Deep Learning based Detection of Covid 19 using CT scan lung images

Ashutosh Sahu

Department of Computer Science and Technology

Sagar Institute of Science and Technology

Gandhi Nagar Bhopal, M.P.

Nov 10, 2021

Abstract

At the end of the year 2019, the entire world witnessed a new threat never seen before. The sudden outbreak of severe acute respiratory syndrome coronavirus also referred to as SARS-CoV-2 or just the COVID-19 virus emerged within the city of Wuhan, China, surprised the planet with its rapid spread and has had a significant impact on the lives of billions of people. Coronavirus is an outsized family of viruses that cause illnesses starting from the cold to more severe diseases, including lung damage and death. In the beginning days, computerized tomography (CT) was considered an efficient tool for the detection and quantification of the disease. This research aimed to construct a system that supported deep learning for detecting COVID-19 on high-resolution CT scans converted to pictures. By using these deep learning algorithms, potentially, the method of differentiating covid patients from non-covid or pneumonic patients is often easily automated.

1. Introduction

The COVID-19 virus first started to gain attention, during December 2019, when the clinical institutions of China, started reporting about the symptoms. On January 24, 2020, the clinical characteristics of 41 patients, indicated that the common onset symptoms were fever, cough, myalgia, fatigue, headache, or gastrointestinal symptoms. These patients had pneumonia and their chest CT [1-2] image test showed abnormalities. The complications included acute respiratory distress syndrome, acute heart injury, and secondary infections. Afterward, more research showed that these symptoms are caused by a virus which was later named Coronavirus disease 2019 (COVID-19). It is an infectious disease that has caused about 50.5 lac deaths all over the world, 4.61 lac in India, among 250 million infected cases worldwide including 34 million from India, as of Nov 6, 2021.

Most of the tests currently are based on reverse transcription-polymerase chain reaction (RT-PCR) or Lateral Flow test (LFT). During the peak time of the COVID-19 outbreak (Jan - July 2020), RT-PCR test kits were in great shortage. As a result, many suspected cases cannot be tested in time and they continue to spread the disease to others, without knowing. To mitigate the shortage and accuracy of RT-PCR test kits, other alternate testing methods are emphasized. Among them, computed tomography (CT) scans have been used for screening and diagnosing COVID-19.

In the Diagnosis, *X. Yang et al.* [3], made by the National Health Commission and State Administration of Traditional Chinese Medicine in China, CT and other chest imaging techniques were found to be an acceptable way to detect COVID-19. According to the radiologists, during the outbreak time, CT is useful for diagnosing COVID-19; out of the outbreak time, CT is not as useful. The reason is that CTs can be used to judge whether a patient is infected by viral pneumonia. However, CTs cannot determine whether it is caused by COVID-19 or another such virus. So, it concludes that CTs cannot be used to confirm whether a patient is infected by COVID-19. But, during the outbreak time, most viral pneumonia is caused by COVID-19. So, it can be said, if a patient is confirmed to have viral pneumonia, then this viral pneumonia is very likely to be COVID-19. Due to this fact, CTs are considered useful for diagnosing COVID-19 during its outbreak.

The major problem with CT scans is that they require time to observe and diagnose. Mostly if a person has just started showing symptoms, it’s hard to read his CT scan and confirm the abnormalities. Due to this, the radiologists became highly occupied, and hospitals started failing to provide enough bandwidth to read several CT scans timely.

Besides, These scans require some senior/experienced radiologists, since this disease was relatively new. To address these problems, several works including *Huang et al., 2020* [4], *Li et al., 2020* [5] have developed deep learning and AI solutions to screen COVID-19 from CTs.

In this presented research, with the availability of datasets from multiple sources, we have constructed and validated a system based on deep learning for identifying Covid CT scans. Since viral pneumonia and Covid may seem similar, but they are different, we have also tried to segregate the results on this basis. The research also includes a comparison training of pretrained models on the imagenet dataset, which are further trained for this classification purpose by transfer learning techniques. The final model suggested by this study gives a comparable performance but takes much less time, hence can be utilized by radiologists to cross-verify once, which can enormously reduce the time invested in reporting CT scans. The study and source code developed in this research work is available at https://github.com/Ashuto7h/CovidScanner and API and web app reference to demonstrate the system are available at https://covidscaner.herokuapp.com.

