to find the bounding boxes for all the images by calculating the histograms.The procedure was quite straightforward. The CT was initially scanned in one pass, with a running count of the number of pixels identified at each intensity value retained. This data was being used to create an appropriate histogram. We got bi-modal histograms because the CT scans were ideal for thresholding, i.e., the pixel intensities were clustered into two well-separated values. A suitable threshold for separating these two groups was found somewhere in between the two peaks in the histogram.

The histogram of an image in computer vision typically corresponds to a histogram of pixel intensity values. it is a graph that displays the number of pixels in an CT scan image at each different intensity point.For our 16-bit grayscale image there are 512 possible intensities, and so the histogram will graphically display 512 numbers showing the distribution of pixels amongst those grayscale values.However, all the CT images had lungs of varying sizes, so we needed to come up with a solution to crop accurately.

Our Proposed Approach for cropping the scans

After experimenting with diverse approaches, we understood that we needed to automate the cropping function for every CT scan images and hence we came up with this approach. For this firstly, the CT scan images were converted into NumPy arrays, for us to calculate the mean values axis wise. That is, x-axis in each graph showed the location (in terms of pixel) at which the average values were plotted, here the average values can be referred as the intensity of the pixels. All the values of the histograms were collected in a separate list and the maxima were spotted in a range of initial and final 33% length (both vertically and horizontally) of the CT scan image. For that, we calculated the maximum image intensity from left-to-right and top-to-bottom for all CT scans, then we converted these values into number i.e., object to list. Then we fetched the staring number from both this maximas, and that were used for cropping each CT scans dynamically. In figure 6, we have plotted the histograms for both open and closed lungs. it can be clearly observed that in case of an open lungs two maximas are spotted, both from left-to-right and top-to-bottom. whereas in case of a closed lung no such maximas are visible.

Next, we created masks of dark pixels (zero values), by considering the first and the last maxima observed for CT scan starting from left-to right, similarly we included the first and the last maxima observed for CT scan starting from top-to bottom and based on these values images were correctly cropped. Overall, we felt that this tracing of location (in terms of pixel) for each maxima was extremely crucial to start and end the cropping process.figure 7 shows the overall cropping process of the open lungs.

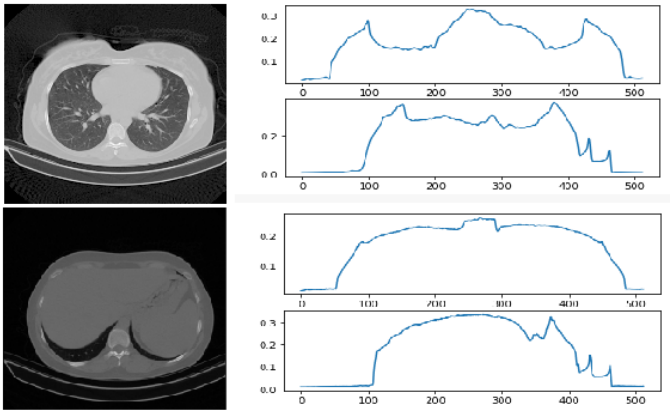

FIGURE 6. top-to-bottom histograms of open and closed lungs (row and column wise).

