




Deep neural network to detect COVID-19: one architecture for both CT Scans and Chest X-rays

Himadri Mukherjee¹ · Subhankar Ghosh² · Ankita Dhar¹ · Sk Md Obaidullah³ · K. C. Santosh⁴  · Kaushik Roy¹

Accepted: 11 September 2020 / Published online: 6 November 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

Since December 2019, the novel COVID-19's spread rate is exponential, and AI-driven tools are used to prevent further spreading [1]. They can help predict, screen, and diagnose COVID-19 positive cases. Within this scope, imaging with Computed Tomography (CT) scans and Chest X-rays (CXRs) are widely used in mass triage situations. In the literature, AI-driven tools are limited to one data type either CT scan or CXR to detect COVID-19 positive cases. Integrating multiple data types could possibly provide more information in detecting anomaly patterns due to COVID-19. Therefore, in this paper, we engineered a Convolutional Neural Network (CNN) -tailored Deep Neural Network (DNN) that can collectively train/test both CT scans and CXRs. In our experiments, we achieved an overall accuracy of 96.28% (AUC = 0.9808 and false negative rate = 0.0208). Further, major existing DNNs provided coherent results while integrating CT scans and CXRs to detect COVID-19 positive cases.

Keywords COVID-19 · Chest X-Ray · Computing tomography scan · Convolutional neural network · Deep neural network

1 Introduction

The recent outbreak of COVID-19 impacts public health across the World. In Wuhan province of China [2], it was first reported in December 2019. As of now, COVID-19 affected more than 5,204,508 people across the world with more than 337,687 deaths death cases [3], Report # 125, dated May 24, 2020. Compared to other well known coronavirus disease, such as Severe Acute Respiratory Syndrome (SARS) and Middle East Respiratory Syndrome (MERS) [4, 5], COVID-19 is found to be highly infectious and contagious. COVID-19's spread rate is exponential, and its mortality rate has not been determined yet. It is therefore required to prevent from further spreading. AI-driven tool can be of a great use in mass triage situations, where imaging techniques are more useful in

understanding disease-related pathology. To detect COVID-19 positive cases, multiple image modalities can be used, such as Computed Tomography (CT) scans and Chest X-rays (CXRs) [6] as they provide consistent COVID-19 manifestations [7]. Therefore, referring to the reported recent study, where the need of multimodal data has been addressed [1]. in this paper, our goal is to observe whether two different modalities (radiological image data) can be trained/tested using one deep neural network. For this, we engineered a Convolutional Neural Network (CNN) -tailored Deep Neural Network (DNN) that can collectively train/test both CT scans and CXRs.

2 Related works: COVID-19 detection using CT scans and CXRs

In January 2020, Huang C et al. reported clinical and para-clinical aspects of COVID-19. Where they stated that abnormalities (using 41 positive cases), such as Ground-Glass Opacity (GGO) can be observed using chest CT scans [8]. CT scans are widely used to identify unusual patterns in confirmed cases of COVID-19 [9–11]. To be precise, Li and Xia [12] experimented on 51 CT scans (images) and in 96.1% cases COVID-19 was successfully detected. Zhou S et al. [13] experimented on 62 COVID-19 and Pneumonia, and

This article belongs to the Topical Collection: *Artificial Intelligence Applications for COVID-19, Detection, Control, Prediction, and Diagnosis*

Himadri Mukherjee and K. C. Santosh contributed equally.

✉ K. C. Santosh
santosh.kc@ieee.org

Extended author information available on the last page of the article.

their results showed diverse patterns that are visually like lung parenchyma and the interstitial diseases. Also, Zheng Ye et al. [12] stated that typical and atypical CT manifestations help and familiarize radiologists in decision-making. In their study, GGOs, amalgamation, reticular marking, and crazy surfacing mark are typical CT indication of COVID-19. Emerging atypical CT indication that includes changes in airway, pleural, nodules, and fibrosis were demonstrated among COVID-19 patients. Fang et al. [11] also reported that COVID is possibly better diagnosed using radiological imaging. In their experiments, 81 patients were used, where for 30 patients initial CT was not performed within 3 days of Reverse Transcription Polymerase Chain Reaction (RT-PCR) and 51 patients were with both initial CT and RT-PCR. Chung et al. [14] reported that among 57% patients the CT findings were mainly GGO whereas among 33% cases peripheral distribution was found. Among 14% of the patients normal chest CTs were reported. Song et al. [15], described predominantly GGOs opacities in 77%, but a larger amount of peripheral distribution in 86% cases and lower lobe involvement in 90% scenario. Furthermore, the author reports that there was notable improvement during follow up chest CT images in 54% cases, in contrast with imaging progression found in 31% of the cases. Feng Pan et al. [16] experimented on 21 patients, where their average age was ranging between 25 years to 63 years with confirmed COVID-19 pneumonia presence. In their study, authors concluded that in the case of patients who recovered from COVID-19 without high difficulty in respiration, most severe abnormalities were observed in the chest CT images (lungs) after 10 days since first symptom.

Wang et al. [17] presented a deep learning-based approach to detect COVID-19 cases from CT images. Experiments were performed on a dataset of 453 COVID-19 positive cases. They reported an accuracy of 82.9%. Further they reported an accuracy of 73.1% with specificity and sensitivity values of 67% and 74% on an external test dataset. Butt et al. [18] studied different CNN models for identifying COVID-19 from CT images. They tested both 2D and 3D CNNs and reported a AUC of 0.996. Further they calculated a sensitivity of 98.2% and a specificity of 92.2%. Their experiments were performed on a dataset of 219 COVID-19 CTs.

Like CT scans, CXRs are widely used to detect COVID-19 positive cases [19–22]. Chen et al. [23] employed 99 cases, where bilateral pneumonia were observed in the CXRs of those patients. Interestingly, Soon et al. [19] observed the relationship between CXRs and CT images, where 9 COVID-19 positive cases were used. Besides, others were focused on the use of neural network-tailored Deep Learning (DL) models, such as COVID-Net [20] and ResNet50 [21]. COVID-Net was tested only on 31 COVID-19 positive cases, while ResNet50 was tested on 25 COVID-19 positive cases. Zhang et al. [22] used classical DL model

to detect COVID-19 positive cases, where 100 COVID-19 samples were used. As of now, the highest accuracy of 96% was reported to detect COVID-19 positive cases [22].

Ozturk et al. [24] presented a deep neural network-based approach to COVID-19 positive cases using CXRs (125 COVID-19 positive cases) and reported the highest accuracy of 98.08%. Further, they experimented with multi-class scenario and reported an accuracy of 87.02%. Narin et al. [25] used DNNs to detect COVID-19 from 50 COVID-19 positive cases, and reported accuracies of 97%, 98% and 87% from three DNN architectures: InceptionV3, ResNet50, and Inception-ResNetV2, respectively.

Mangal et al. [26] presented a deep learning-based system named CovidAID to detect COVID-19 cases from CXRs, where an accuracy of 90.5% with a sensitivity of 100% were reported on dataset size of 155 COVID-19 positive cases.

