Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

Uncertainty-aware convolutional neural network for COVID-19 X-ray images classification

Mahesh Gour [*], Sweta Jain

Maulana Azad National Institute of Technology, Bhopal, MP, 462003, India

ARTICLE INFO

ABSTRACT

Deep learning (DL) has shown great success in the field of medical image analysis. In the wake of the current pandemic situation of SARS-CoV-2, a few pioneering works based on DL have made significant progress in automated screening of COVID-19 disease from the chest X-ray (CXR) images. But these DL models have no inherent way of expressing uncertainty associated with the model's prediction, which is very important in medical image analysis. Therefore, in this paper, we develop an uncertainty-aware convolutional neural network model, named UA-ConvNet, for the automated detection of COVID-19 disease from CXR images, with an estimation of associated uncertainty in the model's predictions. The proposed approach utilizes the EfficientNet-B3 model and Monte Carlo (MC) dropout, where an EfficientNet-B3 model has been fine-tuned on the CXR images. During inference, MC dropout has been applied for M forward passes to obtain the posterior predictive distribution. After that mean and entropy have been calculated on the obtained predictive distribution to get the mean prediction and model uncertainty. The proposed method is evaluated on the three different datasets of chest X-ray images, namely the COVID19CXr, X-ray image, and Kaggle datasets. The proposed UA-ConvNet model achieves a G-mean of 98.02% (with a Confidence Interval (CI) of 97.99–98.07) and sensitivity of 98.15% for the multi-class classification task on the COVID19CXr dataset. For binary classification, the proposed model achieves a G-mean of 99.16% (with a CI of 98.81–99.19) and a sensitivity of 99.30% on the X-ray Image dataset. Our proposed approach shows its superiority over the existing methods for diagnosing the COVID-19 cases from the CXR images.

1. Introduction

Novel Corona Virus disease 2019 (COVID-19) has originated from Wuhan, the Hubei province of China, has spread across the globe and put the livelihood of people in danger [1]. The severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) [2] or COVID-19 disease created a pandemic situation world wide. The primary symptoms of Coronavirus are sore throat, headache, fatigue, muscle pain, fever, dry cough, and breathing difficulty. In some cases, it causes severe respiratory problems such as Pneumonia, lung infections, and kidney failure [3]. This disease is more vulnerable to people who have high blood pressure, heart problems, and diabetes. COVID-19 is a contagious disease, which can be easily transmitted from an infected person to a normal one when the infected person is sneezing or coughing.

This pandemic has brought new challenges to the medical world. People were not getting wards, ventilators, and even a shortage of doctors and nurses in hospitals. A crucial step in combating COVID-19 is

to recognize the infected people so that they are treated promptly and isolated to prevent widespread infection. The most common and gold standard technique used for diagnosing COVID-19 is the RT-PCR [4]. But, it has concerns like poor sensitivity [5,6], reliability and performance. Recently, radiological diagnostic techniques such as computed tomography (CT scan) and X-ray have been used to diagnose COVID-19, which are faster and more accurate [7,8]. As shown in Fig. 1, COVID-19 manifestations in chest X-ray (CXR) images are similar to other viral pneumonia, making it challenging to distinguish manually. Hence, a reliable Computer-Aided Diagnosis system (CADs) for the COVID-19 detection from the chest X-ray (CXR) images can be beneficial for the rapid screening of the infected patient.

Nowadays, deep learning (DL) has become very popular in the medical world to detect and diagnose diseases automatically [9–14]. Recently, many researchers developed DL-based models (or CAD systems) for the COVID-19 detection from the radiological images and have achieved outstanding performances. DL-based models have apparent

