



CoWarriorNet: A Novel Deep-Learning Framework for CoVID-19 Detection from Chest X-Ray Images

Indrani Roy¹ · Rinita Shai² · Arijit Ghosh¹ · Anirban Bej¹ · Soumen Kumar Pati³ 

Received: 5 September 2021 / Accepted: 21 November 2021 / Published online: 3 December 2021
© Ohmsha, Ltd. and Springer Japan KK, part of Springer Nature 2021

Abstract

Even after scavenging the existence of mankind for the past year, the wrath of CoVID-19 is yet to die down. Countries like India are still getting haunted by the devastating conundrum, with coronavirus ripping through its citizens in the concurrent second wave. The surge of cases has prompted rapid intervention, with medical authorities pushing it to the limit to curve a roadblock to its aggressive growth. But, even after effortless work, human intervention remains slow and insufficient. Furthermore, relevant testing methodologies have shown weakness while detecting threats, with the recent growth of post-Covid complexities, thereby leaving a painful mark. This as such created a major requirement for technological advancements, which can cater to the mass. The growth of computational prowess in the past decade made the field of Deep Learning a major contributor in curving out algorithms to solve this. Adding to the excellent foundation of Deep Learning, this paper, proposes a novel CoWarriorNet model for rapid detection of CoVID-19, via chest X-ray images, which adds in an extra layer of precision and confirmation in the detection of cases in both pre-Covid and post-Covid conditions. The proposed classification model curves out an excellent accuracy of 97.8%, with the major eye-candy being the sensitivity rate of 0.99 when detecting CoVID-19 cases. This model introduces a new concept of Alpha Trimmed Average Pooling, which along with the novel architecture adds a subtle touch to its high efficiency, thereby giving a much-needed solution to the medical experts. The two-mouthed architecture provides the added benefit of a confidence score, deducing human aid in case of discrepancy.

Keywords CoVID-19 chest X-ray images · CoWarriorNet · Classification model · Deep learning · Alpha Trimmed Average Pooling

✉ Soumen Kumar Pati
soumenkrpati@gmail.com

Extended author information available on the last page of the article

Introduction

The highly mutative SARS-COV-2 [1], commonly known as CoVID-19, bundled the entire existence of mankind in its initial surge washing away millions of innocent lives all around the world. Its adaptive variants curved out the second wave with India facing the worst of its wrath with around cumulative 383,490 deaths and 29.70 million positive cases as of 17th June 2021 (<https://ourworldindata.org/>). Marking its higher potency and lack of infrastructure to curb the exponential rise of infection, patients are dying without even being tested properly. Even though the testing graph shows quite a high rate, it is quite saturated when compared to the huge upsurge in positive cases.

As of now, the most viable resource for testing is the highly effective Real-Time Reverse Transcription Polymerase Chain Reaction [2], abbreviated as RT-PCR. Even though its sensitivity has marked a spike through repeated testing, it still beckons the question of its below-par one-shot sensitivity rates [3]. Furthermore, an RT-PCR test is still quite slow, with it taking at least a day to hand in a confident result (<https://www.healthline.com/>). To induce a time-efficient approach, Rapid Antigen Test or RAT has been introduced as a feasible alternative, and even though the reports can be handed in within minutes, its sensitivity is extremely low (around 30.2%) [4], showcasing a poor choice in terms of testing. As a result, many works cite the argument of validation through the means of various radiographical examinations [5, 6].

Leaning to find radiographical alternatives, like many other situations, computed tomography (CT) [7] and chest X-ray (CXR) [8] were thoroughly gone through in numerous works. CT scans, in particular, showed excellent sensitivity and specificity in the detection of CoVID-19 [9]. But, from a practical perspective, CT scan imaging brings up quite a lot of difficulties. First, it requiring expensive machinery for its working is more of a disadvantage than its preliminary advantages. Since the highest hit countries of the second wave are developing countries, the budget is quite tight, to say the least, and the bearing cost of installing as many CT machines as possible to carry out testing is quite impractical, to be honest. Second, CT scans require the suspected patients to come in contact with larger portions of the machine, thereby making it very time-consuming, with sanitization being required after each test gulping huge chunks of time. Third, its immobility. CT machines are gigantic and need to be installed in a spacious location, and lack the mobility that other modes of radiography imaging bring to the table. Finally, the need for expert radiologists to track the outputs. In this dire situation, the number of health workers is very bleak and the need for a radiologist to constantly check on the results brings in way more inefficiency in terms of human resource when dealing with the pandemic.

Therefore, the most realistic alternative is utilizing the prospect of chest X-ray imaging for the detection of CoVID-19. It brings up every disadvantage that a CT has and makes them its advantages. **Cost-effective:** X-ray machines are considered rudimentary for any medical facility, as such almost every hospital, nursing homes, and clinics have them. Therefore, a developing country, hugely affected by

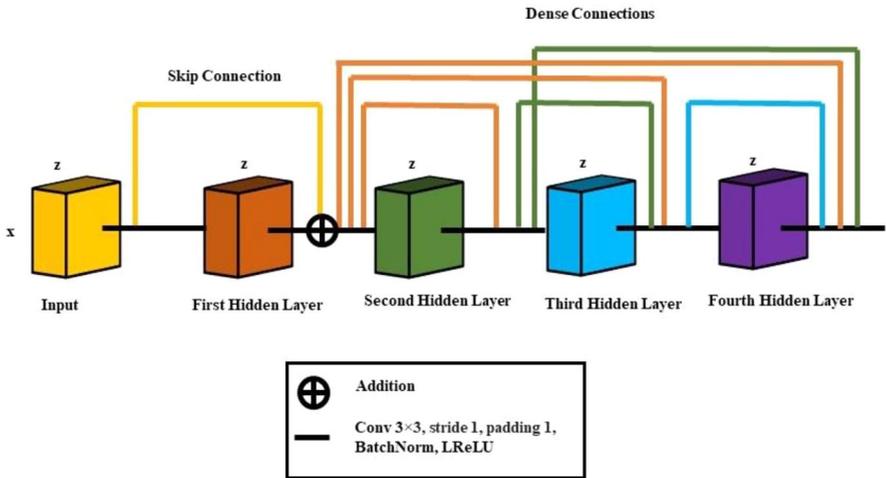


Fig. 2 Our proposed ResiDense module

ResiDense modules, shown in Fig. 2. To further enhance the network capability, a novel Pooling Layer, *Alpha Trimmed Average Pooling* has been introduced. Furthermore, a neural network is not perfect and in delicate scenarios like that of detection of CoVID-19, it does require human intervention from time to time. As such, our network does try to imbibe the need for human intervention through its confidence score along with the predicted output class. Finally, to address the post-covid pneumonia syndrome a lot of focus has been given to the sensitivity rate of pneumonia cases too.

The subsequent “[Related Work](#)” addresses all the related works carried on in this field of study. It is followed up by “[Organization of the Paper](#)” which facilitates a roadmap of the paper for easier scanning.

Related Work

Hussain et al. [11] proposed a model CoroDet based on Convolutional Neural Network (CNN) [12] having 22 layers to perform the CoVID-19 detection on the CoVID-R dataset. However, this model suffers from vanishing gradient and exploding gradient problems. Ismael et al. [13] extended this work with pre-trained deep CNN (VGG16, VGG19, ResNet18, ResNet-50, and ResNet101) for feature extraction. Thereafter, they used Support Vector Machine for classification, utilizing several kernels, namely Linear, Cubic, Quadratic, and Gaussian functions. Since they were Imagenet pre-trained models, the evaluation metrics varied due to the lack of learned features specific for identifying CoVID-19 [14] cases, thereby pushing the models to overfit. Basu et al. [15] proposed a novel concept called domain extension transfer learning (DETL) with deep CNN [16] to classify the data in four classes based on their respective Class Activation Map. This model also suffers from the aforementioned drawbacks. Adding to it, due to the use of Grad-CAM, the model