2. Methodology

2.1 Dataset

For any deep learning research, the basic and essential requirement is a valid dataset, which is utilized for training the model. The dataset that is used for this research is collected by *maftouni et al. 2020* [6]. They curated a large lung CT scan dataset for COVID-19 from 7 public datasets [7-13].

These datasets have been publicly used in COVID-19 diagnosis literature and proven their efficiency in deep learning applications. So, the merged dataset is expected to improve the generalization ability of deep learning methods by learning from all these resources together. Additionally, they did not include images lacking clear class labels or patient information.

In total, there are 7,593 COVID-19 images from 466 patients, 6,893 normal (Non-COVID-19) images from 604 patients, and 2,618 commonly acquired pneumonia (CAP) images from 60 patients.

The average age is 51.2, 52.8, and 64.3 for COVID-19, Normal, and CAP cases, respectively. The country and gender distributions on the whole dataset are shown in the following Figure.

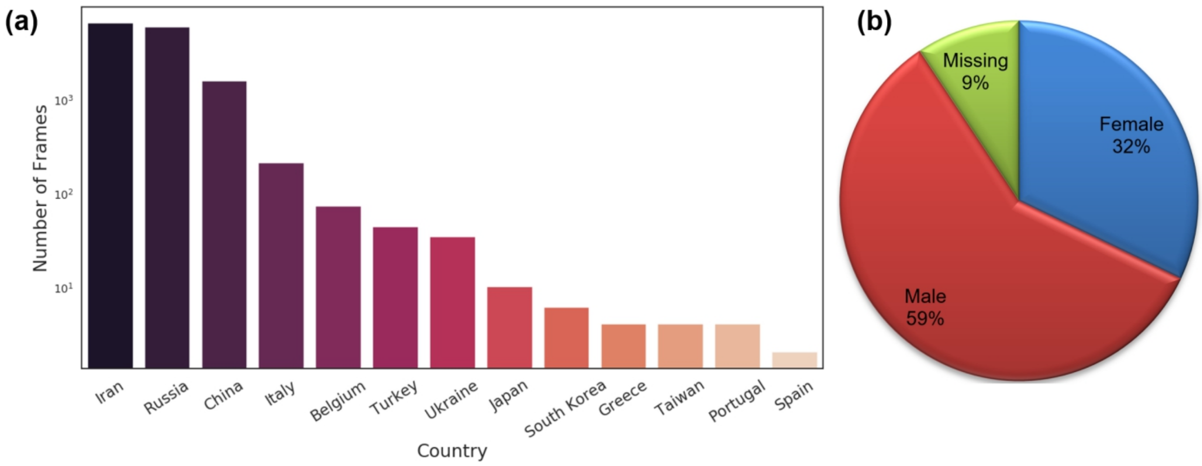

Figure 1: (a) Number of image frames per country
(b) Pie chart depicting gender ratio of dataset

The dataset is available at the following Kaggle link: Large COVID-19 CT scan slice dataset

2.1.1 Are 2D CT scan images good enough for training a deep learning model?

As mentioned by *Xingyi Yang et al.* [14], their dataset received numerous feedback expressing concerns that the COVID-CT dataset may not be useful for training CT-based diagnosis models for COVID-19 since images in COVID-CT are extracted from papers in PDF format, which are likely to have lower-quality compared with original CT scans and images in COVID-CT are single-slice (2D representations) CTs rather than multiple-slice (3D representations) CTs as in original CT scans. They further performed experimental studies to show that besides the extraction methods, their dataset proved to perform very well by providing an acceptable accuracy over comparatively less training time.

2.2 Training and Implementation

For the training purpose, the images were imported by resizing them to 256 x 256 pixels. Using the Keras API, a custom data generator is implemented, which was used to generate batches by random sampling images. The data is first split up into 3 parts, train split (80%), validation split (10%), and test split (10%). The train split was then trained using different models configurations, along with the validation split to validate the training. Test split was used for testing the final model outcome.

Using the experimental logs provided by Weights and Bias [15], the following factors were analyzed and respective configurations were applied:

- ***Data imbalance*** - The dataset is highly imbalanced. The number of Covid samples is greater than Non-Covid and common pneumonia samples are very less in number. To overcome that, balanced class weights are generated and used while training the model.