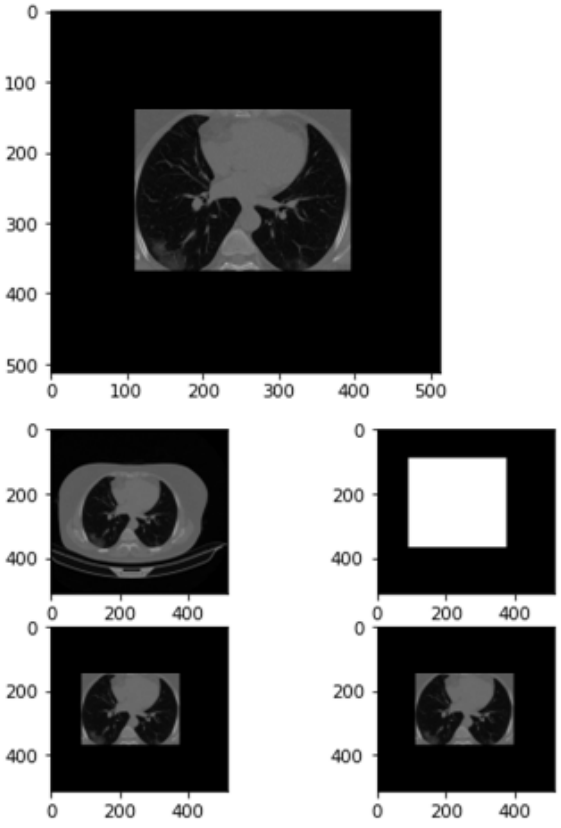

FIGURE 7. Creating masks to crop the open lung

4) Centralize the Cropped Lungs

After cropping the images, we observed that some of the lungs were not properly centralized and hence we need to perform some more operations on it. The reason why this was important was because if all the lungs were not centralized then it would have been a little bit harder for the network to process, but if all the lungs were centralized then the overall task becomes extremely simple for the network since we would then be comparing the pixels that are exactly or mostly on the same position. Likewise, If, this was not done, then the network would have to perform two tasks, the first being the localization of the exact lung region and the second being whether the lung has the SARS-COV 2 or not. For doing this, we saw that the maxima's from the generated histograms divided the left and the right portion of the lungs.

Here we defined the mid maxima for the gap duration between the left and right lung so that we could trace its location and shift the image to centrally align the lungs on each side. Hence, we traced that part and calculated the pixel location of consecutive maxima from left to right. Finally, the shift was made in all the cropped CT scans when we considered the value of X axis *[(w//2)- maximal2r[1]]* i.e., by dividing the width of the image by 2 and subtracting the obtained maxima. Figure 8 shows the centralized cropped lungs.

5) Balancing the Dataset

Since we originally had 63849 CT scan images, which had 48260 images of normal scans and 15589 images of covid-19 scans. It was getting computationally difficult for us to work with the entire dataset and especially because the classes were imbalanced. Hence for having 1:1 ratio of both the classes, we decided to down sample the majority class (normal). Firstly, we identified all the open lung images (using segregation algorithm) from the Covid-19 class, and which came to be about 8774, and we got around 31226 open lung images from the normal patient class. Out of this we randomly selected 8774 images to balance the dataset with a total of 17548 CT scan images.

6) Data Augmentation

Throughout training our network, we incorporated multiple data augmentation techniques because we firmly believe that these approaches made learning more successful and avoided over-fitting of the network, It has been known to everyone that the deep neural networks are as good as the overall data that we feed to them. Hence it was extremely essential to reduce the number of irrelevant features from the data.

In this research, we decided to apply *augmentation on the fly*, particularly on our training dataset. This is because we were dealing with an already large dataset and did not want to further increase it size. Instead, we wanted to perform transformations on the mini-batches that can be fed to the model. Here we explored the Keras Image Augmentation API to perform different augmentation types. The reason we used this API was that we wanted to reduce our memory overhead, hence rather than performing the operations on our entire COVID-CTset dataset in memory, this API was only iterated during the deep CNN model fitting process, creating the augmented CT scan for us just-in-time. We first configured the ImageDataGenerator() constructor then used the fit() function to fit our data. We have used augmentation techniques as an argument to our ImageDataGenerator() constructor, including horizontal flipping (randomly flip the