has exponential time complexity. Jain et al. [17] analyzed the effective deep learning techniques (ResNeXt, Inception V3, and Xception) on a dataset containing 6432 images which achieved a staggering performance accuracy of 97.97%. But one of the major drawbacks of this model is that it worked on augmented images, so a huge chunk of time was consumed in the process of data augmentation. Nayak et al. [18] performed a comparative evaluation between eight effective CNN models (AlexNet, VGG16, GoogleNet, ResNet-34, MobileNet-V2, SqueezeNet ResNet-50 and Inception-V3) to showcase their potential in CoVID-19 classification. The study showed that ResNet-34 achieved the highest accuracy of 98.33%. Moreover, the study also showcased that all the applied models have the overfitting problem—again mainly due to unrelated learned features by pre-trained models. Hemdan et al. [19] derived a CoVID X-Net model which mainly consists of seven various architectures namely, Visual Geometry Group Network (VGG19), Google MobileNet-V2, etc. to classify the chest X-ray images into the positive and negative classes (i.e., binary classification). The model seems to be massively bulky given that it was to be utilized for the binary classification task. Toğaçar et al. [20] used fuzzy colors with the stacking procedure with a deep learning model, Squeeze Net, to maneuver Social Mimic Optimization for multiclass classification. Ouchicha et al. [21] proposed a novel architecture, CVDNet, based on Residual Neural Network, which combines local and global features to classify the chest X-ray images (CoVID-19, normal and viral pneumonia). According to [22], ResNet has a very high inference time per image (best case 8.90 ms and worst case 84.52 ms) which dwindles the average response time of the designed architecture. Chaudhary et al. [23] utilized deep CNN to classify the CoVID CXR images in the CoVIDX dataset. However, the model seems not properly trained, resulting in the constant variation of statistical parameters. This is evident by the results depicted in the paper, which indicates an underfit model. Tang et al. [24] achieve the goal by ensemble multiple snapshots of CoVID Net design which gets a performance accuracy of 93.5% but it has a very low F1 score. Ramirez et al. [25] proposed uncertainty estimation through Monte-Carlo dropout, Softmax scores model in a MixMatch semi-supervised structure. They obtained Jensen–Shannon distance to evaluate its performance, whereas the PSNR, SISIM, FISIM have not been evaluated. Later on, they extend this work in [31] where they used scarce labeled data to overcome the aforementioned drawbacks. Jain et al. [26] used the YoloV3 model to detect the positive cases from chest X-ray images where the Gaussian blur and data augmentation were performed as a necessary pre-processing step. On an average case, YoloV3 takes 18 h to train with a Google Colab GPU. That is why the model is not very time efficient. Haghanifar et al. [27] implemented CheX Net which comprised of U-Net to extract the ROI through image segmentation. Later on, they applied Dense Net 121 as a backbone architecture to perform the detection. Although the model achieved a higher accuracy, the segmentation and detection is a sequential process and not applied in the real-time scenario. Luz et al. [28] used a pre-trained Efficient Net model evaluated on the CoVIDx dataset which also suffered the aforementioned problems of the pre-trained models along with data augmentation. Karakanis et al. [29] obtained Generative Adversarial Network with ResNet8 as the discriminator which utilized transfer learning for real-time weight transfer, thereby deducing exponential space complexity. Ibrahim et al.

[30] utilized the highly popular AlexNet, to showcase the difference between viral pneumonia and CoVID-19 chest X-ray. Sakib et al. [32] proposed DL-CRC which consisted of a Generative Adversarial Neural network and Deep CNN to perform the binary classification. It has also the same drawbacks as mentioned for [29]. Panwar et al. [33] obtained a deep transfer learning algorithm with grad-CAM color visualization to separate the classes based on their activation map. A better alternative could have been the utilization of TSNE which can provide better feature distribution in less time. Kamal et al. [34] also used Deep Convolutional Neural Network to solve the aforementioned problem. However, it suffered from vanishing gradient and exploding gradient problems. Not only that, Gomes et al. [45] proposed an AI framework for texture analysis of the CoVID-19 chest X-ray images, however, for diagnosis, it is not the best approach. Therefore, they gradually shifted to the pseudo-convolutional machines with the help of RT-PCR results to characterize the virus sequences [46]. In the meantime, Ismael et al. [47] published a survey paper to show the performance of the multiresolution approaches on CoVID-19 chest X-ray images which shows sometimes image resolution creates a major issue in detection accuracy. Lastly, Singh et al. [35] proposed a novel design of Gen-ProtoPNet based on NP-complete problem by ensemble 30 prototypes of different CNN architectures for CoVID-19 chest X-ray detection. It is the latest updated architecture to date.

From the above discussion, it can be concluded that most of the deep learning models implement conventional Neural Networks with transfer learning for fine-tuning. Some use GAN for data augmentation and others extract the ROI through image segmentation. All these state-of-the-art approaches faced the problems of overfitting, underfitting, vanishing gradient, exploding gradient, space complexity, and exponential time complexity. To overcome these, a novel architecture is proposed in the following sections.

Organization of the Paper

The paper is oriented in a quite specific chronology with the proposed architecture discussed in the subsequent “[Proposed Methodology](#)”. The results obtained are discussed in “[Experimental Results](#)”, which addresses the efficiency of the architecture and its excellent adaptation to detecting CoVID CXR images. Finally, the proposed method is concluded in “[Conclusion and Future Scope](#)”, pointing to further improvements that can be adopted in the future for better results.

Proposed Methodology

The proposed methodology incorporates three modular parts with each serving specific purposes and integrates them to device a stringent CoVID-19 CXR detection algorithm. All the specific modules are discussed as follows. Figure 3 obtains the workflow of the proposed architecture.

According to Fig. 3, first, the chest X-ray images are fed into the derived UNET model to perform feature mapping. Then, the mapped features are passed

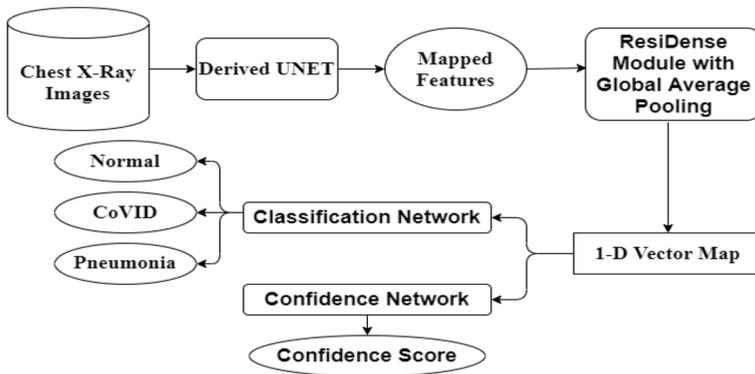


Fig. 3 Workflow of the CoWarriorNet

through ResiDense module which performs feature filtering and the added Global Average Pooling converts the features in the 1-D Vector Map. These are fed into Terminal Networks which consist of two parallel networks namely Classification Network and Confidence Network. The classification network is responsible for the multiclass classification and the second one is responsible to show the efficiency of the obtained classification.

Derived UNET for Feature Mapping

The proposed architecture initiates with a derived architecture of the highly efficient UNET [36], which picks up the input image and reproduces a specific map to serve as specifics of “where to look” for the image. The map is enhanced through automatic gradients with backpropagation and is the quintessential building block for efficient output classification. One of the major differences our architecture has from the original UNET architecture is in the use of Alpha Trimmed Average Pooling (discussed in detail in the latter half of this section) in place of Max-Pooling [37]. The Derived UNET has in total nine Identity blocks with four down-sample blocks, four up-sample blocks, and one bridge block joining the two modules. Each of the nine blocks has two convolutional layers with 3*3 kernels, one step stride, and one added padding pixel along the border, followed by Batch Normalization [38] and Rectified Linear Unit (ReLU) [37] adding non-linearity. The down-sampling block and the up-sampling block differ in the sense that the down-sampling ones are followed by a pooling layer whereas the up-sampling blocks are followed by transposed convolution. Similar, to the original UNET paper, our derived module also concatenates the feature maps from the down-sampled block with the parallel up-sampled block to provide soft attention to the image feature map in contention. The final feature map is of the same spatial dimension as the input, and also has the input image concatenated to it, deriving difference from the original UNET architecture.

ResiDense Module

The output of the Derived UNET is fed into two ResiDense modules, which compresses the features to feed into the classification network. The ResiDense module comprises hybrid residual and dense connections, which adds inefficient computation through clubbing of the hybrid connections. Our ResiDense module consists of five convolutional layers (as shown in Fig. 2), each followed by Batch Normalization and ReLU activation function. In between the initial two layers, an additive residual skip connection is added, which helps in the avoidance of accuracy degradation via the equation given as follows:

$$h(x) = \rho(x) + x, \quad (1)$$

where x is the input, $\rho(x)$ depicts the residue, and $h(x)$ depicts the prediction of the layer with a shortcut. Among the next three layers, densely concatenated connections persist. This type of module is improvised since the initial convolution does not show much change and simple and computationally inexpensive residual connection does enough justice to thrust excellent feature recognition. The later layers capture much deeper orientation, thereby requiring a more adaptive dense connection. The proposed module not only enhances and adjusts to deterioration of the accuracy but also helps in the smooth flowing of gradients while back-propagating for parameter adjustment.