Wang et al. [27] presented a deep CNN named COVID-Net to detect COVID-19 cases from 256 CXRs. They worked with normal and both COVID-19 negative and positive images. They reported sensitivity values 95%, 94% and 91% for the aforementioned types, respectively. However, in the case of normal images, a higher sensitivity of 98% was reported using VGG-19.

Zheng et al. [28] presented a deep learning-based approach for detecting COVID-19 cases from CT images with weak label. Their system was trained with 499 volumes and tested with another 131 volumes and sensitivity, specificity values of 0.907 and 0.911 respectively were reported. Farooq and Hafeez [29] presented a deep learning-based approach for distinguishing COVID-19 cases from CXRs. They reported an accuracy of 96.23% with 41 epochs using ResNet50, where 8 COVID-19 positive cases were used. Hall et al. [30] used a deep learning-based approach for identifying COVID-19 cases from CXRs. In their experiments, an overall accuracy of 89.2% was reported with a true positive rate of 0.8039 along with an AUC of 0.95 from 135 COVID-19 positive cases. Further, an ensemble-based approach was used on test set of 33 CXRs and an accuracy of 91.24% was reported along with a true positive rate and AUC of 0.7879 and 0.94, respectively. Salman et al. [31] used a CNN-based approach to detect COVID-19 cases. On 130 COVID-19 positive cases, an accuracy of 100% was reported.

Zahangir et al. [32] presented a multi-task deep learning-based technique to detect COVID-19 positive cases. They experimented on both CT scans and CXRs and reported accuracies of 84.67% and 98.78%, respectively. Not to be confused, they used separate architectures for CT scans and CXRs.

From AI-driven tool perspective, where imaging techniques are used, COVID-19 does not have rich state-of-the-art literature. Most of the existing tools were designed to train on single data type, either X-rays or CT scans. Further,

the reported works present separate models for handling each data type. As integrating multiple modalities (data types) can provide more information [1]; can we design an architecture that can handle multiple modalities of data? Inspired by computer vision and pattern recognition techniques, where several different classes/objects are used to train on an exact same architecture, in this paper, we proposed a single architecture for CT scans and CXRs to detect COVID-19 positive cases. The proposed architecture was tested on publicly available datasets comprising of both data types: CT scans and CXRs.

The rest of the paper is organized as follows. The proposed DNN is explained in Section 3. We detail experimental setup in Section 4. We provide results and analyze them in Section 5. We also provide comparison study in Section 6, and the paper concludes in Section 7.

3 Proposed deep neural network

Deep learning-based algorithms [33, 34] are the variant of machine learning algorithms that work with networks of multiple layers. The multiple layers are responsible for gradually deriving distinguished features from the given input, which are then passed on to classification stage.

Convolutional neural networks (CNNs) [35] are typically composed of three main facets namely convolution layer, pooling layer, and dense layer. A CNN consists of multiple convolution and pooling layers that are followed by dense layers. The convolution and pooling layers can be arranged in several different ways; and their arrangements are conventionally based on the complexity of the problem. The typical final dense layer’s (output layer) dimension is equal to the number of output classes.

The convolutional layer detects different features from the input. This layer consists of a set of convolutional kernels. The convolutional kernels split the image into smaller chunks that aids in extracting feature patterns or maps. The kernels convolute with the images based on certain weights by multiplying its instances with the corresponding instances of the particular domain. Simply, the functioning of this layer can be expressed as,

$$f_c^k(m, n) = \sum_d \sum_{r,s} j_d(r, s) \cdot i_c^k(v, w), \tag{1}$$

where $j_d(r, s)$ is an instance of the input vector J_d , which is multiplied by $i_c^k(v, w)$ index of the k^{th} kernel of the c^{th} layer. The output mapping of the k^{th} kernel can be measured as,

$$F_c^k = [f_c^k(1, 1), \dots, f_c^k(m, n), \dots, f_c^k(M, N)]. \tag{2}$$

The convolutional layer has the capability of distributing weights that helps in extracting different features with the same set of weights by sliding the kernel making

CNN parameters effective compared to the fully connected architectures.

The pooling layer is arranged between two convolutional layers, where it gets the feature vectors and performs the pooling operation to each of the vectors. It reduces the size of the vectors while keeping their relevancy intact. It aggregates the related information in the region of the receptive domain and outputs the feedback within that region. This layer decreases the number of parameters and computations, which enhances the effectiveness of the architecture and avoids over-fitting: $Y_c^k = O_p(F_c^k)$, where Y_c^k determine the pooled feature map of the c^{th} layer for k^{th} kernel and O_p determines the kind of pooling operation.

A dropout layer is also used before the dense layers, which randomly discards neurons. It helps in preventing from being overfitting. This layer aids towards learning of robust features. Even though it increases the number of iterations for convergence but the training time for each epoch is brought down [36].

The generation of the final output equivalent to the number of categories requires an application of a fully connected layer. This layer accepts the input from the previous stages and globally evaluates the output of all the former layers. Therefore, it makes a non-linear combination of specified features that are used for the classification purpose. The output layer involves loss function: cross-entropy to calculate the error in prediction. Once the forward pass is completed, the back-propagation updates the weight and biases for reducing the error and loss.

In the present experiment, a network consisting of 3 alternating convolution and pooling layers were used. The convolution layers were of 32, 16, and 8 dimensions with filter sizes of 5×5, 4×4 and 3×3, respectively. The convolution layers used Rectified Linear Unit (ReLU) activation function: $f(x) = \max(0, x)$, where x is the input to a neuron.

The pooling window used a 2×2 filter size and a maxpooling function. The stride value in both the convolution and pooling layers were set to 1. This was followed by a dropout layer whose output was passed through 2 dense layers of 256 and 50 dimensions with ReLU activation. The output was finally passed on to a 2 dimensional output layer with softmax activation, $\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}$, where z is an input vector of length K .

The proposed network is illustrated in Fig. 1. In short, the proposed network is limited to nine layers. The numbers of layers in the network along with the different parameters were set based on experimental trials i.e., empirically designed (cf. in Table 3 in Section 5). Following the input image size, dimensions and numbers of parameters vary in our architecture. The parameters were tested for different values whose results are presented in Table 3.

Following Table 3, since $100 \times 100 \times 3$ were chosen (based on the performance), the dimension and number of parameters (for convolutional layer) provided in Table 1 can be explained as follows:

$$D_{conv} = (H_i - H_f + 1) \times (W_i - W_f + 1) \times N_f \text{ and}$$

$$P_{conv} = ((H_f \times W_f \times N_{fp}) + 1) \times N_f,$$

where

- H_i = Height of input image in present layer;
- W_i = Width of input image in present layer;
- H_f = Height of convolution filter in present layer;
- W_f = Width of convolution filter in present layer; and
- N_f = Number of filters in the present layer; and N_{fp} = Number of filters in previous layer.