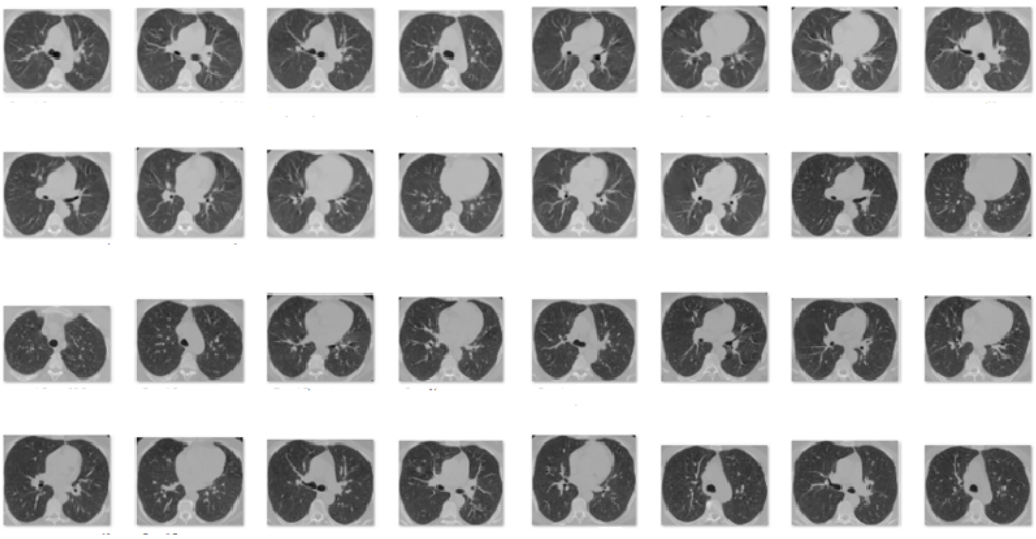

FIGURE 8. Dataset after preprocessing

images), Rotation at 30 degrees (randomly rotate images in the range (degrees, 0 to 180)), Horizontal and vertical shift both by 0.1 (this randomly shifted images both horizontally and vertically by a fraction of 0.1), Feature Standardization (to standardize pixel values across the entire dataset) this can be done by setting the featurewise_std_normalization and samplewise_std_normalization parameter to True. This basically divides the input by its standard deviation of the dataset [23].

VI. CONVOLUTIONAL NEURAL NETWORKS

Once we constructed the pre-processed dataset our objective was to accurately classify the selected CT scan images exported from the CT scan segregation algorithm into normal or COVID-19. For that we trained deep convolutional networks.Deep learning is a subset of a wider set of machine learning techniques focused on representation learning and artificial neural networks. They can be either supervised or unsupervised. In the past,CNN have been used in fields such as computer vision, machine vision, and medical image processing, with findings that were equal to, if not better than, domain experts.

In most CNN's, there is one input layer, one output layer, and one or more hidden layers in between. Any hidden layer has two connections: one to which it receives a signal and the other to which it sends the transformed signal. The output layer is connected to a previous layer, while the input layer is connected to a corresponding layer. CNN, on the other hand, is not one thing. It is a subcategory of algorithms. It also comprises a number of so-called network architectures. It can be thought of as a set of components arranged in a certain order. These elements can be replicated and combined to form blocks of components. In our work, these elements merge to form a CNN. For our research we decided to use CNNs since they are very convenient for working with 2D images. The CT scan images, were first transformed using mathematical transformations using a non-linear activation function (ReLU), and was ended in the form of a classifier. We made use of Keras to develop CNN as a sequence of layers, which provides excellent libraries like TensorFlow for training the neural networks. We made use of sequential model as it allowed us to create the model in a layer-by-layer manner. Now to give some information about its layers, they were three dimensional. This means that the neurons were structured in shape of form (width, height, depth).

Since we have CT scans with 224×224 pixels encoded as Grayscale, the shape of the input layer was (224, 224, 1). We have made use of 12 layers in our network including 2D convolution layer, MaxPooling2D layer, Flatten layer and Dense layer with 13,131,874 trainable parameters. The 2D convolution layer created a convolution kernel that helps to produce a tensor of outputs. It was mainly used to extract important features from the CT scan image. As a good practice we included Layers early in the network architecture (i.e., closer to the actual input image) learn fewer convolutional filters (32) while layers deeper in the network (i.e., closer to the output predictions) will learn more filters (128). Max pooling was then used with a pool size of (2,2) to reduce the spatial dimensions of the output volume. Next, we had a Conv2D layer with 64 filters and our final Conv2D layer had 128 filters followed by a Max pooling layer. Now as far as Kernel size is concerned (to specify the width and height of the 2D convolution window). Because we had CT scans of dimensions 224*224 we decided to take (3,3) as our kernel size. We made use of ReLU as our activation function in the Conv2D layer and softmax function was used in the output layer (since we are dealing with a binary classification problem) just to normalize it. Next to initialize all values in the Conv2D class prior to training the network we used 'he_uniform' as a kernel initializer which draws samples from a truncated normal distribution centered on 0. To avoid

any overfitting, we used L2 regularization with a value of 0.001. We kept the padding as zero i.e., 'same' because we didn't want our output in convolution to differ in size to input.

A. K-FOLD CROSS VALIDATION

Cross validation is a form of model verification that is superior to residuals. The trouble with residual assessments is that they do not show how well the learner can do when told to make new assumptions about information it has not seen before. When training a learner, one way to get around this issue was to not use the whole data collection. As a result, some of the data was eliminated before the training started. After that, the data that had been discarded was used to assess the trained model's output on new data. The model was evaluated with scikit-learn using stratified 5 and 10-fold cross validation. The model's accuracy was estimated using this resampling methodology. For 10-fold CV data was divided into 10-parts and training the model on all of them except one, which serves as a testing sample for analyzing the model's results.The average score of all built models was used as a robust approximation of success after this method was replicated 10 times.Similar process was repeated for 5-fold CV.