Terminal Networks

The output feature map from the ResiDense module gets transformed into a vector through Global Average Pooling [39], which converts the maps into a 1-D vector of size equal to the number of channels of the map. In the proposed architecture, this vector is bifurcated into two parallel networks, one which predicts the output classification (Classifier Network) and one which depicts a confidence score (Confidence Network [40]). The proposed Classifier Network is a single-layered perceptron (shown in Fig. 4a), which outputs three probability scores, each for the classes of the utilized dataset. The Confidence Network is something that brings about subtlety to utilizing Deep Learning in the highly sensitive medical field. Through the Confidence Network, the proposed model tries to predict how confident the predictions are. The main tricky part of the network was to get this score. Popping out the highest predicted probability as the score can indeed be a viable option. But it is to be noted that in most cases, the final

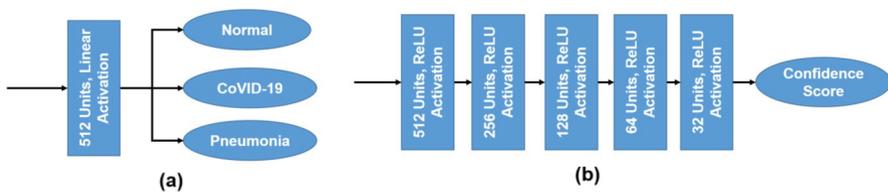


Fig. 4 Terminal Networks. **a** Proposed classification network, **b** proposed confidence network

output is the class that had the highest probability prediction given out by the model. Hence, for this reason, during training, the true class label probability, that is the probability given by the model index for the actual target is learned via a series of layers. This in turn gives out the Confidence Score during evaluation. The proposed Confidence Network is quite simple. It is a five-layered Multilayer Perceptron that terminates into a single neuron giving an output score, as shown in Fig. 4b.

The detailed depiction of the model can be found in Table 1, with each module network classified for better re-implementation. The entire network is trained end-to-end with each of the terminal networks alternatively frozen when the other is being trained. As it is quite evident, the network's error rate was driven by two loss functions, one each for the terminal networks. For the classifier network, since, it was devised as a multiclass problem, the Cross-Entropy Loss (given in the following equation) was selected:

$$L(y, \hat{y}) = \frac{1}{m} \left(\sum_{i=1}^m y^{(i)} \log(\hat{y}^{(i)}) \right), \quad (2)$$

where y is the actual target label, \hat{y} is the predicted label by our network and m is the total number of samples in the dataset.

The confidence terminal network predicts a single value and is, therefore, aligned with the Mean Squared Error loss function (given in the following equation) to enhance its optimization:

$$L(y, \hat{y}) = \frac{1}{m} \sum_{i=1}^m (y^{(i)} - \hat{y}^{(i)})^2, \quad (3)$$

Introducing Alpha Trimmed Average Pooling

One of the highlights of our proposed network is through the novel pooling layer, named Alpha Trimmed Average Pooling. The X-ray images, in general, have a trend of partaking in salt and pepper noise. These noises have either very low values (in the case of pepper noise) or very high values (in the case of salt noise). As such, during the progression of this architecture, these artifacts do get incorporated by the convolutional layers. Utilizing the much-preferred Max-Pooling in this scenario diminishes the performance as a whole. As such, Alpha Trimmed Average Pooling is adopted which removes a percentage of the highest and the lowest values (after sorting) and averages the remaining ones to pop out an output. The percentage of values to be removed is taken as a hyperparameter, d .

Suppose there are k values and Alpha Trimmed Average Pooling are applied on them. First, a value is taken for d , and using that, it is decided how many values are to be removed from the k set of values using the following equation:

$$r = d\% * k. \quad (4)$$

After this, the $r/2$ highest and $r/2$ lowest values are removed from k , and then they are averaged using the following equation:

Table 1 Ablation study of CoWarriorNet

Module name	Blocks	Type/stride	Filter shape	Input size
Derived UNET Encoder	4 * (Identity Block + Pooling)	[Conv/1 (same padding, ReLU activation) BatchNorm Conv/1 (same padding, ReLU activation) BatchNorm], [Pool/2]	3 * 3 * 64 (conv) 2 * 2 (pool) 3 * 3 * 128 (conv) 2 * 2 (pool) 3 * 3 * 256 (conv) 2 * 2 (pool) 3 * 3 * 512 (conv) 2 * 2 (pool)	64 * 64 * 3
Derived UNET Bridge	1 * Identity Block	[Conv/1 (same padding, ReLU activation) BatchNorm Conv/1 (same padding, ReLU activation) BatchNorm]	3 * 3 * 1024	4 * 4 * 512
Derived UNET Decoder	4 * (Transposed Convolution + Concatenation + Identity Block)	[Conv/1 (same padding, ReLU activation) BatchNorm Conv/1 (same padding, ReLU activation) BatchNorm Conv/1 (same padding, ReLU activation) BatchNorm Conv/1 (same padding, ReLU activation) BatchNorm]	2 * 2 * 512 (conv_trans) 3 * 3 * 512 (conv) 2 * 2 * 256 (conv_trans) 3 * 3 * 256 (conv) 2 * 2 * 128 (conv_trans) 3 * 3 * 128 (conv) 2 * 2 * 64 (conv_trans) 3 * 3 * 64 (conv)	4 * 4 * 1024
Final Feature Map Concat with Original Input	1 * Convolution Layer	[Conv/1], [Concat with original input]	1 * 1 * 64	64 * 64 * 64
Residence module	2 * Residence Block	5 * [Conv/1 + BatchNorm]	3 * 3 * 64	6 * 64 * 2
Terminal Networks	Classifier + ConfidNet	[Classifier = Linear (512 → 3)] [ConfidNet = Linear (512 → 256), ReLU(), Linear (256 → 128), ReLU(), Linear (128 → 64), ReLU(), Linear (64 → 32), ReLU(), Linear (32 → 1)]	–	512 * 1

$$m = \frac{1}{(k - r)} \sum_{i=r/2}^{k-r/2} k_{\text{sorted}}(i), \tag{5}$$

where m is the output after the pooling layer. A much clearer pictorial depiction is given in Fig. 5 and an algorithmic depiction when using with convolutional layer is also given in Algorithm 1.

Algorithm *Begin*

- Step 1: Select a pool size of $(k * k)$.
- Step 2: Put this kernel on top of the image.
- Step 3: Pick the values of the image under the selected pool, $M = [m_1, m_2, m_3, \dots, m_{k^2}]$.
- Step 4: Sort M as M_{sorted} .
- Step 5: Pick a value for the hyperparameter d (in percentage).
- Step 6: Calculate $r = d\% * \text{len}(M_{\text{sorted}})$.
- Step 7: Remove $r/2$ values lowest values and $r/2$ highest values from M_{sorted} .
- Step 8: Calculate the output, m , using Eq. (5).

End

Experimental Results

This section ushers the light to the effective nature of our network via the gathered results in the detection of CoVID-19 through CXR images.

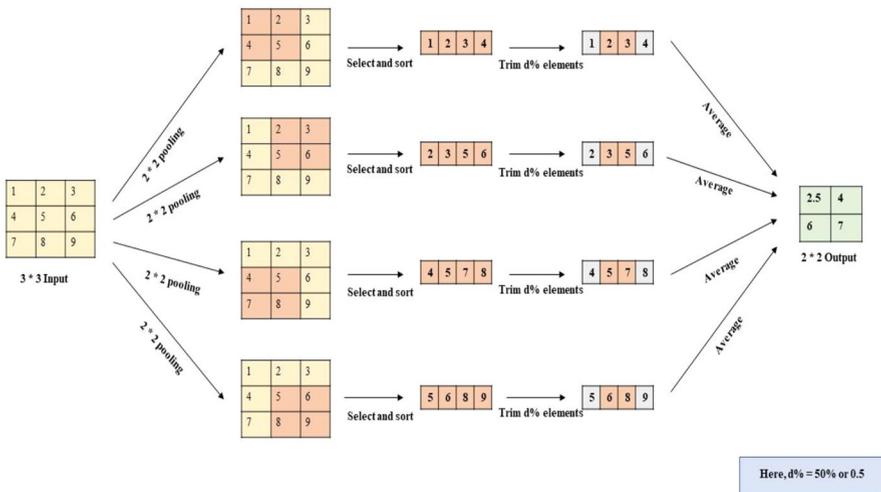


Fig. 5 Working principle of the proposed Alpha Trimmed Average Pooling

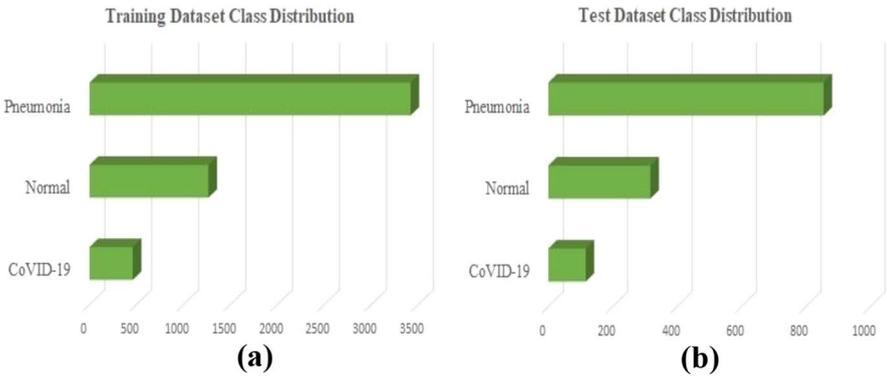


Fig.6 **a** Train dataset class distribution. **b** Test dataset class distribution