- ***Grayscale vs RGB*** - The color mode of images is an important factor in training. RGB (red-green-blue) images in such cases can lead to misinterpretations. During the initial training steps, the results were compared based on grayscale and RGB color modes of images. Grayscale images were fast to train custom models since they consist of a single channel, but for transfer learning, we required 3 channel images. We used the stacking method to create a 3 channel image out of grayscales. RGB images provided better results during training and testing both.

- ***Lung masking*** - Lung segmentation or masking highlights only the lung part while removing all other features from images. They help the model in paying attention to the lung region. While lung masks narrow down the search space, they are still coarse-grained. Within the lung regions, areas exactly containing anomalies are very small and can be neglected due to larger undamaged portions of the lungs. Furthermore, while applying lung masks, some dark contrast images appeared mostly dark and light contrast images appeared pure white. Masking was dropped from experiments due to these wrong outcomes.

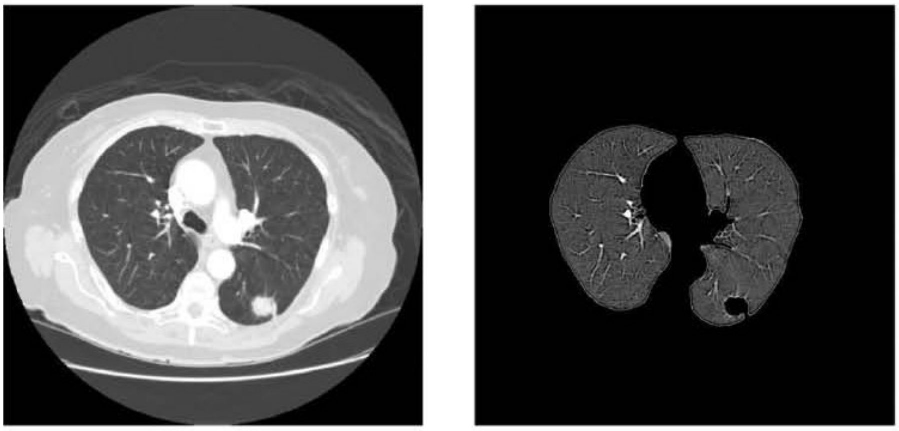

Figure 2: Original CT image (left), Lung Segmented mask (right)

- ***Augmentation*** **-** Image augmentation is the method of artificially increasing the size of the dataset. This process is used mainly to represent various variations in the images while training the model, to get more accurate results on the test dataset. The dataset used in this paper already has variations due to multiple sources. Still, augmentations like random brightness factor 0.2, horizontal flip, width, and height shifting are provided. All CTs have the same orientation, so providing a rotational augmentation will only ruin the classification.

- ***Flatten vs GlobalAvgPooling2D layers*** **-** When using Flatten layer at the top of all convolution layers, the accuracy on the validation set was as low as 0.53. After replacing the Flatten layer with GlobalAvgPooling2D and applying BatchNormalization after each dropout, the validation accuracy of the same models increased to $\simeq$ 0.85.

- ***Dropout ratio*** **-** performance of ResNet50 and Xception showed slight positive variations with dropout ratio up to 0.35 but started decreasing after that.

The training was performed by implementing transfer learning on models ResNet50, Xception, Mobilenet, and Densenet121 pretrained on weights from the imagenet dataset.

Using Tensorflow and Keras API, a customized model architecture was also implemented. The model consists of an input layer followed by a Convolution2D layer, then 3 groups of layers (MaxPooling2D, SeparableConv2D, SeparableConv2D, BatchNormalization). The First convolution layer has 32 filters of size 3x3, with step size 1, padding same, and activation function RELU. The filters were doubled in each consecutive group after that. At the top of this, a GlobalAvgPooling2D layer, 2 groups of fully connected layers (Dense, Dropout, BatchNormalization) followed by an output dense layer with activation - softmax are provided. The same layers are used as top layers for transfer learning.