Using these aforementioned equations, for the convolutional layer 1, we have $D_{conv} = (100 - 5 + 1) \times (100 - 5 + 1) \times 32 = 96 \times 96 \times 32$ and $P_{conv} = ((5 \times 5 \times 3) + 1) \times 32 = 2432$, where convolution filter sizes were empirically designed (as mentioned earlier). We followed exact similar procedure for other convolutional layers. In case of dense layer, the number of parameters can be expressed as,

$$P_{dense} = O_s \times (I_s + 1),$$

where O_s = output size and I_s = input size. With this equation, for dense layer 1, we have $P_{conv} = (256 \times (800 + 1)) = 205056$. We followed exactly similar procedure for other dense layers.

On the whole, we performed a series of tests and selected the one that gave us the best performance. For detailed tests, we refer to Table 3. The number of generated parameters for the proposed network is presented in Table 1. It is 0.009, 0.004, and 0.031 times the number of parameters used in InceptionV3, ResNet, and MobileNet, respectively (cf. Table 8 in Section 6). Moreover, after every convolution the results are pooled, which ensures computational complexity reduction. This makes the tool computationally efficient, and is preferred in resource-constrained regions. Importantly, the number of generated parameters is less as compared to existing DNNs.

For a quick and better understanding, we provide a few feature maps for CT scans and CXRs (both COVID-19 positive and negative) in Fig. 2. In Fig. 2, the highlighted

Table 1 Number of parameters for the different layers of the CNN architecture

Layer	Output dimension	Parameters
Convolution 1	$96 \times 96 \times 32$	2432
Convolution 2	$45 \times 45 \times 16$	8208
Convolution 3	$20 \times 20 \times 8$	1160
Dense 1	256	205056
Dense 2	50	12850
Dense 3 (Output layer)	2	102
Total	—	229,808

portions (in yellow) depict the areas of interest selected by the network.

4 Experimental setup

4.1 Dataset collection

In our collection, since both CXRs and CTs were not available from a single source, multiple sources were used. This means that, to avoid possible bias that can happen in DNN due to imbalance dataset, in our experiments, we created two different datasets from different sources.

a) CXR collection [37, 38]:

Two different sources were used for COVID-19 frontal CXR collections. The first collection [37] was composed of 168 COVID-19 positive cases and 65, non COVID-19 cases. Non-COVID-19 cases included several other lung diseases in the thick of MERS, SARS, and ARDS. To balance the number of instances in the non COVID-19 cases, other CXRs (50 normal and 53 pneumonia cases) were added from another collection [38]. Altogether, there were 168 COVID-19 and 168 non COVID-19 cases. These images were collected from different published papers available online or from pdfs. These images were of different qualities and sizes. Moreover, different images were noisy, skewed and had different orientations. Handling such characteristics was an extremely challenging task.

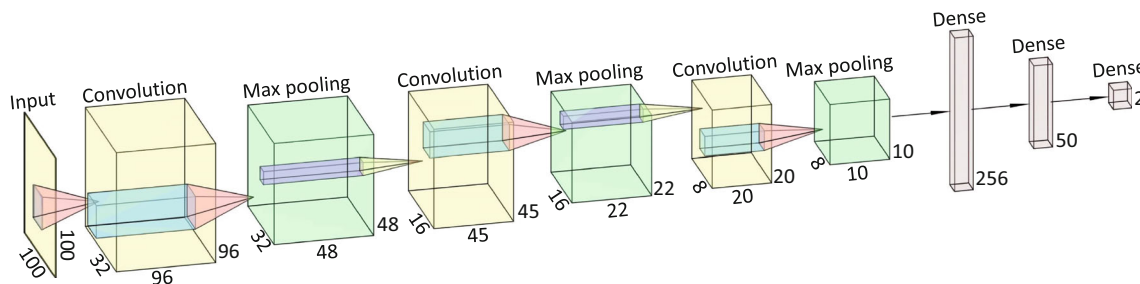
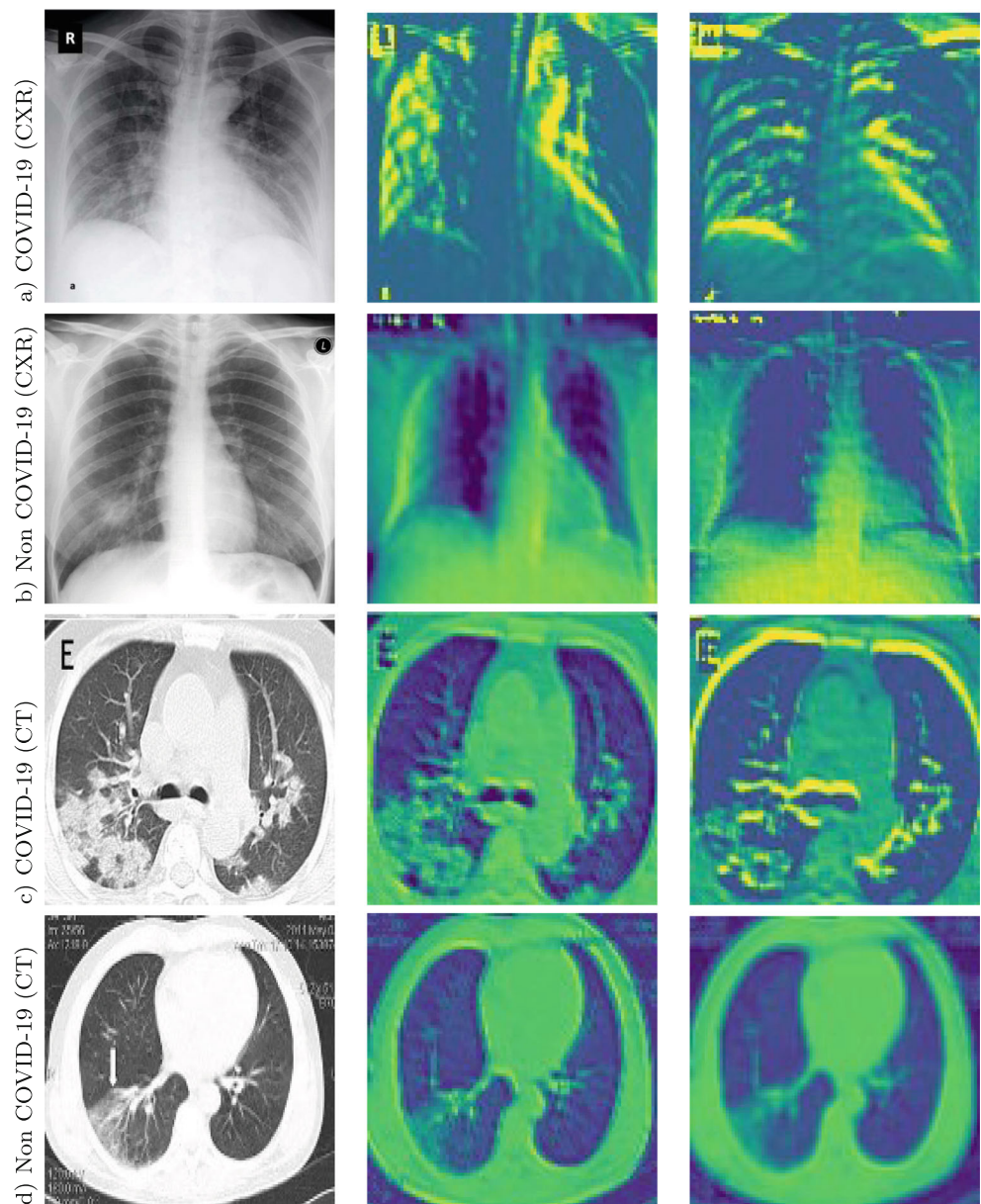