Table 2 Dataset description

Datasets	Total	Training			Testing		
		Normal	CoVID	Pneumonia	Normal	CoVID	Pneumonia
Kaggle	7432	3432	1264	492	812	246	98
Initiative	55	–	–	–	24	18	13
Synthetic	21,295	4321	3322	3310	4520	3122	3010

Implementation Details

The entire model is trained on Kaggle Kernel with accelerated GPU. The kernel built has a cumulative 13 GB RAM and 16 GB VRAM/GPU. The proposed model is fully running on a Jupiter Notebook environment with Python 3.7.10 and Pytorch version 1.7.

The model is tested and evaluated on the publicly available dataset obtained from Kaggle (Chest X-ray (CoVID-19 and Pneumonia) | Kaggle) [48]. The obtained dataset has a total of 5144 training data points and 1288 testing data points. The dataset is trifurcated into three classes, namely Normal, Pneumonia and CoVID-19. Besides that, CoVID-19 chest X-ray Dataset Initiative [49] and Synthetic CoVID-19 chest X-ray dataset [50] have also been utilized for the performance evaluation and comparative study. The distribution of the training and testing datasets (Kaggle) concerning the classes are depicted in Fig. 6a and b, respectively, and a detailed data description has been reported in Table 2. Due to the uneven distribution among classes, a batch-wise weight-based sampler is deduced which fed in images in batches that have even class distributions. The weight-based sampling and all the codes and results are documented in Python notebooks and will be made available once the paper has been accepted.

From Table 2, it can be concluded that the conducting experiment succeed to achieve the zero bias level as in the case of Kaggle dataset it is split into a 3:1 ratio, whereas for the synthetic dataset, it turns to a 1:1 ratio. Not only that, the Total

Initiative dataset has been used for testing purposes only. It also helps to reduce the prone of overfitting issues and make the proposed model the best fit for global acceptance.

Evaluation Metrics

The proposed model is evaluated based on the industry-norm metrics of Accuracy, Precision, Recall, and F1-Score. A detailed description of each of these is given as follows.

Accuracy

It refers to the total number of predictions given out correctly by the hypothesized model and is given by the following equation:

$$\text{Accuracy} = \frac{\text{len}(h_{\text{correct}})}{\text{len}(h_{\text{total}})}, \quad (6)$$

where h represents the prediction of our model, and $\text{len}(h_{\text{correct}})$ and $\text{len}(h_{\text{total}})$ refer to the number of correct predictions and the total number of positive predictions, respectively.

The accuracy curve of our proposed model can be seen in Fig. 7. The training accuracy has shown vivid increment in due course with final data showcasing a **97.8%** accuracy on board. On the validation set, even though there were initial fluctuations, it finally stabilized at around **93.05%**. The small difference is partly due to the small number of images regarded in the validation set. This reason becomes much more evident, as the model performs better in the test set compared to the validation set with an accuracy of **94.11%**.

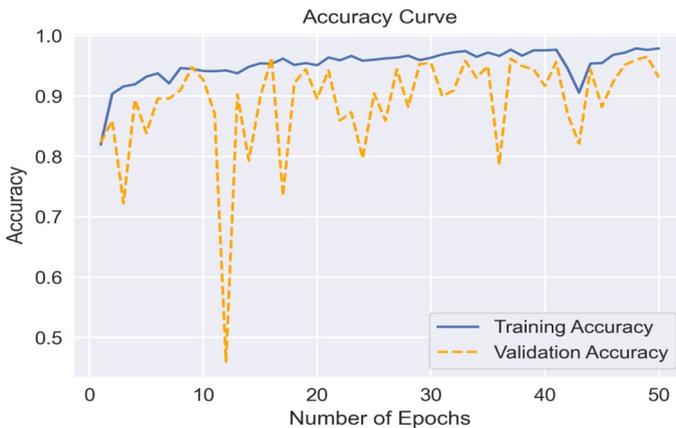


Fig. 7 Accuracy curve of the proposed model

Precision

Precision refers to the ratio of the correct number of positive predictions ($h_{class^i} == y_{class^i}$) given by the hypothesized model to the total number of positive responses predicted by the model (h_{class^i}), given in the following equation:

$$\text{Precision} = \frac{\text{len}(h_{class^i} == y_{class^i})}{\text{len}(h_{class^i})} \tag{7}$$

In the proposed case, due to multiple classes, the method is tried to gather the precision of each of the classes. The proposed method is articulated this by first picking out the prediction, then one at a time selecting one of the classes and setting it as a “positive” class, then seeking out precision value using Eq. (7).

The precision relative to this model predictions, in terms of each class, is given in Fig. 8. In each of the cases, the validation curve seems quite deviating but similar to the accuracy curve, the validation precision of each did shallow off being optimal. But, the most important aspect of the model of getting better results for covid did fetch fruitful outcomes with both the training and the validation sets capping off over **0.99** precision with regards to covid classes. The model showed its superiority in pneumonia cases too with a **0.97** and **0.94** precision in training and validation set, respectively. The precision for normal cases is a bit off the par in terms of the validation set score. Yet, it achieved quite a high **0.88** in the validation set and **0.97** in the training set. The test case result though fared highly, with covid, normal, and pneumonia precision showcasing values **0.97**, **0.94**, and **0.93**, respectively.

Recall/Sensitivity

Complementary to precision metric, Recall/Sensitivity refers to the ratio of the correct number of positive predictions ($h_{class^i} == y_{class^i}$) given by the hypothesized

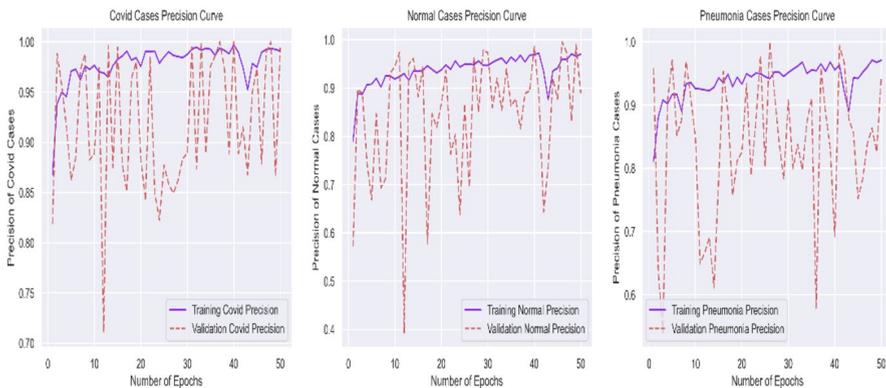


Fig. 8 Precision curve of the proposed model

model to the total number of actual positive responses in the dataset (y_{class^i}), given in the following equation:

$$\text{Recall/Sensitivity} = \frac{\text{len}(h_{class^i} == y_{class^i})}{\text{len}(y_{class^i})}. \tag{8}$$

The main objective coming to build this model is to get an appropriate sensitivity score. Based on such desires, the proposed model does fare amazingly well as shown in Fig. 9. The model in both the covid cases data point and the pneumonia cases data point in training and validation sets achieved values higher than **0.99**. On the other hand, when recognizing normal cases, the training sensitivity is mapped at **0.97**, but, in the validation set, it dropped down to **0.87**. This low grading does not bear much importance since the model is precisely created for the sole purpose of getting better Recall in covid cases. This is proven further as in the test set, this novel architecture devised a Recall of **0.93**, **0.83**, and **0.98** in covid, normal, and pneumonia cases, respectively.