Table 1: Custom model architecture and layer description

Name	Type	Parameters	Output Shape
input_1	InputLayer	0	,256,256,3
conv2d	Conv2D	896	None, 256, 256, 32
max_pooling2d	MaxPooling2D	0	None, 128, 128, 32
separable_conv2d	SeparableConv2D	2400	None, 128, 128, 64
separable_conv2d_1	SeparableConv2D	4736	None, 128, 128, 64
batch_normalization	BatchNormalization	256	None, 128, 128, 64
max_pooling2d_1	MaxPooling2D	0	None, 64, 64, 64
separable_conv2d_2	SeparableConv2D	8896	None, 64, 64, 128
separable_conv2d_3	SeparableConv2D	17664	None, 64, 64, 128
batch_normalization_1	BatchNormalization	512	None, 64, 64, 128
max_pooling2d_2	MaxPooling2D	0	None, 32, 32, 128
separable_conv2d_4	SeparableConv2D	34176	None, 32, 32, 256
separable_conv2d_5	SeparableConv2D	68096	None, 32, 32, 256
batch_normalization_2	BatchNormalization	1024	None, 32, 32, 256
max_pooling2d_3	MaxPooling2D	0	None, 16, 16, 256
separable_conv2d_6	SeparableConv2D	133888	None, 16, 16, 512

separable_conv2d_7	SeparableConv2D	267264	None, 16, 16, 512
batch_normalization_3	BatchNormalization	2048	None, 16, 16, 512
max_pooling2d_4	MaxPooling2D	0	None, 8, 8, 512
global_average_pooling2d	GlobalAveragePooling2D	0	None, 512
dense	Dense	65664	None, 128
dropout	Dropout	0	None, 128
batch_normalization_4	BatchNormalization	512	None, 128
dense_1	Dense	8256	None, 64
dropout_1	Dropout	0	None, 64
batch_normalization_5	BatchNormalization	256	None, 64
dense_2	Dense	195	None, 3

Total params: 616,739
Trainable params: 614,435
Non-trainable params: 2,304

All models were trained initially with optimizer Adam along with a learning rate scheduler keeping learning rate 0.015, decay steps 1,00,000, decay rate 0.96. Categorical CrossEntropy was used as loss and the base number of epochs was 100, an early stopping callback based on metric loss with patience 10 was applied to stop the training if losses didn't decrease further. The training was performed over Kaggle systems having OS Linux-5.10.68+-x86_64, Python version 3.7.10, 2 CPU, and 1 GPU - Tesla P100-PCIE-16GB.

The customized model was further hyper tuned using random hyperparameter search provided by Weight and Bias sweep run API. Keeping validation accuracy as a metric, batch size up to 32 resulted in increasing validation accuracy then decreased it gradually. Decay Rate and decay steps provided maximum accuracy at max limits. The dropout ratio between 0.245 to 0.381 provided high accuracy while the learning rate as low as 0.0018 provided maximum accuracy.

The best configuration obtained by this search has learning rate of 0.001886, batch size 32, decay rate 0.936, decay steps 99855, and dropout 0.2455. Rest other parameters are the same.

3. Results

The Following table shows the performance comparison of various models

Table 3.1 - Results of trained models

Model	Runtime	Training Accuracy	Validation Accuracy	Test Accuracy	Precision	Recall	F1 score
DenseNet 121	45m	90.74%	16.04%	41.69%	62%	42%	26%
MobileNet V1	48m	97.88%	46.23%	58.25%	51%	58%	53%
ResNet 50 V1	1h 23m	94.95%	78.13%	97.64%	98%	98%	98%
Xception	2h 10m	97.80%	63.33%	97.76%	98%	98%	98%
Customized Model	1h 49m	**98.67%**	**97.05%**	**97.64%**	98%	98%	98%
Customized Model Hypertuned	1h 28m	**98.65%**	**95.82%**	**97.79%**	98%	98%	98%