VII. EXPERIMENTAL CONDITIONS AND SCENARIOS

In the experiment, the results were obtained using 5 and 10-fold cross-validation technique. The proposed neural network consists of 6 convolutional layers, and 3 maxpooling layers as described in figure 1, the learning rate is 0.000001, and the maximum epoch number is 85, as determined experimentally. The CNN was implemented using Python and the Keras package with TensorFlow2 on an Intel(R) Core (TM) i7-1.5 GHz processor. In addition, the experiments were executed in Google Colab Pro, using the Graphical Processing Unit (GPU) NVIDIA-SMI 460.67, Tesla P100 with 20 GB and RAM.

VIII. RESULTS

A. PERFORMANCE METRICS

To evaluate the effectiveness of the suggested neural network, we used the following metrics: True positive (TP) denotes correctly predicted COVID-19 cases, false positive (FP) denotes normal cases misclassified as COVID-19 by the network, true negative (TN) denotes correctly identified normal cases, and false negative (FN) denotes COVID-19 cases misdiagnosed as normal cases.

$$Accuracy = (TP + TN) / (TP + FP + TN + FN)$$
$$Precision = (TP) / (TP + FP)$$
$$Recall = (TP) / (TP + FN)$$
$$F1\text{-}Score = 2 * [(Precision * Recall) / (Precision + Recall)]$$

B. NETWORK EVALUTION

Below we have shown the accuracy and loss plots for the best and the worst fold in our 5-fold and 10-fold cross validation process for 85 epochs.it can be seen that the accuracy increases with each epoch.

Firstly for 5-fold cross-validation we got the finest results for the 2nd fold where the training and validation accuracy started with 0.54 and 0.62 respectively and reached upto 0.99 and 0.94 respectively at the last epoch.The loss reduced significantly with each passing epoch. At the end we got an average accuracy of 94.07%, f1-score (for covid) of 0.95, precision and recall (for covid) of 0.93 and 0.97 respectively, and loss of 5.88.

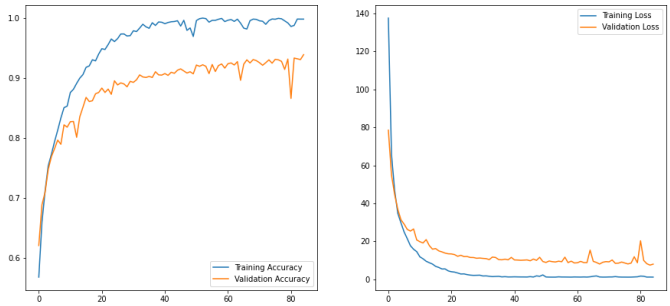

FIGURE 9. 5-fold CV: Fold 2 (Best): Accuracy and Loss (Training and Validation)

Next, we got indigent results for the 3rd fold where the training and validation accuracy started with 0.54 and 0.58 respectively and reached upto 0.99 and 0.90 respectively at the last epoch.The loss reduced significantly with each passing epoch. At the end we got an average accuracy of 90.17%, f1-score (for covid) of 0.93, precision and recall (for covid) of 0.87 and 0.99 respectively, and loss of 15.57.

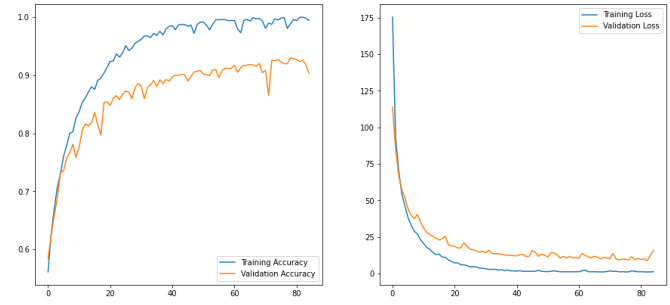

FIGURE 10. 5-fold CV: Fold 3 (Worst): Accuracy and Loss (Training and Validation)

After 5-folds we computed the average validation scores for the folds, which came to be, a accuracy of 92.86% (+-1.42) and loss of 8.30. The f1-score was 0.94 (for covid) and 0.90 (for normal), the precision was 0.92 (for covid) and 0.94 (for normal), and recall of 0.99 (for covid) and 0.87 (for normal). Table 1 and 4 shows the average classification report and fold-wise scores respectively for 5-fold CV.