F1-Score

There is a typical bias relative to an evaluation in Precision and Recall scores. As such, a far concrete evaluation metric, namely F1-Score, has been deciphered which considers both the values of Precision and Recall to give out a better evaluation score, as given in the following equation:

$$\text{F1 - score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{9}$$

In regards to this model, the F1-Score values with regard to each epoch are given in Fig. 10. As with the Recall and F1-Score, the values do flatten off at the end, citing optimal weights. The F1-Scores for covid cases lied above **0.99** for both the training and validation set—giving the objective quite a fruitful context. The normal case data points earned an F1-Score of **0.97** and **0.87** in training and validation sets,

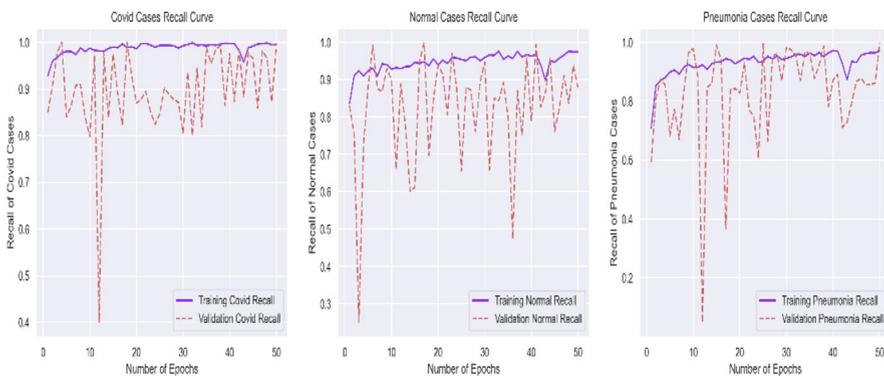


Fig. 9 Recall curve of the proposed model

respectively. Whereas, the pneumonia cases achieved an F1 score of **0.97** and **0.92** in train and validation sets, respectively. When put through the test set, the covid, normal and pneumonia cases achieved **0.95**, **0.88**, and **0.95** F1-Score, respectively.

Model Predictions: The Rights and the Wrongs for Kaggle Dataset

Our proposed model fetches quite a unique orientation when predicting hypothesized outputs. The two-terminal networks see to the higher so-called confidence of the model. The confidence network somewhat does a sanity check on the classification head by giving out a confidence value. The main motive is to make the model more confident, and with the results shown in Figs. 11 and 12, it can ascertain that notion.

In Fig. 11, one can see some of the correct predictions in the test set as predicted by the model. The figure also shows that the correct predictions bear a high confidence score (>0.98), given out by our model, indicating that the model has indeed learned quite a bit about the CoVID CXR Images.

On the other hand, as given in Fig. 12, some of the model predictions are quite wrong, to say the least. But, one can see that at that point of wrong predictions, the confidence scores are pretty low (<0.80) when compared to the scores of the correct predictions. These confidence scores give the reliability and the confidence to put out the method in real-time use since it helps to indicate the perfect instance for human intervention due to discrepancy by the model.

Performance Evaluation on the Synthetic Dataset

In the previous section, the performance of the Kaggle dataset has been obtained. However, to prove, the robustness of the developed architecture performance evaluation on the primary dataset is not enough. It has to be performed well on

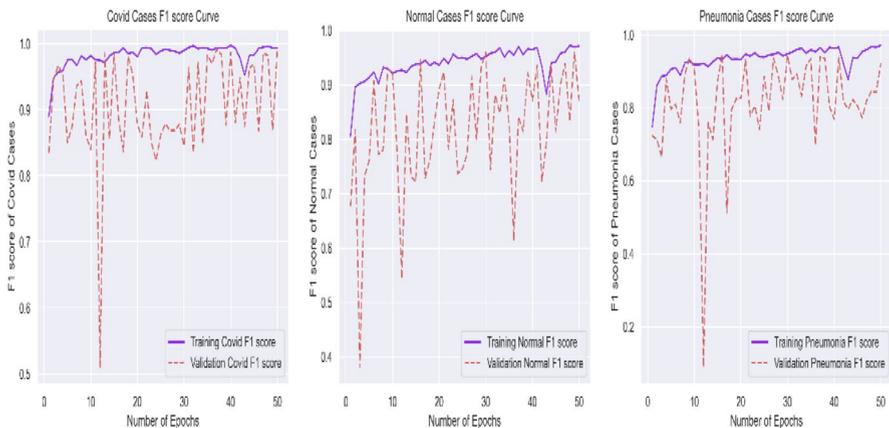


Fig. 10 The F1-Score curve of the proposed model

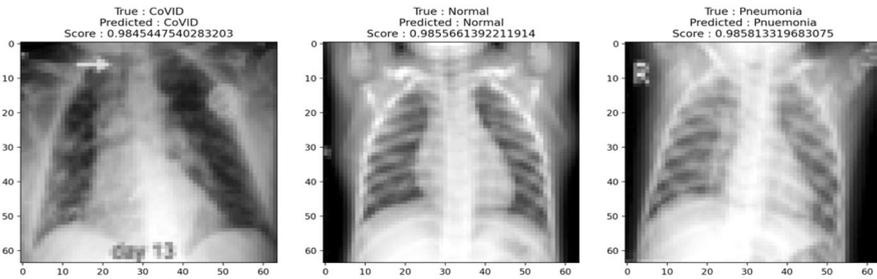


Fig. 11 Few samples of correct class labels are predicted by the proposed model, along with the respective confidence score

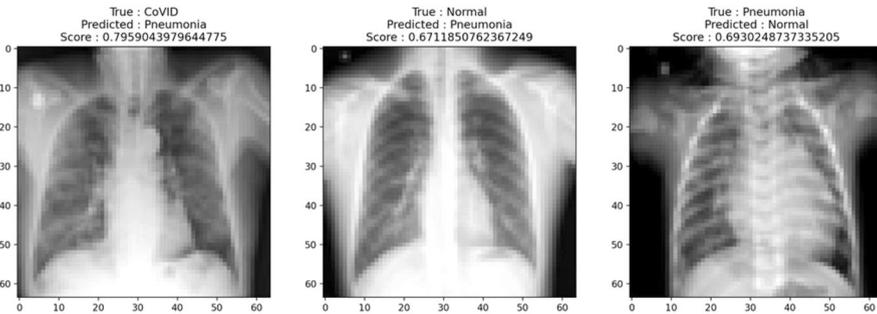


Fig. 12 Few samples of wrong class labels are predicted by the proposed model, along with the respective confidence score

synthetic datasets too as it contains images from various databases. Figure 13 gives a brief idea how a synthetic dataset looks like.

In Fig. 13, the first row represents the CoVID-19 chest X-ray images, whereas, the second and third rows represent viral pneumonia and normal images, respectively. Not only had that, to show the robustness, two-dimensional UMAP embeddings been obtained in Fig. 14.

From Fig. 14, it can be concluded that it is very difficult to get distribution for the CoVID-19 in UMAP as X-ray images are not relevant in detecting CoVID-19 which depicts the complexities of the problem that has been tackled by CoWarriorNet. Not only that, this paper has been generated a mixed class in a synthetic dataset without ground truth to show the effectiveness of the proposed architecture in the real-time scenario.

However, the proposed model obtained a higher accuracy (~98%) still some failure cases have been reported. To demonstrate, the region heatmap activation of all the three classes has been reported in Fig. 15.

From Fig. 15, it can be observed that there is a negligible difference among the three classes heatmaps, still failure cases are less than (~2%). This shows the necessity of the derived UNET and residence module as they performed the feature mapping very effectively. As all the analyses on the synthetic dataset have

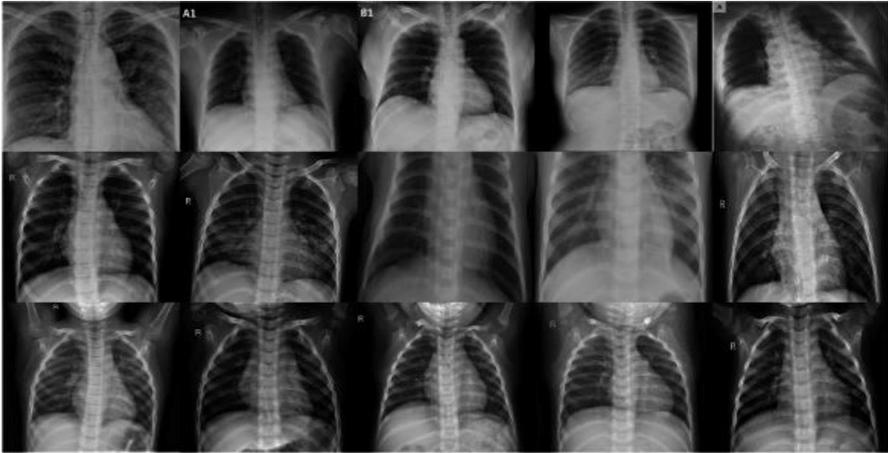


Fig. 13 Synthetic database

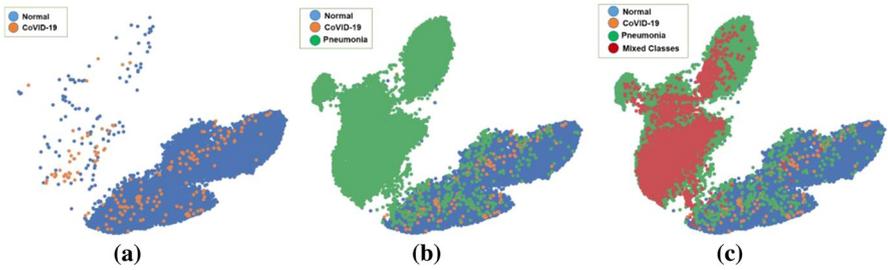


Fig. 14 Two-dimensional UMAP embeddings. **a** Normal vs CoVID-19, **b** Normal vs CoVID-19 vs Pneumonia, **c** Normal vs CoVID-19 vs Pneumonia vs Mixed Classes