Fig. 1 The proposed CNN-tailored Deep Neural Network (DNN)

Fig. 2 Feature maps using both CXRs and CT scans by taking both classes: COVID-19 and non-COVID-19 cases



The system had to be robust enough and is not a surprise when we address real-world scenario.

b) CT scan collection [37, 39]:

Like before, two different sources were used to collect CT scans [37, 39]. The first collection was composed of 22 COVID-19 positive cases and 1 non-COVID-19 case [37]. As before, to ensure uniformity with that of the CXR collections, CT images were added

(146 COVID-19 positive cases and 167 non-COVID-19 cases) from another dataset [39]. In total, we have 168 COVID-19 cases and 168 non-COVID cases. In this collection, images were taken from disparate preprints as well as other existing datasets. The COVID-19 positive cases were taken from preprints. The labels were manually assigned using the figure captions. However, the text bodies related to the figures were

Table 2 Dataset collections

Collections (Image modality)	COVID-19 cases	Non COVID-19 cases	Total
CXR	168	168	336
CT	168	168	336
CXR + CT	336	336	672

considered, where the category of the images were not clear from the figure caption. A total of 349 COVID-19 positive CT images were obtained with various sizes and qualities. Moreover, the image quality also varied across different images. The minimum and maximum height of the images were 153 and 1853 pixels, respectively. The minimum and maximum width of the CT images were 124 and 1485 pixels, respectively. The average height and width of the images were 491 and 383. These images were from 216 patient cases. In the case of non COVID-19 cases, CT images were collected from disparate sources encompassing LUNA, MedPix, PubMed Central, and Radiopaedia website. 463 images were collected from these sources out of which the aforementioned sources had 36, 195, 202, and 30 images, respectively. Further, there were test and validation sets whose details are presented in [40].

As a result, we created a mixed dataset (and balanced dataset) by taking bot CXRs and CT scans, as provided in Table 2. Few samples are shown in Fig. 3.

4.2 Evaluation protocol

To validate the proposed model, 10-fold cross validation was used in the experiments. Cross validation ensures that each instance of the dataset is subjected to testing and training at least once. This in turn aids to avoid biased modeling of outliers. In each fold, we computed

the following evaluation metrics: Accuracy, Precision, Sensitivity (Recall), Specificity, F1 score, and Area under ROC curve (AUC). They were computed as,

$$\begin{aligned} \text{Accuracy} &= \frac{T_P + T_N}{T_P + T_N + F_P + F_N}, \\ \text{Precision} &= \frac{T_P}{T_P + F_P}, \\ \text{Sensitivity (Recall)} &= \frac{T_P}{T_P + F_N}, \\ \text{Specificity} &= \frac{T_N}{T_N + F_P}, \text{ and} \\ \text{F1 score} &= 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \end{aligned} \quad (3)$$

where T_P , T_N , F_P , and F_N refer to true positive, true negative, false positive, and false negative, respectively.

5 Our results and analysis

For the proposed DNN, the images were resized to 50×50 pixels, the batch size was 50 instances, and the number of training epochs was 100. These values are required to be tuned, since they vary from one dataset to another. In other words, they are application dependent. For optimal performance, training with the right parameters is the must. To understand this, following Table 3, we computed accuracies on different a) image sizes; b) batch sizes; c) training epochs; and d) dropouts.

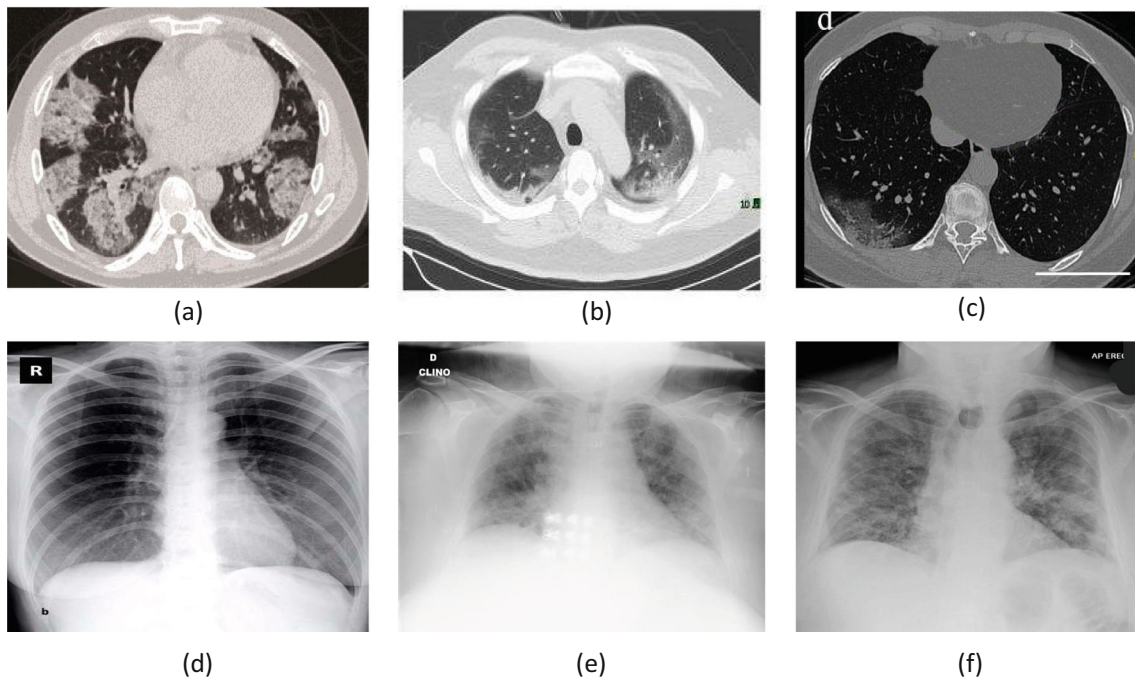


Fig. 3 Few samples of **a-c**: CT scans and **d-f**: CXR images

Table 3 Accuracies (ACC, in %) on different a) image sizes; b) batch sizes; c) training epochs; and d) dropouts

a)	
Image size	ACC (in %)
50×50	92.71
100×100	96.28
150×150	96.28
200×200	96.28
b)	
Batch size	ACC (in %)
50	92.71
100	88.39
150	81.85
200	81.85
c)	
Epoch	ACC (in %)
100	92.71
200	89.14
300	92.71
400	95.39
500	90.92
d)	
Dropout (%)	ACC (in %)
10	95.68
20	95.54
30	96.28
40	96.13
50	95.39
60	95.68
70	94.05
80	94.20
90	88.39