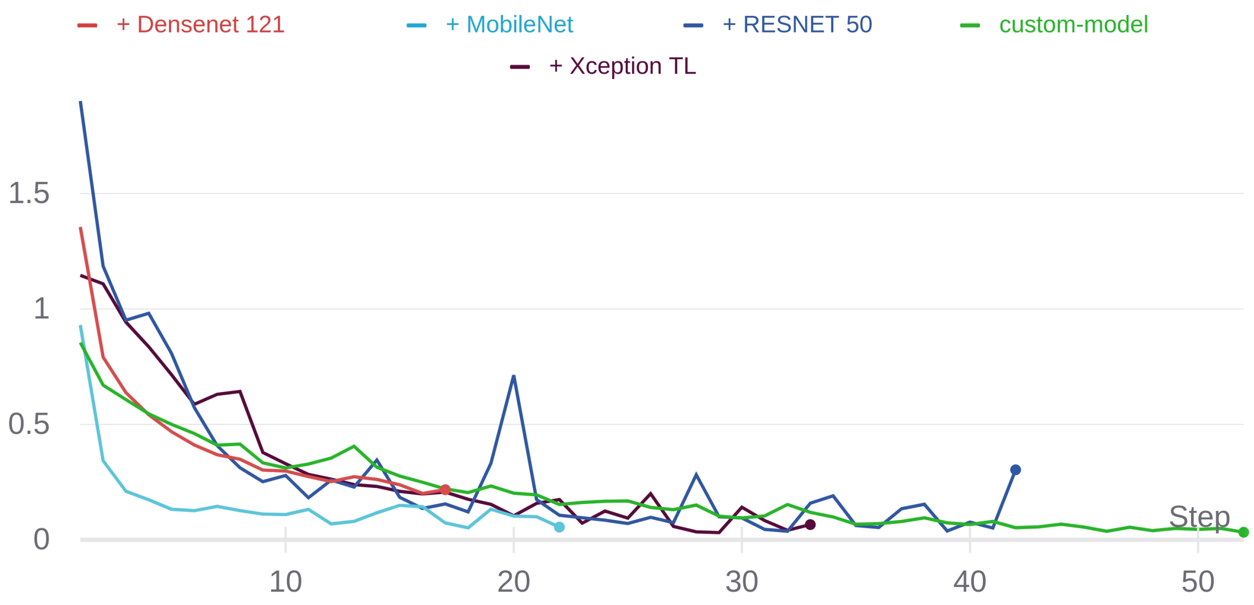

Figure 3: Training Loss vs number of Epochs for all models

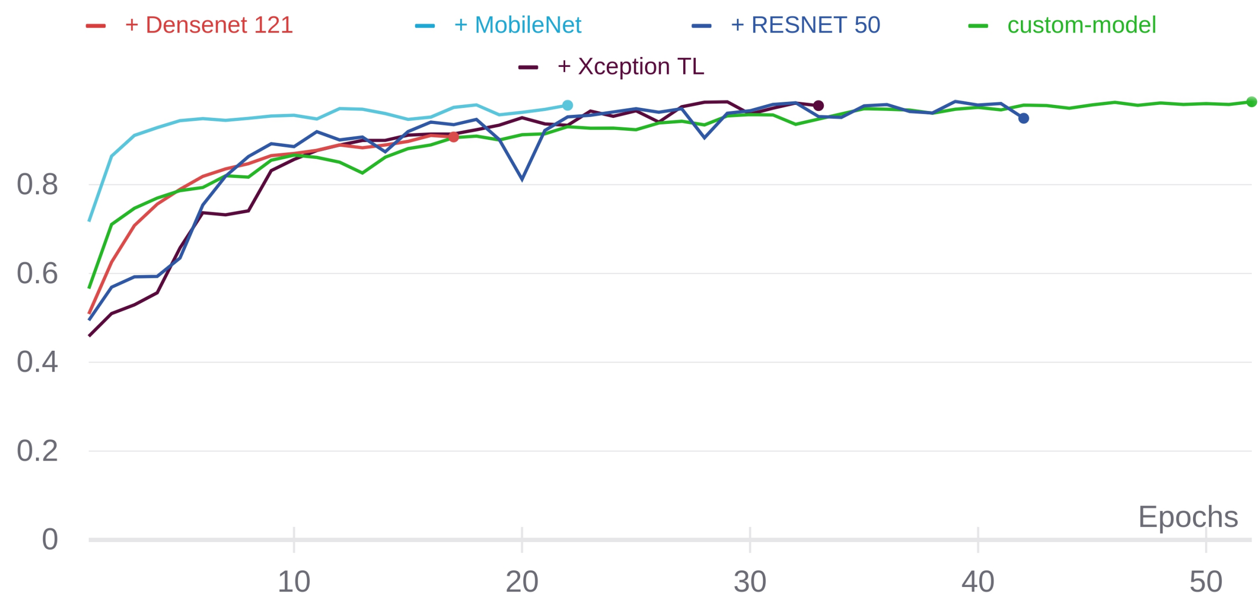

Figure 4: Training accuracy vs number of Epochs for all models

As seen from the above results, the custom model is receiving ~ 97.6% accuracy on the test split. The model is able to accurately identify 98% of non-Covid image labels, 97% of Covid image labels, and 100% of common pneumonia labels. Even considering covid and pneumonia both as Covid, it accurately predicts 97.8% of labels as Covid.