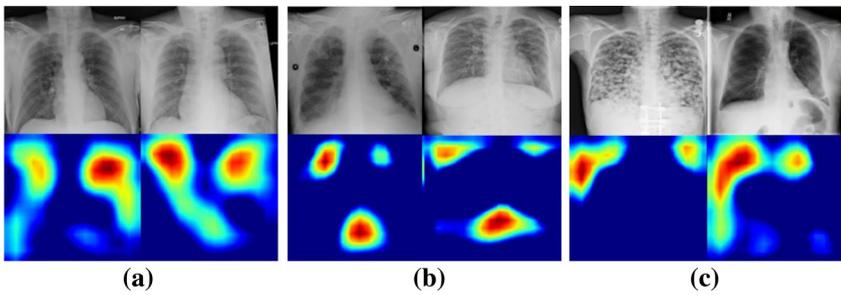


Fig. 15 Heatmap activation of CoWarriorNet: **a** CoVID-19, **b** Normal, **c** Pneumonia

been completed, the classification performance of the CoWarriorNet on the synthetic dataset is obtained in Fig. 16.

As depicted in Fig. 16, CoWarriorNet performs significantly well on the synthetic dataset too. Now, a detailed comparative study has been obtained in the following subsection to check whether the proposed architecture outperforms the existing ones or not.

A Comparative Study with the Other Architectures

Any study is quite incomplete without the comparison of the proposed architecture with other contemporary models. Here, the proposed framework has been compared with Ismael et al. [13], Abbas et al. [41], Minaee et al. [42], Wang et al. [43], and Tabik et al. [44] based on their confusion matrix and Precision, Recall, F1-Score of each of the three classes (Fig. 17).

From the above figure (Fig. 17), it can be concluded that the methods [13, 41–43] have suffered from overfitting whereas [44] goes underfitting. On the other hand, CoWarriorNet reduces the prone to overfitting. Now to prove its effectiveness the Precision, Recall, and F1-Score of the CoVID-19, Normal, and Pneumonia have been reported in Tables 3, 4, and 5, respectively.

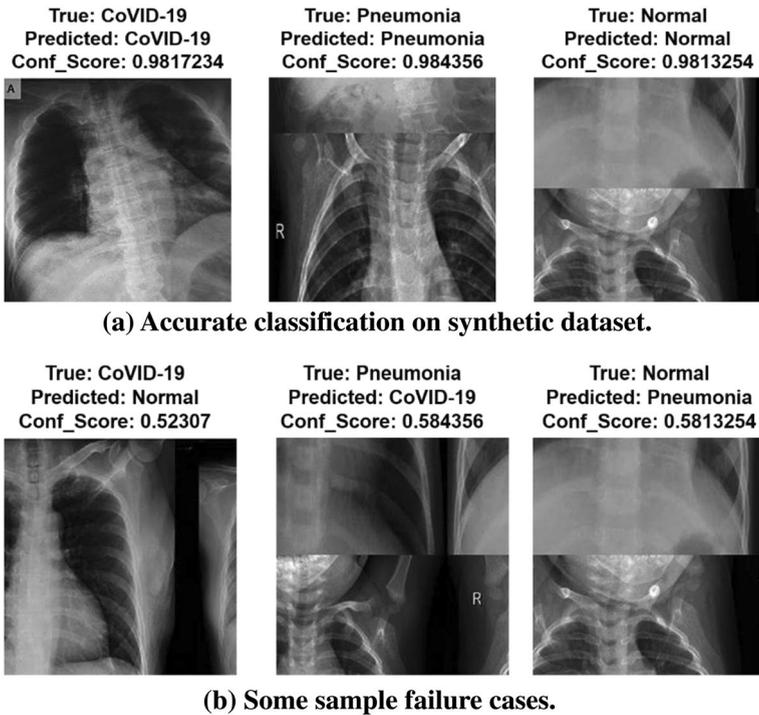


Fig. 16 Performance evaluation on synthetic dataset

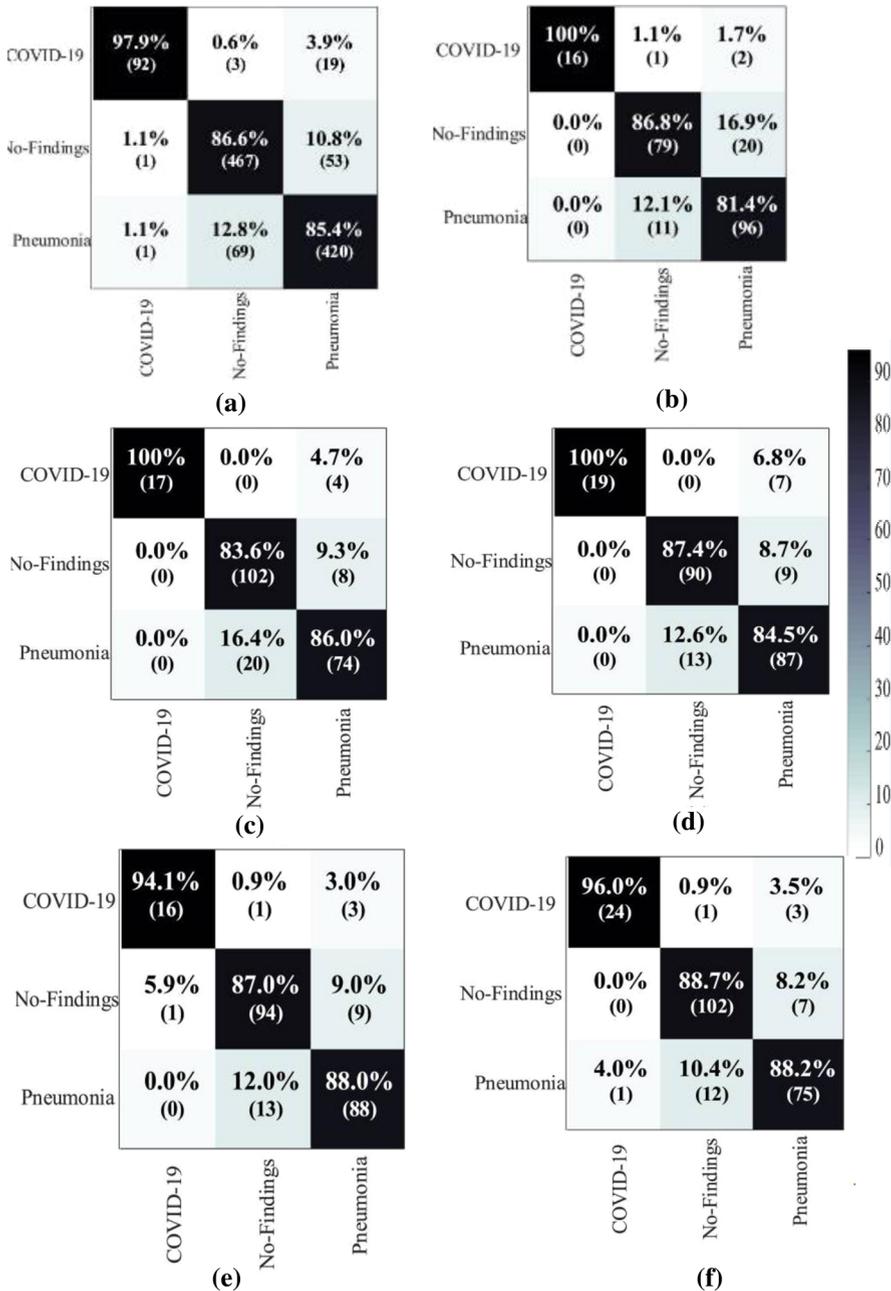


Fig. 17 The confusion matrix of a Ismael et al. [13], b Abbas et al. [41], c Minaee et al. [42], d Wang et al. [43], e Tabik et al. [44], and f CoWarriorNet on Kaggle data