In case of 10-fold cross validation we obtained the foremost results For the 2nd fold, where the training and validation accuracy started with 0.53 and 0.62 respectively and reached upto 1.00 and 0.95 respectively at the last epoch.The

	Precision	Recall	F1-Score	Support
NORMAL	0.94	0.87	0.90	1125
COVID-19	0.92	0.97	0.94	1776
accuracy			0.93	2901
macro avg	0.93	0.92	0.93	2901
weighted avg	0.93	0.93	0.93	2901

TABLE 1. Average Classification Report for 5-folds CV

	Precision	Recall	F1-Score	Support
NORMAL	0.96	0.91	0.93	1112
COVID-19	0.94	0.98	0.96	1795
accuracy			0.95	2907
macro avg	0.95	0.94	0.95	2907
weighted avg	0.95	0.95	0.95	2907

TABLE 2. Average Classification Report for 10-folds CV

loss reduced significantly with each passing epoch. At the end we got an average accuracy of 95.80%, f1-score (for covid) of 0.96, precision and recall (for covid) of 0.95 and 0.98 respectively, and loss of 5.16.

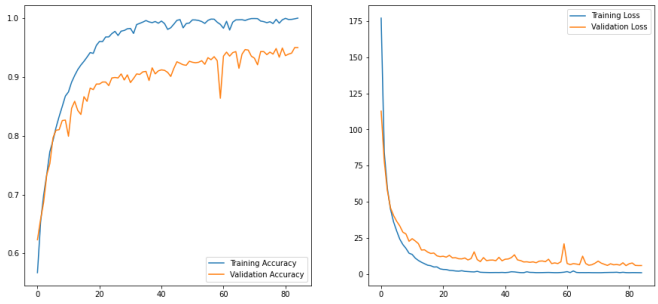

FIGURE 11. 10-fold CV: Fold 2 (Best): Accuracy and Loss (Training and Validation)

Now, when it comes to the most inferior case , for the 8th fold training and validation accuracy started with 0.58 and 0.60 respectively and reached upto 1.00 and 0.93 respectively at the last epoch.The loss reduced significantly with each passing epoch. At the end we got an average accuracy of 94.15%, f1-score (for covid) of 0.95, precision and recall (for covid) of 0.95 and 0.96 respectively, and loss of 5.56.

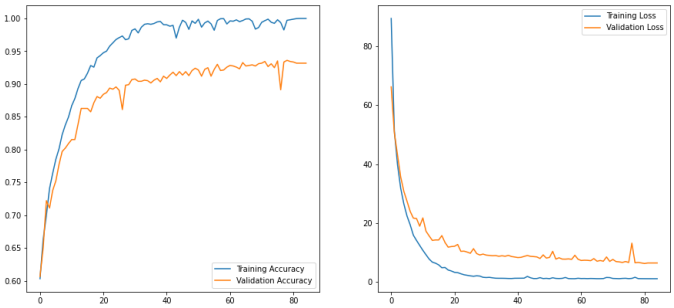

FIGURE 12. 10-fold CV: Fold 8 (Worst): Accuracy and Loss (Training and Validation)

After 10-folds we computed the average validation scores for the folds, which came to be, a accuracy of 95.02% (+-0.44) and loss of 7.24. The f1-score was 0.96 (for covid) and 0.93 (for normal), the precision was 0.94 (for covid) and 0.96 (for normal), and recall of 0.98 (for covid) and 0.91 (for normal). Table 2 and 5 shows the average classification report and fold-wise scores respectively for 10-fold CV.

Our primary aim to accurately predict the covid positive cases from all the scans was achieved when we employed 10-fold cross validation. With a recall (sensitivity) of 0.98, it could be very well interpreted that our network was successfully able to predict 8544 covid-19 positive cases, therefore only misclassifying 220 of them as normal. Moreover, we got a precision of 0.96 for normal cases which means that 8423 normal slices were correctly predicted, and only 351 being misclassified.