---
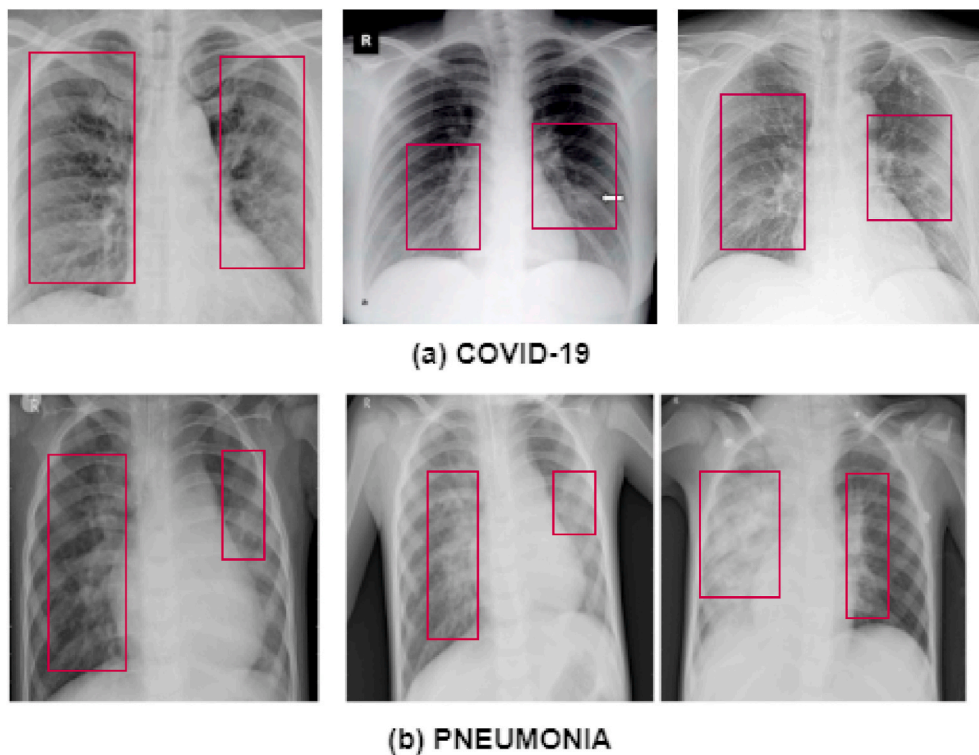
Fig. 1. Examples of (a) COVID-19 and (b) Viral Pneumonia in X-ray images as shown in the top and the bottom row, respectively. Rectangle markers represent findings (disease patterns) in the CXR images.

benefits of increased performance, but they do not have intrinsic means to express uncertainty associated with the model's prediction [15,16]. Since automatic COVID-19 disease detection is a field where it is of crucial importance to represent the model uncertainty for reliable and safe diagnosis. Any wrong diagnosis can be very dangerous to the patient and society. Hence, in this study, we have developed an uncertainty-aware convolutional neural network (CNN) model referred to as UA-ConvNet that detects COVID-19 disease from X-ray images, with an estimation of associated uncertainty in model's predictions. Our main contributions are as follows:

- We incorporate the recent advances in the field of DL to develop a DL model that detects COVID-19 from the CXR images.
- We demonstrate how uncertainty can be modeled in the CAD system for COVID-19 detection from the CXR images.
- We add Monte Carlo (MC) dropout with the EfficientNet-B3 network for making it Bayesian. We estimate the model uncertainty by modeling the posterior predictive distribution of M predictions of the model.
- This framework provides a generic error estimator and demonstrates how much the model is confident in classifying the given X-ray images. Results show that our model has high confidence in decision-making in order to classify the CXR images.
- We also investigate the performances of pre-trained networks, namely, VGG19 [17], ResNet [18], InceptionV3 [19], DenseNet [20], Xception [21], MobileNet [22] and NASNetLarge [23], for COVID-19 disease detection from CXR images.

The remaining paper is structured as follows: Section 2 presents the related work. Section 3 describes the methods used and the proposed UA-ConvNet model. Section 4 details the COVID19CXr, X-ray image and Kaggle datasets and describes the experimental results and performance comparison. Section 5 concludes this study.

2. Related work

DL based CAD systems have proven to be successful in lung disease detection and classification from radiological images [24,25]. It is being used in medical practice to assist in the diagnosis of disease. Motivated by these successes, several CAD systems have recently been developed to automatically diagnosis COVID-19 disease using X-ray images [26,27]. Narin et al. [28] have proposed a method based on transfer learning, in which three pre-trained models, namely ResNet50, InceptionV3, and Inception-ResNetV2, have been used to detect COVID-19 disease using CXR images. They have achieved the best classification accuracy of 98% with the ResNet50 model. Zhang et al. [29] have proposed a two-stage transfer learning-based approach for detecting COVID-19, in which first they have fine-tuned ResNet34 on a large X-ray image dataset; after that, they have further fine-tuned it on a small dataset of COVID-19 CXR images. Their proposed framework achieved an accuracy of 91.08%. Ardakani et al. [30] have tested ten advanced CNN models for COVID-19 detection using CT images. The authors reported the highest accuracy of 99.51% for the ResNet101 CNN model.