Table 3 Comparative study of state-of-the-art approaches on CoVID-19 classes

Methodology	Kaggle			Initiative			Synthetic		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Ismael et al. [13]	0.84	0.82	0.83	0.85	0.83	0.84	0.85	0.83	0.93
Abbas et al. [41]	0.83	0.79	0.81	0.84	0.80	0.82	0.84	0.89	0.84
Minaee et al. [42]	0.84	0.79	0.82	0.85	0.80	0.83	0.85	0.89	0.87
Wang et al. [43]	0.86	0.81	0.83	0.87	0.82	0.84	0.87	0.91	0.85
Tabik et al. [44]	0.85	0.83	0.84	0.86	0.84	0.85	0.86	0.92	0.88
CoWarrior-Net	0.97	0.93	0.95	0.98	0.94	0.96	0.98	0.93	0.95

Table 4 Comparative study of state-of-the-art approaches on Normal classes

Methodology	Kaggle			Initiative			Synthetic		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Ismael et al. [13]	0.82	0.74	0.78	0.83	0.75	0.79	0.85	0.76	0.80
Abbas et al. [41]	0.85	0.75	0.80	0.86	0.76	0.81	0.87	0.77	0.82
Minaee et al. [42]	0.81	0.79	0.80	0.82	0.80	0.81	0.83	0.81	0.82
Wang et al. [43]	0.84	0.77	0.80	0.85	0.78	0.81	0.86	0.79	0.83
Tabik et al. [44]	0.87	0.79	0.83	0.88	0.84	0.84	0.89	0.81	0.84
CoWarrior-Net	0.94	0.83	0.88	0.95	0.84	0.89	0.96	0.83	0.90

In Table 3, the CoWarriorNet always performs better than the state-of-the-art approaches as they are mainly trained on RGB images whereas X-ray images are mainly greyscale that is why their feature information does not matche with the RGB ones.

As depicted in Table 4, CoWarriorNet easily outperforms the existing ones as they are using the pre-trained model with the help of transfer learning whereas CoWarriorNet has been trained from scratch without any help from auxiliary models.

As CoWarriorNet outperforms the state-of-the-art approaches in Precision and Recall, it is obvious that it is outperformed in F1-Score too.

From the above discussion, it can be concluded that the proposed methodology provides more optimized statistically accurate results in a very lesser time as compared to the state-of-the-art approaches. It has the potential to overcome the

Table 5 Comparative study of state-of-the-art approaches on Pneumonia classes

Methodology	Kaggle			Initiative			Synthetic		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Ismael et al. [13]	0.86	0.81	0.83	0.87	0.82	0.84	0.88	0.84	0.86
Abbas et al. [41]	0.85	0.83	0.84	0.86	0.84	0.85	0.87	0.86	0.86
Minaee et al. [42]	0.85	0.83	0.84	0.86	0.84	0.85	0.87	0.86	0.87
Wang et al. [43]	0.86	0.84	0.85	0.87	0.85	0.86	0.88	0.87	0.88
Tabik et al. [44]	0.87	0.85	0.86	0.88	0.86	0.87	0.89	0.88	0.87
CoWarrior-Net	0.93	0.98	0.95	0.94	0.98	0.96	0.96	0.98	0.96

drawbacks highlighted in the literature review and create its pathway by making a notable advancement in the classification of CoVID-19 chest X-ray images.

Conclusion and Future Scope

Through this paper, we have tried to build upon a new deep convolutional network that can detect CoVID-19 from chest X-ray images. Apart from a novel architecture, the paper also introduces a new pooling layer which has contributed to the excellent result that the model has shown. The future scope of the line of work can be in two ways.

The first is based on the advancement of further deep learning works to detect CoVID-19 with the proposed model being the platform. This can be done with a much better quality of images or through the artificial generation of a much diverse set of images. Furthermore, in our work, pre-processing is quite low. In the future, the researchers can try to implement newer pre-processing to fetch better outcomes. Apart from that, the attention-based models on top of the novel architecture can be visited to see the difference in the outcomes.

The second is related to the use of this network module in other sectors of study. This novel architecture can be utilized in other unknown areas of classification and detection. The novel alpha trimmed average pooling can be utilized in other areas of study to mark its importance. All in all, the proposed model has performed significantly well in detecting CoVID-19 cases from chest X-ray images and does provide a new benchmark for future works.

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1007/s00354-021-00143-1>.

Acknowledgements We would like to thank and dedicate this paper to all the health workers around the world who have spent their hearts and soul in combating this deadly disease. Without their selfless effort,

most of us would not have been here alive. As such, we would like to thank each one of the health workers from the bottom of our hearts.

References

- Zheng, J.: SARS-CoV-2: an emerging coronavirus that causes a global threat. *Int J Biol Sci* **16**(10), 1678–1685 (2020). <https://doi.org/10.7150/ijbs.45053>
- Wang, W., Xu, Y., Gao, R., Lu, R., Han, K., Wu, G., Tan, W.: Detection of SARS-CoV-2 in different types of clinical specimens. *JAMA* **323**(18), 1843–1844 (2020)
- West, C.P., Montori, V.M., Sampathkumar, P.: COVID-19 testing: the threat of false-negative results. *Mayo Clin Proc* **95**(6), 1127–1129 (2020). (**Elsevier**)
- Scohy, A., Anantharajah, A., Bodéus, M., Kabamba-Mukadi, B., Verroken, A., Rodriguez-Villalobos, H.: Low performance of rapid antigen detection test as frontline testing for COVID-19 diagnosis. *J Clin Virol* **129**, 104455 (2020)
- Kanne, J.P., Little, B.P., Chung, J.H., Elicker, B.M., Ketai, L.H.: Essentials for radiologists on COVID-19: an update—radiology scientific expert panel. *Radiology* **296**(2), E113–E114 (2020)
- Xie, X., Zhong, Z., Zhao, W., Zheng, C., Wang, F., Liu, J.: Chest CT for typical coronavirus disease 2019 (COVID-19) pneumonia: relationship to negative RT-PCR testing. *Radiology* **296**(2), E41–E45 (2020)
- Buzug, T.M.: Computed tomography: In Springer handbook of medical technology, pp. 311–342. Springer, Berlin (2011)
- Mangal, A., Kalia, S., Rajgopal, H., Rangarajan, K., Nambodiri, V., Banerjee, S. and Arora, C.: CovidAID: COVID-19 detection using chest X-ray. arXiv preprint (2020). <http://arxiv.org/abs/2004.09803>.
- Fang, Y., et al.: Sensitivity of chest CT for COVID-19: comparison to RT-PCR. *Radiology* (2020). <https://doi.org/10.1148/radiol.2020200432>
- Ali, R.M.M., Ghonimy, M.B.I.: Post-COVID-19 pneumonia lung fibrosis: a worrisome sequelae in surviving patients. *Egypt. J. Radiol. Nuclear Med.* **52**(1), 1–8 (2021)
- Hussain, E., Hasan, M., Rahman, M.A., Lee, I., Tamanna, T., Parvez, M.Z.: CoroDet: a deep learning-based classification for COVID-19 detection using chest x-ray images. *Chaos Solit. Fractals* **142**, 110495 (2021)
- Madaan, V., Roy, A., Gupta, C., et al.: XCOVNet: Chest x-ray image classification for COVID-19 early detection using convolutional neural networks. *New Gener. Comput.* (2021). <https://doi.org/10.1007/s00354-021-00121-7>
- Ismael, A.M., Şengür, A.: Deep learning approaches for COVID-19 detection based on chest x-ray images. *Expert Syst. Appl.* **164**, 114054 (2021)
- Banerjee, A., Pati, S.K.: Predicting antiviral drugs for COVID-19 treatment using artificial intelligence based approach. In: Nayak, J., Naik, B., Abraham, A. (eds.) *Understanding COVID-19: the role of computational intelligence*. Studies in computational intelligence, vol. 963. Springer, Cham (2022). https://doi.org/10.1007/978-3-030-74761-9_11
- Basu, S., Mitra, S. and Saha, N.: Deep learning for screening COVID-19 using chest x-ray images. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*. pp. 2521–2527 (2020). IEEE
- Dash, S., Chakravarty, S., Mohanty, S.N., et al.: A Deep Learning Method to Forecast COVID-19 Outbreak. *New Gener. Comput.* (2021). <https://doi.org/10.1007/s00354-021-00129-z>
- Jain, R., Gupta, M., Taneja, S., Hemanth, D.J.: Deep learning-based detection and analysis of COVID-19 on chest x-ray images. *Appl. Intell.* **51**(3), 1690–1700 (2021)
- Nayak, S.R., Nayak, D.R., Sinha, U., Arora, V., Pachori, R.B.: Application of deep learning techniques for detection of COVID-19 cases using chest x-ray images: a comprehensive study. *Biomed. Signal Process. Control* **64**, 102365 (2021)
- Hemdan, E.E.D., Shouman, M.A. and Karar, M.E.: Covidx-net: A framework of deep learning classifiers to diagnose COVID-19 in x-ray images. arXiv preprint (2020). <http://arxiv.org/abs/2003.11055>.