The figures in bold-face are best of all in that particular category

Table 4 Performance scores (using complete dataset: CXRs + CT scans)

Metrics	Scores
Sensitivity (Recall)	0.9792
Specificity	0.9464
Precision	0.9481
False positive rate	0.0536
False negative rate	0.0208
Accuracy (%)	96.28
F1 Score	0.9634
AUC	0.9808

a) Image size:

In our data collection, since raw image sizes were varied, we resized them into fixed dimensions. For our experiment, the resized dimensions were varied from 50×50 to 200×200 with a step of 50 pixels. The primary reason behind using different image sizes is to check whether we can empirically fix the dimension for rest of the experiments. Using mixed dataset (refer to Section 4.1), the obtained results are provided in Table 3 (a). We observed that the best result was obtained for 100×100 dimensions, and results remained to be constant for higher dimensions. Since, smaller image size implies lesser computational overhead, 100×100 pixel image size was considered.

With such image size, we were able to calculate hyper-paramters used in our architecture as shown in Table 1 in Section 1.

b) Batch size:

In a similar fashion, the batch sizes during training were varied from 50 to 200 instances with a step of 50, whose results are listed in Table 3 (b). We observed that performance gains were not remarkable on further increasing the batch size. Instead, a steady performance degradation was observed.

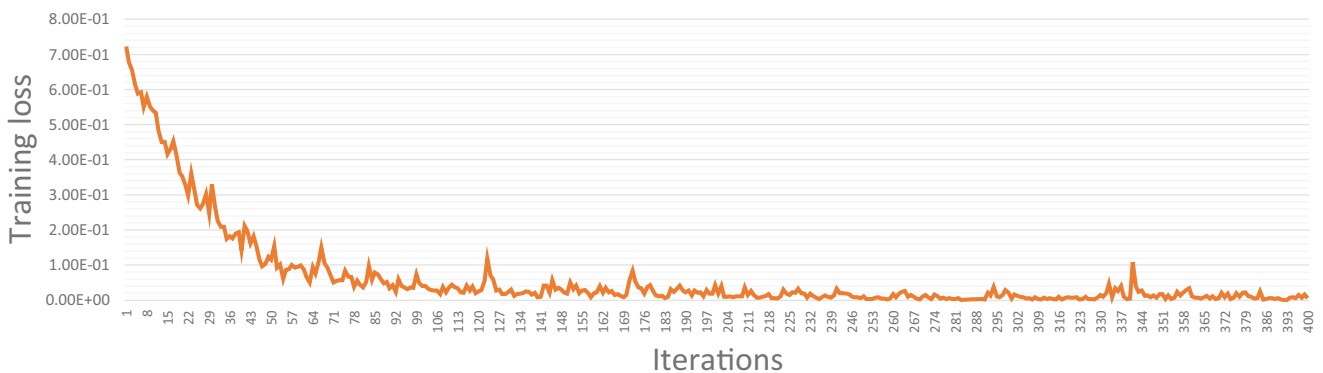


Fig. 4 Training loss for the proposed architecture

Table 5 a) Confusion matrix between COVID-19 and non COVID-19 categories and b) misclassified CXRs and CT scans in both categories: COVID-19 and non COVID-19

a)		
	COVID-19	Non COVID-19
COVID-19	329	7
Non COVID-19	18	318

b)		
	COVID-19	Non COVID-19
X-Ray	1	0
CT	6	18

c) Training epochs:

Further, in our experiments, the training epochs were varied from 100 to 500 with a step size of 100. The obtained accuracies are presented in Table 3 (c). The best performance was received from 400 training epochs.

d) Dropout (in %):

Finally, the percentage of dropout was varied from 10 to 90%, and their corresponding results are presented in Table 3 (d). In this experiment, we observed that the best result was received from 30% dropout. The performance was found to be decreases as the dropout was increased up to 60%. In the case of 90%, the accuracy dropped sharply by approximately 8.19%. The inter-class confusions for the 30% dropout setup is provided in Table 3 (d). In addition, the training loss is provided in Fig. 4.

With these learned parameters (empirically designed, as shown in Table 3), we computed performance scores for all

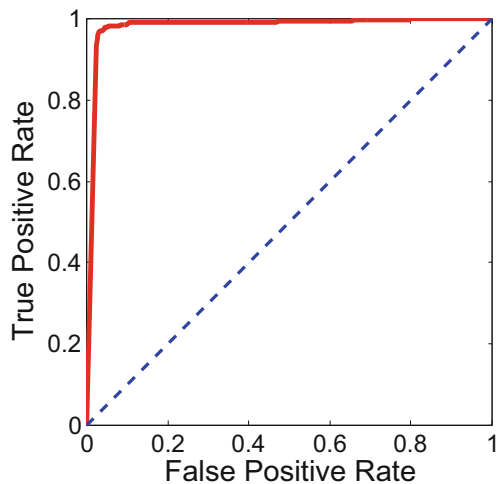


Fig. 5 ROC curve using proposed DNN (using complete dataset: CXRs + CT scans)

Table 6 Performance scores using CXRs and CT scans dataset separately

Metrics	CXR	CT
Sensitivity (Recall)	0.9940	0.9345
Specificity	0.9286	0.9821
Precision	0.9330	0.9813
False positive rate	0.0714	0.0179
False negative rate	0.006	0.0655
Accuracy (%)	96.13	95.83
F1 Score	0.9625	0.9573
AUC	0.9908	0.9731

evaluation metrics (as mentioned in Section 4.2). In Table 4, using the whole dataset, the proposed DNN was evaluated.

Confusion matrix, on the other hand, can help understand misclassified cases that include important information, such as false positive and true positive cases (see Table 5 (a)). We observed that only 7 cases of COVID-19 were misclassified as non COVID-19, and 18 non COVID-19 cases were classified as COVID-19. This could primarily be due to the inconsistency in data. In other words, images were not having exact same resolution since they were collected from different machines. In a few cases, multifarious images had foreign bodies and texts. As a consequence, such an issue impacted the model and led to the misclassification. The number of false negatives were analyzed, and we found that out of the 7 misclassifications, 6 were CT scans while 1 was CXR. In the case of false positives, all 18 misclassification cases were occurred for CT scans. For better understanding, misclassified cases for both data types are provided in Table 5 (b).

Further, to better understand the performance of the proposed DNN, ROC curve is provided in Fig. 5.