CNN models require a large amount of data for training, but the COVID-19 pandemic has arisen recently, so limited CXR images data is available. To generate more X-rays images, Waheed et al. [31] have presented an Auxiliary Classifier Generative Adversarial Network (ACGAN) model for generating synthetic images for COVID-19 diagnosis. Their proposed model has achieved a classification accuracy of 95%. Wang et al. [32] have developed a model by combining two 3D-ResNets for categorizing chest CT images into COVID-19 and interstitial lung disease via a prior-attention strategy. Their model has extended residual learning with the prior-attention mechanism and achieved an accuracy of 93.3%. Togacar et al. [33] have used image processing techniques like fuzzy color and stacking. They have extracted deep features by using MobileNetV2 and SqueezeNet from the X-ray images. Further, SVM has applied on the features to classify the images into COVID-19, Normal, and Pneumonia classes. Their method achieved an accuracy of 99.27%.

A patch-based CNN architecture for COVID-19 diagnosis has been proposed by Oh et al. [34]. They have applied majority voting strategy on the decisions obtained from various patches for making the final decision. They have reported an accuracy of 91.9%. Gour and Jain [35] have developed a stacked ensemble-based approach for the CXR image classification into COVID-19, Pneumonia, and Normal classes, in which they have extracted five sub-models from the VGG19 model and proposed CovNet30 model. Finally, logistic regression has been applied to the outputs of the sub-models. Their proposed approach achieved a classification accuracy of 92.74%. Similarly, Autee et al. [36] have proposed an ensemble-based model for COVID-19 X-ray image classification, in which four different CNN models have been combined. Their proposed approach achieved an accuracy of 95.07%. Gianchandani et al. [37] have developed two ensemble models using VGG16, ResNet152, and DenseNet201 networks for the classification of CXR images into binary and multiple classes. Their ensemble model achieved an accuracy of 96.15% and 99.21% for multi-class and binary classification tasks, respectively. Wang et al. [38] have designed a deep learning-based model called COVID-Net to detect the COVID-19 cases from the CXR images. The authors have reported accuracy of 93.3%.

A 2D deep learning architecture with a U-Net model has been proposed to segment the lung spaces, and COVID-19 anomalies from chest CT scans by Kuchana et al. [39]. They reported a mean IOU (intersection over the union) of 84.6%. Similarly, Zhao et al. [40] have developed a D2A U-Net model using the dual attention strategy, and hybrid dilated convolutions for the segmentation of COVID-19 lesion in the CT scan images. Their method achieved a Dice score of 0.7298. Ozturk et al. [41] have proposed the DarkcovNet model for automatic COVID-19 detection using CXR images, in which DarkNet19 is used as a classifier for the YOLO object detection system. They reported a classification accuracy of 98.08%. Mishra et al. [42] have proposed a method based on VGG16 and ResNet50 using transfer learning for COVID-19 detection from the CT images. Their method achieved an accuracy of 86.74% and 88.52% for VGG16 and ResNet50, respectively. Maheshwari et al. [43] have extracted local binary pattern (LBP) based and image-based features from the CXR images and applied support vector machine (SVM) on the extracted features to classify them into Normal, Pneumonia, and COVID-19 classes. The authors reported a sensitivity of 97.86% for CXR image classification. Shamsi et al. [44] have proposed an uncertainty-aware framework using transfer learning for COVID-19 detection from X-ray and CT images. In their proposed framework, different pre-trained CNNs have been used for extracting deep features from the CXR and CT images. Furthermore, they have applied various machine learning and statistical modeling methods for COVID-19 diagnosis and measuring the epistemic uncertainty of classification results. Calderon-Ramirez et al. [45] have proposed MixMatch semi-supervised framework for improving uncertainty estimation of model for COVID-19 detection. They have tested uncertainty estimation techniques like softmax scores, deterministic uncertainty quantification and MC dropout. Similarly, Dong et al. [46] have proposed a deep learning framework named RCoNetk for COVID-19 detection and uncertainty estimation. In which, they have utilized Deformable Mixed High-order Moment Feature, Mutual Information Maximization, and Multiexpert Uncertainty-aware learning. Their proposed RCoNetK model achieved a sensitivity of 97.76% for X-ray images classification into COVID-19, Pneumonia, and Normal classes.