20. Toğaçar, M., Ergen, B., Cömert, Z.: COVID-19 detection using deep learning models to exploit Social Mimic Optimization and structured chest x-ray images using fuzzy color and stacking approaches. *Comput. Boil. Med.* **121**, 103805 (2020)
21. Ouchicha, C., Ammor, O., Meknassi, M.: CVDNet: a novel deep learning architecture for detection of coronavirus (COVID-19) from chest x-ray images. *Chaos Solit. Fractals* **140**, 110245 (2020)
22. Bianco, S., Cadene, R., Celona, L., Napoletano, P.: Benchmark analysis of representative deep neural network architectures. *IEEE Access* **6**, 64270–64277 (2018)
23. Chaudhary, Y., Mehta, M., Sharma, R., Gupta, D., Khanna, A., and Rodrigues, J.J.: Efficient-Covid-Net: deep learning-based COVID-19 detection from chest x-ray images. In 2020 IEEE International Conference on E-health Networking, Application & Services (HEALTHCOM), pp. 1–6 (2021)
24. Tang, S., Wang, C., Nie, J., Kumar, N., Zhang, Y., Xiong, Z., Barnawi, A.: EDL-COVID: ensemble deep learning for COVID-19 cases detection from chest x-ray images. *IEEE Trans. Ind. Inform.* **17**(9), 6539–6549 (2021)
25. Calderon-Ramirez, S., Yang, S., Moemeni, A., Colreavy-Donnelly, S., Elizondo, D.A., Oala, L., Rodríguez-Capitán, J., Jiménez-Navarro, M., López-Rubio, E., Molina-Cabello, M.A.: Improving uncertainty estimation with semi-supervised deep learning for COVID-19 detection using chest x-ray images. *IEEE Access* **9**, 85442–85454 (2021)
26. Jain, G., Mittal, D., Thakur, D., Mittal, M.K.: A deep learning approach to detect COVID-19 coronavirus with x-ray images. *Biocybernet. Biomed. Eng.* **40**(4), 1391–1405 (2020)
27. Haghanifar, A., Majdabadi, M.M., Choi, Y., Deivalakshmi, S. and Ko, S.: Covid-cxnet: Detecting COVID-19 in frontal chest x-ray images using deep learning. arXiv preprint (2020). <http://arxiv.org/abs/2006.13807>.
28. Luz, E., Silva, P., Silva, R., Silva, L., Guimarães, J., Miozzo, G., Moreira, G., Menotti, D.: Towards an effective and efficient deep learning model for COVID-19 patterns detection in X-ray images. *Res. Biomed. Eng.* (2021). <https://doi.org/10.1007/s42600-021-00151-6>
29. Karakanis, S., Leontidis, G.: Lightweight deep learning models for detecting COVID-19 from chest x-ray images. *Comput. Biol. Med.* **130**, 104181 (2021)
30. Ibrahim, A.U., Ozsoz, M., Serte, S., Al-Turjman, F., Yakoi, P.S.: Pneumonia classification using deep learning from chest X-ray images during COVID-19. *Cogn. Comput.* (2021). <https://doi.org/10.1007/s12559-020-09787-5>
31. Calderon-Ramirez, S., Giri, R., Yang, S., Moemeni, A., Umana, M., Elizondo, D., Torrents-Barrena, J. and Molina-Cabello, M.A.: Dealing with scarce labelled data: Semi-supervised deep learning with mix match for COVID-19 detection using chest x-ray images. In 2020 25th International Conference on Pattern Recognition (ICPR), pp. 5294–5301 (2021)
32. Sakib, S., Tazrin, T., Fouda, M.M., Fadlullah, Z.M., Guizani, M.: DL-CRC: deep learning-based chest radiograph classification for COVID-19 detection: a novel approach. *IEEE Access* **8**, 171575–171589 (2020)
33. Panwar, H., Gupta, P.K., Siddiqui, M.K., Morales-Menendez, R., Bhardwaj, P., Singh, V.: A deep learning and grad-CAM based color visualization approach for fast detection of COVID-19 cases using chest x-ray and CT-Scan images. *Chaos Solit. Fractals* **140**, 110190 (2020)
34. Kamal, K.C., Yin, Z., Wu, M., Wu, Z.: Evaluation of deep learning-based approaches for COVID-19 classification based on chest X-ray images. *Signal Image Video Process SIVIP* **15**, 959–966 (2021). <https://doi.org/10.1007/s11760-020-01820-2>
35. Singh, G., Yow, K.C.: An interpretable deep learning model for COVID-19 detection with chest x-ray images. *IEEE Access* **9**, 85198–85208 (2021)
36. Ronneberger, O., Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation. In International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 234–241. Springer, Cham (2015)
37. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. *Adv. Neural. Inf. Process. Syst.* **25**, 1097–1105 (2012)
38. Ioffe, S. and Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning. pp. 448–456 (2015).
39. Hsiao, T.Y., Chang, Y.C., Chou, H.H., Chiu, C.T.: Filter-based deep-compression with global average pooling for convolutional networks. *J. Syst. Architect.* **95**, 9–18 (2019)
40. Corbière, C., Thome, N., Bar-Hen, A., Cord, M. and Pérez, P.: Addressing failure prediction by learning model confidence. arXiv preprint (2019). <http://arxiv.org/abs/1910.04851>.
41. Abbas, A., Abdelsamea, M.M., Gaber, M.M.: Classification of COVID-19 in chest x-ray images using DeTraC deep convolutional neural network. *Appl. Intell.* **51**(2), 854–864 (2021)

42. Minaee, S., Kafieh, R., Sonka, M., Yazdani, S., Soufi, G.J.: Deep-COVID: predicting COVID-19 from chest x-ray images using deep transfer learning. *Med. Image Anal.* **65**, 101794 (2020)
43. Wang, L., Lin, Z.Q., Wong, A.: COVID-net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest x-ray images. *Sci. Rep.* **10**(1), 1–12 (2020)
44. Tabik, S., Gómez-Ríos, A., Martín-Rodríguez, J.L., Sevillano-García, I., Rey-Area, M., Charte, D., Guirado, E., Suárez, J.L., Luengo, J., Valero-González, M.A., García-Villanova, P.: COVIDGR dataset and COVID-SDNet methodology for predicting COVID-19 based on Chest X-Ray images. *IEEE J. Biomed. Health Inform.* **24**(12), 3595–3605 (2020)
45. Gomes, J.C., Barbosa, V.A.F., Santana, M.A., et al.: IKONOS: an intelligent tool to support diagnosis of COVID-19 by texture analysis of x-ray images. *Res. Biomed. Eng.* (2020). <https://doi.org/10.1007/s42600-020-00091-7>
46. Gomes, J.C., Masood, A.I., Silva, L.H.S., et al.: COVID-19 diagnosis by combining RT-PCR and pseudo-convolutional machines to characterize virus sequences. *Sci. Rep.* **11**, 11545 (2021). <https://doi.org/10.1038/s41598-021-90766-7>
47. Ismael, A.M., Şengür, A.: The investigation of multiresolution approaches for chest x-ray image based COVID-19 detection. *Health Inf. Sci. Syst.* **8**, 29 (2020). <https://doi.org/10.1007/s13755-020-00116-6>
48. COVID-19 Radiography Database. <https://www.kaggle.com/tawsifurrahman/covid19-radiography-database>, Accessed 2 July 2020.
49. COVID-19 Chest X-Ray Dataset Initiative. <https://github.com/agchung/figure1-covid-chestxray-dataset>. Accessed 2 July 2020.
50. Zunair, H. and Hamza, A.B.: Synthetic COVID-19 Chest X-ray Dataset for Computer-Aided Diagnosis. *arXiv preprint* (2021). <http://arxiv.org/abs/2106.09759>.

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

Authors and Affiliations

Indrani Roy¹ · Rinita Shai² · Arijit Ghosh¹ · Anirban Bej¹ · Soumen Kumar Pati³ 

Indrani Roy
indraniroy1910@gmail.com

Rinita Shai
rinitashai15@gmail.com

Arijit Ghosh
sohonjit.ghosh@gmail.com

Anirban Bej
anirbanbej8@gmail.com

¹ Department of Electronics and Communication Engineering, Calcutta Institute of Engineering and Management, Kolkata 700040, WB, India

² Department of Mathematics, Behala College, Calcutta University, Kolkata 700060, WB, India

³ Department of Bioinformatics, Maulana Abul Kalam Azad University of Technology, Haringhata 741249, WB, India