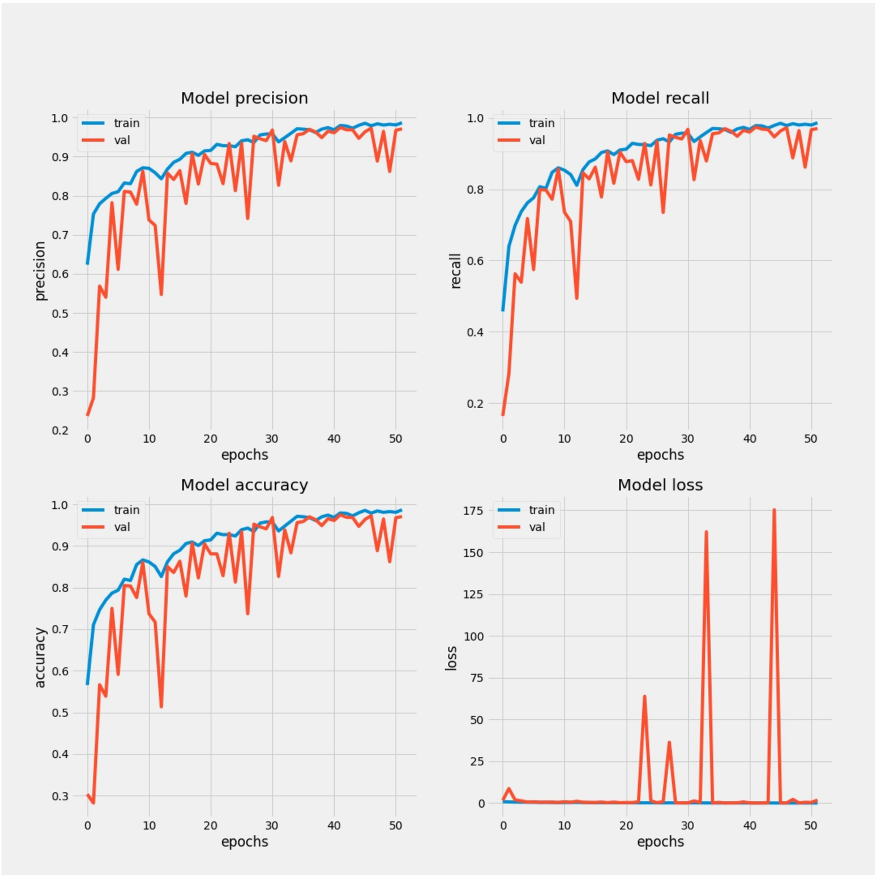

Figure 5: Custom CNN model training history plots

Predicted Labels

Actual Labels	Non-Covid	Covid	Common Pneumonia
Non-Covid	686	21	0
Covid	17	718	0
Common Pneumonia	0	2	252

Figure 6: Confusion matrix of Custom CNN model

As shown in the confusion metric, The model predicts 17 Covid images as non-Covid and 21 as non-Covid images as covid. Considering the numbers of True positive and True negative, these values are small and can be ignored.

4. Conclusion

The model gives an acceptable accuracy in comparison with other models. So, It can be used for real-time prediction of Covid-19. A web application and API demonstrating this model are available at https://covidscaner.herokuapp.com. Experiment and logs are available at https://wandb.ai/ashuto7h/covid19. The web app can be used by a radiologist to upload CT scans and get predictions. Though human verification is required just to confirm the results, still it reduces the time taken in deeply analyzing all the CTs.

Future plans include the following areas to work on -

- Applying transfer learning on the custom model to train more different datasets.
- Trying with different resolution and image sizes,
- Hyper Parameter tuning on Resnet50 and Xception models,
- Including EfficientNet and Vgg16 in experiments.

This research work shows that technological advancements can replace humans to some extent but not completely. Deep learning methods have provided solutions for many big problems reducing manpower and human resources in the past. This paper is just another example. Since the Covid-19 has arisen, there have been many efforts on deep learning methods to perform Covid-19 screening. Many other research works related to the same problem have been provided in the past, including 2D and 3D CNN models, but the dataset and other important materials of studies have been kept private in most of them.