All these studies have reported promising results for the CNN model to detect COVID-19 from chest X-ray images. But these models have trained on small-sized datasets. In contrast, deep learning models require massive data for training hundreds and thousands of learnable parameters. Thus, limited training data may lead to epistemic uncertainty [47]. The majority of existing approaches [28–43] have not provided information about the uncertainty associated with their predictions. Only a few studies [44–46] have provided their model uncertainty information. Whereas an uncertainty estimation of the model is very important in the medical domain for a reliable and safe CAD system. Motivated by these shortcomings, we develop a reliable uncertainty-aware CNN model for COVID-19 disease detection from chest X-ray images.

3. Methodology

This section introduces the proposed uncertainty-aware CNN model and describes the methods utilized in this study. First, we discuss the details of the pre-trained EfficientNet model and MC dropout in Section 3.1 and Section 3.2 respectively. Next, we present our proposed UA-ConvNet model, in which we will explain how the uncertainty is incorporated in the EfficientNet-B3 model in Section 3.3.

3.1. EfficientNet

The EfficientNets [48] are a family of pre-trained CNN models (EfficientNet-B0 to EfficientNet-B7), which are trained on the ImageNet dataset [49]. The main idea used in the EfficientNet model is that the network's performance and efficiency can be improved by scaling up the network's dimensions such as depth *(D)*, width *(W)*, and resolution *(R)*. Therefore, Tan et al. [48] have proposed a compound scaling method in which they used compound coefficient θ to uniformly scale up the network's dimensions, as shown in Eq. (1).

$$D = d^{\theta}, W = w^{\theta}, R = r^{\theta}$$
$$\text{s.t. } d \cdot w^2 \cdot r^2 \approx 2 \tag{1}$$
$$d \geq 1, w \geq 1, r \geq 1$$

Where *d*, *w*, and *r* are the constants that have been determined using grid search. EfficientNet-B1 to B7 have been obtained by scaling up the base model EfficientNet-B0 with different θ and keeping *d, w, r* as constants in Eq. (1). For image recognition, EfficientNet-B7 achieved state-of-the-art 84.4% top-1 accuracy, and it is 8.4x smaller and 6.1x faster than the best existing GPipe [50]. EfficientNet models outperformed the other state-of-the-art pre-trained CNNs, namely NASNet-A, Inception-v2, ResNet-50, DenseNet-201, ResNet-152, Xception, Inception-ResNet-v2, ResNeXt-101 and SENet on the ImageNet dataset in terms of accuracy with fewer parameters and FLOPS. Also, it achieved outstanding performance for the transfer learning on other image recognition datasets, namely CIFAR-100, Birdsnap, Flowers, FGVC Aircraft, and Food-101 [48].

Motivated by the benefits of the EfficientNet models, we have investigated the performance of the EfficinetNet models and other existing pre-trained CNN models for COVID-19 detection from the X-ray images. Our experimental results show that the performance of the EfficientNet-B3 model is better than the other EfficientNet models and existing pre-trained CNN models. Therefore, in this study, we have utilized the EfficientNet-B3 model for COVID-19 disease detection from the CXR images, using transfer learning. The EfficientNet-B3 is trained for 1000-classes of the ImageNet dataset. To fine-tune it for 3-classes of the COVID19CXr dataset (or 2-classes of X-ray Image dataset), the top layer of the EfficientNet-B3 model needs to be replaced with the new top layers. Therefore, we have removed the top layers of EfficientNet-B3 such as Global average pooling, Dropout, Dense, and activation (or output) and included new layers, namely Flatten, Dense, MC dropout, and Softmax in the EfficientNet-B3 model. MC dropout layer is mainly included for making network Bayesian to measure to uncertainty. The details of the original top layers and new top layers are represented in Table 1.

3.2. MC dropout

Dropout [51] is massively used as a regularization technique in the deep neural networks (NN). It mitigates the problem of the network's overfitting and improves network generalizability. The main idea in the dropout regularization is that we do not want any neurons to become

Table 1
Details of the original top layers and new top layers.

Original top layer (s)			New top layer (s)		
Layer	Output size	Trainable parameters	Layer	Output size	Trainable parameters
Global average Pooling	1536	0	Flatten	75264	0
Dropout	1536	0	Dense	1024	77071360
Dense	1000	1537000	MC dropout (0.425)	1024	0
Activation	1000	0	Softmax	3	3075

overly specified, and prediction depends on those neurons. Therefore, neurons of the given hidden layer *i* are randomly disabled with dropout probability *p*. This way, it makes sure all neurons are involved in the prediction without overfitting.