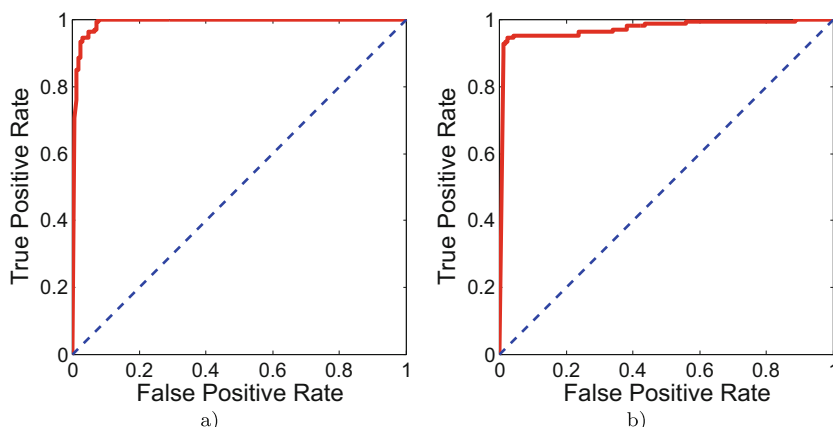
Considering those performance scores (cf. Table 4), confusion matrix (cf. Table 5), and ROC curve (cf. Figure 5), the proposed DNN provides encouraging results. These results demonstrated the fact that multiple modalities can be

Table 7 Confusion matrices: a) CXR dataset (ACC = 96.13%) and b) CT scan (ACC = 95.83%)

a)		
	COVID-19	Non COVID-19
COVID-19	167	1
Non COVID-19	12	156

b)		
	COVID-19	Non COVID-19
COVID-19	157	11
Non COVID-19	3	165

Fig. 6 ROC curves for separate data types: a) CXRs and b) CT scans, using the proposed DNN



used to training and testing a single deep neural network to detect COVID-19 positive cases. To prove such statement, it is important to check whether there is any loss or improvement in case CT scans and CXRs are separately employed. Therefore, in what follows, we discuss on separate experimental results from CXRs and CT scans. For both data types: CXR and CT scans, note that experiments were performed on exactly same architecture as before.

Using 10-fold cross validation, using CXR and CT scans separately, we computed performance scores for all evaluation metrics as in Table 4. For both data types, separate results are provided in Table 6. As before, confusion matrices are provided in Table 7. For better understanding, separate ROC curves are shown in Fig. 6. With these additional tests (*cf.* Table 6 and Fig. 6), we observed that one DNN can be used to train/test, since we did not find any remarkable interventions from one data type to another. The statement can be checked by taking two different experimental results in Tables 4 and 6.

6 Comparison with other DNNs

Instead of relying on the performance from the proposed DNN, we further experimented by taking three different DNNs to check whether it is possible to train/test multimodal radiological image data: CT scans and CXRs. The primary idea is to check; can one architecture be used to train/test both CT scans and CXRs?

Table 8 Parameters used in different DNNs

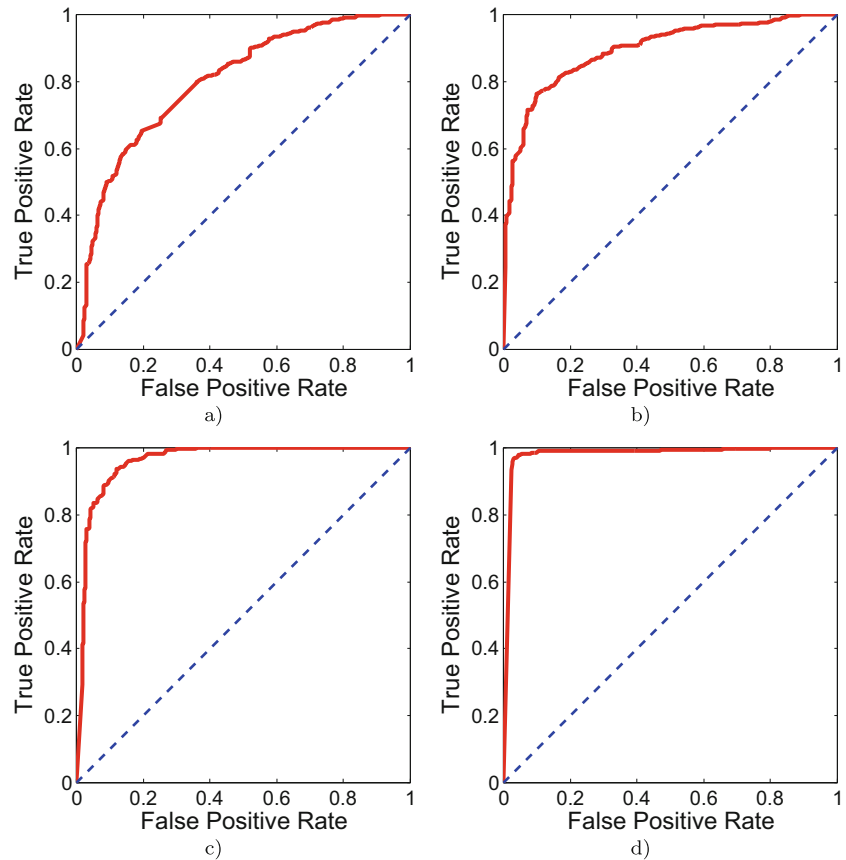
Architecture	Parameters
InceptionNet	26,522,146.00
MobileNet	7,423,938.00
ResNet	49,278,594.00
Proposed DNN	229,808.00

- c) Inception [41]: It binds a sparse CNN with a normal dense network. Due to the effectiveness of small number of neurons, the number of the convolutional filter for a specific kernel is kept small. Also, it applies convolutions of varied dimensions to get the information of different ranges. Another important factor is that it has a bottleneck layer, which helps in reducing possible heavy computation.
- a) MobileNet [42]: It is a lightweight network designed to use depth-wise distinct convolutions, which represents using a single convolution on each channel instead of combining all the convolutions. The network efficiently enhances the performances of embedded systems having limited resources.
- b) ResNet [43]: This network is designed, where the future layer is formed by merging the previous layers. It is done in order to force the architecture to get the knowledge about the surplus. This network also applies skip connections similar to LSTM, where these connections are drives through number of gates. The amount of information pass through the skip connection is estimated by the gates. It is capable of training hundreds or even thousands of layers and attains imperative performance.

Table 9 Comparison (using CXRs + CT scans): Existing DNNs and the proposed DNN

Metrics	InceptionV3	MobileNet	ResNet	Proposed DNN
Sensitivity (Recall)	0.6935	0.7708	0.9375	0.9792
Specificity	0.744	0.8869	0.878	0.9464
Precision	0.7304	0.8721	0.8848	0.9481
False positive rate	0.256	0.1131	0.122	0.0536
False negative rate	0.3065	0.2292	0.0625	0.0208
Accuracy (%)	71.88	82.89	90.77	96.28
F1 Score	0.7115	0.8183	0.9104	0.9634
AUC	0.8011	0.8951	0.9616	0.9808

Fig. 7 ROC curves using **a** InceptionV3, **b** MobileNet, **c** ResNet, and **d** Proposed DNN



For these experiments, as in Table 8, different numbers of parameters were used. For a quick comparison, the number of parameters used in the proposed DNN is provided, where it is clear that the proposed architecture is computationally efficient. Like before, we followed the exact same dataset (complete dataset: CXRs + CT scans) and evaluation protocol. In Table 9, a complete comparative study was provided. In addition, the proposed DNN was compared. Further, ROC curves from all DNNs are shown in Figure 7.