The whole world fought Covid with whatever it could. Since the outbreak, the conditions have become much worse. Many people were affected and suffered from it, many of them lost themselves or someone close to them. It spread faster and became hard to control. With help of actions like curfew, lockdowns, vaccination the conditions have improved very much over the past year. Medical helps and testing kits are easily available today and have become the best methods to test Covid-19. Probably there will be no requirement for CT scans in not too far future.

References -

[1] Covid CT Scans viewer https://v.raioss.rocks/viewer?url=https://amazonraioss.s3.amazonaws.com/public/coronacases_01.raioss [Accessed 8 November 2020]

[2] https://coronacases.org/ [Accessed 8 November 2020]

[3] X. Yang, X. He, J. Zhao, Y. Zhang, S. Zhang & P. Xie (2020). COVID-CT-Dataset: A CT Image Dataset about COVID-19 and Treatment Protocol for Novel Coronavirus Pneumonia.

[4] 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. *Radiology: Cardiothoracic Imaging*, *2*(2), e200075. https://doi.org/10.1148/ryct.2020200075

[5] Li, L., Qin, L., Xu, Z., Yin, Y., Wang, X., Kong, B., Bai, J., Lu, Y., Fang, Z., Song, Q., Cao, K., Liu, D., Wang, G., Xu, Q., Fang, X., Zhang, S., Xia, J., & Xia, J. (2020). Using Artificial Intelligence to Detect COVID-19 and Community-acquired Pneumonia Based on Pulmonary CT: Evaluation of the Diagnostic Accuracy. *Radiology*, *296*(2), E65–E71. https://doi.org/10.1148/radiol.2020200905

[6] Maftouni, M., Law, A.C, Shen, B., Zhou, Y., Yazdi, N., and Kong, Z.J. "A Robust Ensemble-Deep Learning Model for COVID-19 Diagnosis based on an Integrated CT Scan Images Database," Proceedings of the 2021 Industrial and Systems Engineering Conference, Virtual Conference, May 22-25, 2021.

[7] J. Zhao, Y. Zhang, X. He, and P. Xie, "COVID-CT-Dataset: a CT scan dataset about COVID-19," arXiv preprint arXiv:2003.13865, 2020.

[8] P. Afshar et al., "COVID-CT-MD: COVID-19 Computed Tomography (CT) Scan Dataset Applicable in Machine Learning and Deep Learning," arXiv preprint arXiv:2009.14623, 2020.

[9] J. P. Cohen, P. Morrison, L. Dao, K. Roth, T. Q. Duong, and M. Ghassemi, "COVID-19 image data collection: Prospective predictions are the future," arXiv preprint arXiv:2006.11988, 2020.

[10] S. Morozov et al., "MosMedData: Chest CT Scans With COVID-19 Related Findings Dataset," arXiv preprint arXiv:2005.06465, 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] M. Jun et al., "COVID-19 CT Lung and Infection Segmentation Dataset," Zenodo, Apr, vol. 20, 2020.

[13] "COVID-19." 2020. http://medicalsegmentation.com/covid19/ [Accessed 8 November, 2020].

[14] X. Yang, X. He, J. Zhao, Y. Zhang, S. Zhang, P. Xie, "COVID-CT-Dataset: A CT Scan Dataset about COVID-19", arXiv preprint arXiv:2003.13865v3, 2020

[15] MLOps platform Weights and biases - https://wandb.ai/site [Accessed 8 November 2020]

[16] Jin, Jonghoon & Dundar, Aysegul & Culurciello, Eugenio. “Flattened Convolutional Neural Networks for Feedforward Acceleration”, 2014

[16] K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition", arXiv preprint arXiv:1512.03385, 2015.

[17] F. Chollet,"Xception: Deep Learning with Depthwise Separable Convolutions", arXiv preprint arXiv:1610.02357v3, 2016

[18] J. Jin, A. Dundar, E. Culurciello “Flattened Convolutional Neural Networks for Feedforward Acceleration“, arXiv preprint arXiv:1412.5474v4, 2014

[19] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications” arXiv preprint arXiv:1704.04861v1, 2017

[20] Gao Huang, Zhuang Liu, Laurens van der Maaten, Kilian Q. Weinberger, “Densely Connected Convolutional Networks”, arXiv preprint arXiv:1608.06993v5, 2018