Dropout is usually applied during the network training, in which each layer randomly drops the neurons in each training step. Hence, we get a different subset of network architecture every time. To make a deep neural network Bayesian, MC dropout [52] can be used, in which dropout is not only applied during network training but also during the inference. Now, we can obtain the predictive distribution by applying the same input many times. This predictive distribution can be utilized to compute the mean of prediction and confidence measure in terms of distributional variances.

3.3. Proposed UA-ConvNet model

DL achieved tremendous success in the field of biomedical image analysis. Recently, many CAD systems has been developed for nodule detection [9], tumor segmentation [11], nuclei segmentation [12], image classification [13], COVID-19 disease detection [28–43] etc. DL methods such as deep neural networks, CNN, and dropout are used extensively. These standard DL methods do not capture model uncertainty. Whereas in the medical image analysis, it is of vital significance to representing model uncertainty. In this study, we have developed an

uncertainty-aware CNN (UA-ConvNet) model that detects COVID-19 disease from the CXR images and provides model uncertainty estimations. A schematic diagram of the proposed method is shown in Fig. 2. In this approach, first, the pre-trained EfficientNet-B3 model is fine-tuned on the CXR images using transfer learning for CXR image classification. Further, to estimate model uncertainty, we make the EfficientNet-B3 Basiyan network by adding MC-dropout in the EfficientNet-B3 model (as discussed in Section 3.2). The Bayesian neural network includes uncertainty by modeling the posterior distribution of the NN weights. With Bayesian modeling, we can measure uncertainty by evaluating the posterior distribution of the NN weights. Let us assume, given a trained neural network $f^{\mathbf{w}^*}(\cdot)$, where f denotes network architecture and $\mathbf{w}^*$ denotes network weights parameters. Also given an X-ray images dataset $I = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$, where $x^{(i)}$ denotes an CXR image and $y^{(i)}$ denotes its corresponding label (*l*) and *N* denotes the number of sample in the dataset *I*. With new pair of samples $\{x^*, y^*\}$, predictive distribution of Bayesian NN can be obtained by modeling posterior distribution as:

$$q(\mathbf{y}^*|\mathbf{x}^*, I) = \int_{\mathbf{w}} q(\mathbf{y}^*|f^{\mathbf{w}}(x^*)) d(\mathbf{w}|\mathbf{x}^*, I) d\mathbf{w} \tag{2}$$

Where $q(\mathbf{y}^*|f^{\mathbf{w}}(x^*))$ is a softmax function, and $d(\mathbf{w}|\mathbf{x}^*, I)$ is a posterior distribution of the network weights. Gal et al. [52] suggests that using MC dropout, the predictive distribution from Eq. (2) can be approximated as follows:

$$p(\mathbf{y}^*|\mathbf{x}^*) = q(\mathbf{y}^*|\mathbf{x}^*, I) = \frac{1}{M} \sum_{m=1}^{M} Softmax(f^{\mathbf{w}_m^*}(\mathbf{x}^*)) \tag{3}$$

In Eq. (3), $\mathbf{w}_m^*$ represents a set of sampled weights, and *M* represents the number of forward pass (sets). As discussed in section 3.2, by running *M* forward passes of a network with dropout can produce *M* predictions, and that can estimate the predictive distribution from Eq. (3). To obtain the model uncertainty, we calculate the standard deviation σ over the output of the softmax function of the *M* samples using Eq. (4).
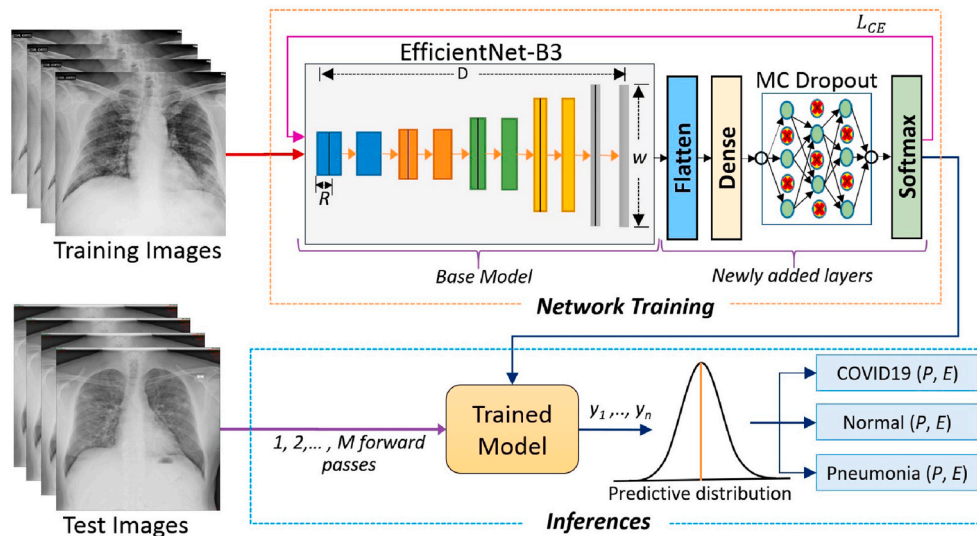
Fig. 2. Schematic diagram of the proposed method. Top layer of EfficientNet-B3 is replaced with Flatten, Dense, MC dropout and Softmax layers. During the inference test images are given to the trained model for *M* forward passes and obtained predictive distribution of size *M*.