Training Parameters	
Parameter Name	Our Network
Learning Rate	0.000001
Batch Size	100
Epochs	85
Activation Function	ReLU
Horizontal Flipping	True
Zoom Range	0.2
Rotation Range	30 degree
Width / Height Shift	0.1
Kernel Initializer	he_uniform
Optimizer	Adam
Loss Function	SparseCategorical Crossentropy

TABLE 3. Training Parameters for 2D CNN

IX. DISCUSSION

COVID-19 has caused significant public health and safety issues and has thus become a worldwide problem. Also, when it comes to working with CNN, we claim that an accomplished radiologist would usually make decisions on the probability of patient having COVID-19; nevertheless, contextual considerations and human expertise could potentially impact those judgments. In conjunction, deep-learning based screening models show more specific and consistent findings by means of digitizing and standardising the details from the lung CT slices. It can help in identifying and isolating patients with signs of covid-19. Consequently, they can help doctors make better and precise clinical choices. Our framework is generic, which means it can be applied to a variety of classification tasks and by analyzing the results, it is demonstrated that a combination of relevant preprocessing and CNN has significant effects on the detection of COVID-19 based on the automatic extraction of features from CT scan images.

Fold	Accuracy	F1-Score(c)	F1-Score(n)	Precision(c)	Recall(c)	Precision(n)	Recall(n)	Wrong predictions(c)	Correct predictions(c)
1	0.93	0.94	0.91	0.93	0.96	0.93	0.89	350	8423
2	0.94	0.95	0.92	0.93	0.97	0.95	0.89	263	8510
3	0.90	0.93	0.86	0.87	0.99	0.98	0.76	88	8686
4	0.93	0.95	0.91	0.92	0.97	0.95	0.87	264	8510
5	0.94	0.95	0.92	0.96	0.94	0.91	0.94	526	8247
Avg	**0.93**	0.94	0.90	0.92	**0.97**	0.94	0.87	298	**8476**

TABLE 4. Results for 5-folds CV

Fold	Accuracy	F1-Score(c)	F1-Score(n)	Precision(c)	Recall(c)	Precision(n)	Recall(n)	Wrong predictions(c)	Correct predictions(c)
1	0.95	0.96	0.93	0.93	0.99	0.98	0.89	87	8687
2	0.96	0.97	0.94	0.95	0.98	0.97	0.92	175	8599
3	0.95	0.96	0.93	0.93	0.98	0.97	0.89	175	8599
4	0.95	0.96	0.93	0.95	0.97	0.95	0.91	264	8510
5	0.95	0.96	0.93	0.94	0.98	0.96	0.91	175	8599
6	0.95	0.96	0.93	0.93	0.98	0.97	0.89	175	8599
7	0.96	0.96	0.94	0.96	0.96	0.94	0.94	351	8423
8	0.94	0.95	0.92	0.95	0.96	0.93	0.91	351	8423
9	0.95	0.96	0.94	0.96	0.97	0.95	0.93	264	8510
10	0.95	0.96	0.93	0.94	0.98	0.97	0.90	175	8599
Avg	**0.95**	0.96	0.93	0.94	**0.98**	0.96	0.91	220	**8554**

TABLE 5. Results for 10-folds CV

The proposed system with 10-fold CV could distinguish COVID-19 from normal cases with high accuracy of 95% and a F1 score of 0.96 for covid CT scans. Also, we achieved a recall(sensitivity) score of 0.98. The reason we have put emphasis on the recall score is because we are dealing with a very sensitive medical issue, and our goal throughout this work has been to accurately identify patients with sars-cov2 and to minmize the chances of missing any positive cases as that would overburden the healthcare system.

A comparison between existing systems and our proposed system, in terms of average accuracy, precision, recall and total correct predictions (covid-19) , is shown in Table 6. here (n) denotes normal and (c) denotes covid-19. we compared our network with 5 other existing methods that we studied in the literature review section. it is distinctly observed that our network achieved a way better precision and recall then all the existing networks. Also, we have evaluated our system on highest number of CT scans for covid-19. The most significant aspect of our work lies in the fact that we did not utilise and deep learning strategies for our pre-processing part and that we believe has given us competitive ramifications.

X. CONCLUSION

To conclude, in this research, we have used a deep convolutional neural network for detection of COVID-19 using CT-scan images. We trained our model on one of the largest publicly available datasets, and provided a detailed experimental study, by focusing on model's performance in terms of Accuracy, F1-score, and precision-recall curve. Furthermore, we achieved a high accuracy rate when compared to already existing works and obtained more features from the dataset not manually but by automating it with end-to-end structure. In addition, for the pre-processing part, we believe it made our problem much simpler and resulted in optimal network performance. Overall we are of the opinion that our system can accurately detect COVID-19 and serve as a supplement to RT-PCR test.