7 Conclusion

In this paper, we have addressed the usefulness of a single CNN architecture for different data modalities (or data types) to detect COVID-19 positive cases, where we proposed a lightweight (9 layered) CNN-tailored deep neural network. We have trained and tested both data types: Chest X-ray and CT scan images, and have achieved an overall accuracy of 96.28% (AUC = 0.9808, and false negative rate = 0.0208). With these results, we have observed that multiple radiological imaging data to one architecture. Further, we have achieved coherent results in detecting COVID-19 positive cases from major existing DNNs, such as InceptionV3, MobileNet, and ResNet.

As mentioned state-of-the-art literature [1], our immediate plan is to work on computationally efficient CNN-tailored DNN by taking multimodal data, not just limited to two data types.

Compliance with Ethical Standards

Conflict of interests The authors declare no conflict of interest. All authors have read and agreed to the published version of the manuscript.

Abbreviations The following abbreviations are used in this manuscript: AI, Artificial Intelligence; COVID-19, Coronavirus Disease 2019; SERS, Severe Acute Respiratory Syndrome; MERS, Middle East Respiratory Syndrome; CXR, Chest X-ray; CT, Computed Tomography; CNN, Convolutional Neural Network; GGO, Ground-Glass Opacity; DNN, Deep Neural Network; RT-PCR, Reverse Transcription Polymerase Chain Reaction.

References

1. Santosh KC (2020) Ai-driven tools for coronavirus outbreak: need of active learning and cross-population train/test models on multitudinal/multimodal data. *J Med Syst* 44(5):1–5
2. <http://www.who.int/csr/don/12-january-2020-novel-coronavirus-china/en/>. Novel coronavirus – China. Online 2020
3. <https://www.who.int/emergencies/diseases/novel-coronavirus-2019/situationreports>. WHO Report. Online 2020

4. https://www.who.int/csr/sars/country/table2004_04_21/en/. Summary of probable SARS cases with onset of illness from 1 November 2002 to 31 July 2003. Online 2020
5. <http://www.who.int/emergencies/merscov/en/>. MiddleEastrespiratorysyndromecoronavirus(MERS-CoV). Online 2020
6. Ball L, Vercesi V, Costantino F, Chandrapatham K, Pelosi P (2017) Lung imaging: how to get better look inside the lung. *Annals of translational medicine*, 5(14)
7. Lomoro P, Verde F, Zerboni F, Simonetti I, Borghi C, Fachinetti C, Natalizi A, Martegani A (2020) Covid-19 pneumonia manifestations at the admission on chest ultrasound, radiographs, and ct: single-center study and comprehensive radiologic literature review. *European journal of radiology open*, pp 100231
8. Huang C, Wang Y, Li X, Ren L, Zhao J, Hu Y, Zhang L, Fan G, Xu J, Gu X et al (2020) Clinical features of patients infected with 2019 novel coronavirus in wuhan, china. *Lancet* 395(10223):497–506
9. Fang Y, Zhang H, Xie J, Lin M, Ying L, Pang P, Ji W (2020) Sensitivity of chest ct for covid-19: comparison to rt-pcr. *Radiology*, pp 200432
10. Ng M-Y, Lee EYP, Yang J, Yang F, Li X, Wang H, Lui MM-s, Lo CS-Y, Leung B, Khong P-L et al (2020) Imaging profile of the covid-19 infection: radiologic findings and literature review. *Radiol Cardiothor Imaging* 2(1):e200034
11. Li Y, Xia L (2020) Coronavirus disease 2019 (covid-19): role of chest ct in diagnosis and management. *American Journal of Roentgenology*, pp 1–7
12. Ye Z, Zhang Y, Yi W, Huang Z, Song B (2020) Chest ct manifestations of new coronavirus disease 2019 (covid-19): a pictorial review. *European Radiology*, pp 1–9
13. Zhou S, Wang Y, Zhu T, Xia L (2020) Ct features of coronavirus disease 2019 (covid-19) pneumonia in 62 patients in wuhan, china. *American Journal of Roentgenology*, pp 1–8
14. Chung M, Bernheim A, Mei X, Zhang N, Huang M, Zeng X, Cui J, Xu W, Yang Y, Fayad ZA et al (2020) Ct imaging features of 2019 novel coronavirus (2019-ncov). *Radiology* 295(1):202–207
15. Song F, Shi N, Shan F, Zhang Z, Shen J, Lu H, Ling Y, Jiang Y, Shi Y (2020) Emerging 2019 novel coronavirus (2019-ncov) pneumonia. *Radiology* 295(1):210–217
16. Pan F, Ye T, Sun P, Gui S, Liang B, Li L, Zheng D, Wang J, Hesketh RL, Yang L et al (2020) Time course of lung changes on chest ct during recovery from 2019 novel coronavirus (covid-19) pneumonia. *Radiology*, pp 200370
17. Wang S, Kang B, Ma J, Zeng X, Xiao M, Guo J, Cai M, Yang J, Li Y, Meng X et al (2020) A deep learning algorithm using ct images to screen for corona virus disease (covid-19) MedRxiv
18. Butt C, Gill J, Chun D, Babu BA (2020) Deep learning system to screen coronavirus disease 2019 pneumonia. *Applied Intelligence*, pp 1
19. Yoon SH, Lee KH, Kim JY, Lee YK, Ko H, Ki HK, Park CM, Kim Y-H (2020) Chest radiographic and ct findings of the 2019 novel coronavirus disease (covid-19): analysis of nine patients treated in korea. *Korean J Radiol* 21(4):494–500
20. Wang L, Wong A (2020) Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest radiography images. arXiv:2003.09871
21. Sethy PK, Behera SK (2020) Detection of coronavirus disease (covid-19) based on deep features. Preprints, 2020030300:2020
22. Zhang J, Xie Y, Li Y, Shen C, Xia Y (2020) Covid-19 screening on chest x-ray images using deep learning based anomaly detection. arXiv:2003.12338
23. Chen N, Zhou M, Dong X, Qu J, Gong F, Han Y, Qiu Y, Wang J, Liu Y, Wei Y et al (2020) Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in wuhan, china: a descriptive study. *Lancet* 395(10223):507–513
24. Ozturk T, Talo M, Yildirim EA, Baloglu UB, Yildirim O, Acharya UR (2020) Automated detection of covid-19 cases using deep neural networks with x-ray images *Computers in Biology and Medicine*, pp 103792
25. Narin A, Kaya C, Pamuk Z (2020) Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. arXiv:2003.10849
26. Mangal A, Kalia S, Rajgopal H, Rangarajan K, Namboodiri V (2020) Subhashis banerjee, and chetan arora. Covidaid: Covid-19 detection using chest x-ray. arXiv:2004.09803
27. Wang L, Wong A (2020) Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images. arXiv:2003.09871
28. Zheng C, Deng X, Fu Q, Zhou Q, Feng J, Ma H, Liu W, Wang X (2020) Deep learning-based detection for covid-19 from chest ct using weak label. medRxiv
29. Farooq M, Hafeez A (2020) Covid-resnet: A deep learning framework for screening of covid19 from radiographs. arXiv:2003.14395
30. Hall LO, Paul R, Goldgof DB, Goldgof GM (2020) Finding covid-19 from chest x-rays using deep learning on a small dataset. arXiv:2004.02060
31. Salman FM, Abu-Naser SS, Alajrami E, Abu-Nasser BS, Alashqar BAM (2020) Covid-19 detection using artificial intelligence
32. Alom MdZ, Rahman MM, Nasrin MS, Taha TM, Asari VK (2020) Covid_mtnet: Covid-19 detection with multi-task deep learning approaches. arXiv:2004.03747
33. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521(7553):436–444
34. Goodfellow I, Bengio Y, Courville A (2016) Deep learning. MIT Press, Cambridge
35. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*, pp 1097–1105
36. <https://medium.com/@amarbudhiraja/https-medium-com-amarbudhiraja-learning-less-to-learn-better-dropout-in-deep-machine-learning-74334da4bfc5>. CNN. Online 2020
37. <https://github.com/ieee8023/covid-chestxraydataset>. Covid chest XRay. Online 2020
38. <https://www.kaggle.com/paultimothymooney/chest-xraypneumonia>. Chest XRay (Pneumonia). Online 2020
39. <https://github.com/UCSDAI4H/COVID-CT>. COVID CT. Online 2020
40. Zhao J, Zhang Y, He X, Xie P (2020) Covid-ct-dataset: a ct scan dataset about covid-19. arXiv:2003.13865
41. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2016) Rethinking the inception architecture for computer vision. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp 2818–2826
42. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H (2017) Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv:1704.04861
43. Targ S, Almeida D, Lyman K (2016) Resnet in resnet: Generalizing residual architectures. arXiv:1603.08029

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Himadri Mukherjee has completed B.Sc. in Computer Science from Acharya Prafulla Chandra College, M.Sc. (Gold Medalist) and Ph.D. in Computer Science from West Bengal State University in the year 2013, 2015 and 2020 respectively. He is currently working as a project linked personnel in the Department of Computer Science of the same University. He has published more than 60 papers in International journals and conferences. His research interests

include audio signal processing, musical information retrieval, image processing, natural language processing, pattern recognition and machine intelligence.



Subhankar Ghosh has received B.Sc. degree in Computer Science from University of Calcutta, Kolkata in 2013 and M.Sc. degree in the discipline from West Bengal State University in 2016. He is currently working as a Research Engineer in Indian Statistical Institute, Kolkata. He has published more than 10 papers in International journals and conferences. His research interests include image processing, computer vision and deep learning.



Ankita Dhar has completed B.Sc. in Computer Science from East Calcutta Girls' College, M.Sc. (Gold Medalist) in Computer Science from West Bengal State University in the year 2011 and 2013 respectively. She is currently a DST Inspire Fellow and working as a Research scholar in the department of Computer Science of the same University. She has published over 30 research papers in reputed conferences and journals. Her research interest includes Nat-

ural Language Processing, Information Retrieval, Audio Signal Processing and Machine Intelligence.



Sk Md Obaidullah has received Ph.D.(Engg.) from Jadavpur University, M.Tech in Computer Science & Application from University of Calcutta and B.E in Computer Science & Engineering from Vidyasagar University in the year 2017, 2009, 2004 respectively. He was Erasmus Post-Doctoral fellow at University of Evora, Portugal from Nov. 2017 to Sept. 2018. In the year 2019 he was awarded the prestigious TARE Fellowship by

DST SERB, Govt. of India. Dr. Obaidullah has eleven years of professional experience in academics and research. Presently he is working as an Associate Professor in the Department of Computer Science & Engineering, Aliah University, Kolkata. He has published more than 90 research papers in reputed peer reviewed journals and national/international conferences. His research interests are Pattern Recognition, Machine Learning, Document Image Processing, Medical Image Analysis, Signal Processing. His details biography can be found here: <https://aliah.ac.in/faculty-details.php?id=33>.



Dr. K. C. Santosh (IEEE Senior Member) is the Chair and an Associate Professor of the Department of Computer Science at the University of South Dakota (USD). Before joining USD, Dr. Santosh worked as a Research Fellow at the US National Library of Medicine (NLM), National Institutes of Health (NIH). He worked as a Postdoctoral Research Scientist at the LORIA Research Centre, Universite de Lorraine, in direct collaboration with ITESOFT, France. He also served as a Research Scientist


at the INRIA Nancy Grand Est Research Centre, France, where he has received his Ph.D. diploma in Computer Science. Dr. Santosh has published more than 70 peer-reviewed research articles, 100 conference proceedings, and 11 book chapters. He has authored 4 books and edited 3 books, 14 journal issues, and 6 conference proceedings. He is currently Editor-in-Chief of IJSIP and an Associate Editor for several journals, such as International Journal of Machine Learning and Cybernetics and IEEE Access. He has also chaired more than 10 international conference events. His research projects have been funded by multiple agencies, including the SDCRGP, Department of Education (DOE), and the National Science Foundation (NSF). Dr. Santosh is the proud recipient of the Presidents Research Excellence Award (USD, 2019) and an award from the Department of Health & Human Services (2014).



Kaushik Roy Kaushik Roy has completed B.E in Computer Science & Engineering from NIT Silchar, M.E and PhD (Engg.) in Computer Science & Engg. from Jadavpur University in the year 1998, 2002 and 2008 respectively. He is currently working as a Professor and Head of the Department of Computer Science, West Bengal State University, Barasat, India. In 2004 he received the Young IT Professional award from the Computer Society of India. He has

published 2 books and more than 50 research articles in International Journals and more than 140 papers in various reputed Conferences and book chapters including ICPR, ICDAR, ICFHR, CBMS, ICVGIP etc. His research interest includes pattern recognition, document image processing, medical image analysis, online handwriting recognition, speech recognition and audio signal processing. He is Life Member of IUPRAI (an unit of IAPR) and Computer Society of India.

Affiliations

Himadri Mukherjee¹ · Subhankar Ghosh² · Ankita Dhar¹ · Sk Md Obaidullah³ · K. C. Santosh⁴  · Kaushik Roy¹

Himadri Mukherjee
himadrim027@gmail.com

Subhankar Ghosh
sgcs2005@gmail.com

Ankita Dhar
ankita.ankie@gmail.com

Sk Md Obaidullah
sk.obaidullah@gmail.com

Kaushik Roy
kaushik.mrg@gmail.com

¹ Department of Computer Science, West Bengal State University, West Bengal, India

² CVPR Unit, Indian Statistical Institute, Kolkata, India

³ Department of Computer Science, Engineering, Aliah University, Kolkata, India

⁴ Department of Computer Science, The University of South Dakota, Vermillion, SD, USA