XI. FUTURE WORK

We initially planned to use the transfer learning approach to validate our pre-processed dataset, but got underwhelming results for pre-trained models like ResNet50 and VGG19, we felt that this was because our images were of 2D (2 CHANNEL) grayscale color space whereas in general all the pre-trained models expect a 3D image as an input, so this is something which can be worked upon in the near future. Moreover, using the concept of convex hulls, open and closed lungs can be segregated and other machine learning architectures can also be implemented in the future.

REFERENCES

[1] W. H. Organization et al., "Laboratory testing for coronavirus disease (covid-19) in suspected human cases: interim guidance, 19 march 2020," World Health Organization, Tech. Rep., 2020.

Author	Model	Avg Accuracy	Total scans(c)	Precision(c)	Recall(c)	Wrong predictions(c)	Correct predictions(c)
Rahimzadeh et al [11]	ResNet50V2+FPN	0.98	450	0.81	0.94	23	433
Y. Song et al [15]	DeepPneumonia	0.86	777	0.79	0.96	31	746
L. Li et al [16]	COVNet	N/A	1292	N/A	0.90	130	1162
V. Shah et al [18]	CTnet-10	0.82	349	0.80	0.82	63	286
S. Hu et al [23]	NTS-NET	0.91	150	0.74	0.83	25	125
Proposed System	CNN	0.95	**8774**	**0.94**	**0.98**	220	**8554**

TABLE 6. Comparison of the proposed system with existing system

[2] W.-j. Guan, Z.-y. Ni, Y. Hu, W.-h. Liang, C.-q. Ou, J.-x. He, L. Liu, H. Shan, C.-l. Lei, D. S. Hui et al., "Clinical characteristics of coronavirus disease 2019 in china," New England journal of medicine, vol. 382, no. 18, pp. 1708–1720, 2020.

[3] Z. Wu and J. M. McGoogan, "Characteristics of and important lessons from the coronavirus disease 2019 (covid-19) outbreak in china: summary of a report of 72 314 cases from the chinese center for disease control and prevention," Jama, vol. 323, no. 13, pp. 1239–1242, 2020.

[4] J. F.-W. Chan, S. Yuan, K.-H. Kok, K. K.-W. To, H. Chu, J. Yang, F. Xing, J. Liu, C. C.-Y. Yip, R. W.-S. Poon et al., "A familial cluster of pneumonia associated with the 2019 novel coronavirus indicating person-to-person transmission: a study of a family cluster," The Lancet, vol. 395, no. 10223, pp. 514–523, 2020.

[5] J. Chan, "W, yuan s, kok k-h, et al," A familial cluster of pneumonia associated with the, pp. 30 154–9, 2019.

[6] D. S. W. Ting, L. Carin, V. Dzau, and T. Y. Wong, "Digital technology and covid-19," Nature medicine, vol. 26, no. 4, pp. 459–461, 2020.

[7] M. L. Kyoung and J. Lee, "Drive-through trend sweeps across multiple sectors, korea. net."

[8] H. Shi, X. Han, N. Jiang, Y. Cao, O. Alwalid, J. Gu, Y. Fan, and C. Zheng, "Radiological findings from 81 patients with covid-19 pneumonia in wuhan, china: a descriptive study," The Lancet Infectious Diseases, 2020.

[9] A. Narin, C. Kaya, and Z. Pamuk, "Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks," arXiv preprint arXiv:2003.10849, 2020.

[10] T. Ai, Z. Yang, H. Hou, C. Zhan, C. Chen, W. Lv, Q. Tao, Z. Sun, and L. Xia, "Correlation of chest ct and rt-pcr testing in coronavirus disease 2019 (covid-19) in china: a report of 1014 cases," Radiology, p. 200642, 2020.

[11] M. Rahimzadeh, A. Attar, and S. M. Sakhaei, "A fully automated deep learning-based network for detecting covid-19 from a new and large lung ct scan dataset," medRxiv, 2020.

[12] X. Mei, H.-C. Lee, K.-y. Diao, M. Huang, B. Lin, C. Liu, Z. Xie, Y. Ma, P. M. Robson, M. Chung et al., "Qihua long, sharon steinberger, adam jacobi, timothy deyer, marta luksza, fang liu, brent p. little, zahi a. fayad, and yang. 2020."," Artificial Intelligence-enabled Rapid Diagnosis of Patients with COVID-19." Nature Medicine.

[13] K. Zhang, X. Liu, J. Shen, Z. Li, Y. Sang, X. Wu, Y. Zha, W. Liang, C. Wang, K. Wang et al., "Clinically applicable ai system for accurate diagnosis, quantitative measurements, and prognosis of covid-19 pneumonia using computed tomography," Cell, 2020.

[14] C. Jin, W. Chen, Y. Cao, Z. Xu, X. Zhang, L. Deng, C. Zheng, J. Zhou, H. Shi, and J. Feng, "Development and evaluation of an ai system for covid-19 diagnosis," medRxiv, 2020.

[15] Y. Song, S. Zheng, L. Li, X. Zhang, X. Zhang, Z. Huang, J. Chen, H. Zhao, Y. Jie, R. Wang et al., "Deep learning enables accurate diagnosis of novel coronavirus (covid-19) with ct images," medRxiv, 2020.

[16] L. Li, L. Qin, Z. Xu, Y. Yin, X. Wang, B. Kong, J. Bai, Y. Lu, Z. Fang, Q. Song et al., "Using artificial intelligence to detect covid-19 and community-acquired pneumonia based on pulmonary ct: evaluation of the diagnostic accuracy," Radiology, vol. 296, no. 2, 2020.

[17] A. A. Ardakani, A. R. Kanafi, U. R. Acharya, N. Khadem, and A. Mohammadi, "Application of deep learning technique to manage covid-19 in routine clinical practice using ct images: Results of 10 convolutional neural networks," Computers in Biology and Medicine, p. 103795, 2020.

[18] V. Shah, R. Keniya, A. Shridharani, M. Punjabi, J. Shah, and N. Mehendale, "Diagnosis of covid-19 using ct scan images and deep learning techniques," medRxiv, 2020.

[19] J. Chen, L. Wu, J. Zhang, L. Zhang, D. Gong, Y. Zhao, S. Hu, Y. Wang, X. Hu, B. Zheng et al., "Deep learning-based model for detecting 2019 novel coronavirus pneumonia on high-resolution computed tomography: a prospective study," MedRxiv, 2020.

[20] A. M. Hasan, M. M. AL-Jawad, H. A. Jalab, H. Shaiba, R. W. Ibrahim, and A. R. AL-Shamasneh, "Classification of covid-19 coronavirus, pneumonia and healthy lungs in ct scans using q-deformed entropy and deep learning features," Entropy, vol. 22, no. 5, p. 517, 2020.

[21] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 3431–3440.

[22] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," IEEE transactions on pattern analysis and machine intelligence, vol. 39, no. 4, pp. 640–651, 2017.

[23] S. Hu, Y. Gao, Z. Niu, Y. Jiang, L. Li, X. Xiao, M. Wang, E. F. Fang, W. Menpes-Smith, J. Xia et al., "Weakly supervised deep learning for covid-19 infection detection and classification from ct images," IEEE Access, vol. 8, pp. 118 869–118 883, 2020.

[24] S. Yazdani, S. Minaee, R. Kafieh, N. Saeedizadeh, and M. Sonka, "Covid ct-net: Predicting covid-19 from chest ct images using attentional convolutional network," arXiv preprint arXiv:2009.05096, 2020.

[25] A. Ulhaq, A. Khan, D. Gomes, and M. Pau, "Computer vision for covid-19 control: A survey," arXiv preprint arXiv:2004.09420, 2020.

[26] A. Saeedi, M. Saeedi, and A. Maghsoudi, "A novel and reliable deep learning web-based tool to detect covid-19 infection form chest ct-scan," arXiv preprint arXiv:2006.14419, 2020.

ABHISHEK NAGRECHA received the bachelors degree in Computer Engineering from SVIT, Vasad, Gujarat, India in 2018. He had industrial experience as Data Analyst at Codium Techlabs, India. He did his Masters in Computer Science at Lakehead University, Canada. This research was part of his final year project at University.

RAJESH AYTHA received the bachelors degree in Computer Science from CVR college of engineering, Hyderabad, Telangana, India in 2018. He had industrial experience as Associate Software Engineer in Accenture, India. He did his Masters in Computer Science at Lakehead University, Canada. This research was part of his final year project at University.