Uncertainty: $\quad \sigma = \sqrt{\frac{1}{M} \sum_{m=1}^{M} \left[Softmax(f^{\mathbf{w}_m^*}(\mathbf{x}^*)) - p(\mathbf{y}^*|\mathbf{x}^*) \right]^2}$ (4)

The steps of the proposed method are represented in Algorithm 1.

Algorithm 1.

Algorithm 1 Uncertainty-Aware CNN

Input: Training set $Tr(x', y')$, Test set $Ts(x^*, y^*)$, and pre-trained EfficientNet-B3 network $f^{\mathbf{w}^*}(\cdot)$

Output: Prediction mean y_μ^*, model uncertainty σ.

    // Fine-tuning of EfficientNet-B3 network $f^{\mathbf{w}^*}(\cdot)$

1: Replace top layers (Global average pooling, Dropout, Dense and Activation) of network $f^{\mathbf{w}^*}(\cdot)$ with new layers (Flatten, Dense, MC dropout, and Softmax).

2: $f^{\mathbf{w}^*}(\cdot) = f^{\mathbf{w}^*}(\cdot) \cdot Train(Tr(x', y'))$

    // Predictive distribution, using MC dropout

3: **for** $m = 1$ to M **do**

4:     $P_{(m)}^* = $ Dropout (Softmax) $\left(f^{\mathbf{w}_m^*}(Ts(x^*, y^*)) \right)$

5: **end for**

    // Predictive Posterior Mean

6: $y_\mu^* = p(\mathbf{y}^* \mid \mathbf{x}^*) = \frac{1}{M} \sum_{m=1}^{M} P_{(m)}^*$

    // Model Uncertainty

7: $\sigma = \sqrt{\frac{1}{M} \sum_{m=1}^{M} \left(P_{(m)}^* - y_\mu^* \right)^2}$

8: **return** $y_\mu^*, \ \sigma$

---

4. Experiments

This section presents the implementation details of the proposed method, details of the datasets, and followed by the results of the experiments.

4.1. Datasets

The performance of the proposed method has been evaluated on three CXR image datasets, namely the COVID19CXr dataset, the X-ray image dataset and the Kaggle dataset. Details of the datasets are given below.

COVID19CXr dataset (Dataset-A): The COVID19CXr dataset has made available by Gour et al. [35]. This dataset is the combination of three publicly available datasets, namely *"COVID-19 Image Data Collection"* [53], *"Figure-1 COVID-19 CXR Dataset Initiative"* [54] and *"Mendeley data"* [55]. The COVID19CXr dataset includes 3040 chest X-ray images of COVID-19, Pneumonia, and Normal classes of 1930 patients. The images present in this dataset are of different sizes. In this study, we have referred to the COVID19CXr dataset as Dataset-A.

Table 2 represents the distribution of the images of the Dataset-A in the COVID19, Normal, and Pneumonia classes, respectively.

X-ray dataset (Dataset-B): The X-ray image dataset is provided by Ozturk et al. [41], in which they have combined the CXR images from the two publicly available datasets [53, 56]. The obtained X-ray image dataset consists of 627 CXR images of different sizes, in which, 127 CXR images of COVID-19 patients are included from the *"COVID-19 Image Data Collection"* [53] and 500 CXR images of No-findings (non--COVID-19) cases included from the *"ChestX-ray8 database"* [56]. We have referred to the X-ray image dataset as Dataset-B.

Kaggle dataset (Dataset-C): This dataset is obtained from the Kaggle repository [57]. The Kaggle dataset contains 2905 CXR images. Out of 2905 images, 219 images belong to the COVID-19 class, 1341 images belong to the Normal class, and 1345 images belong to the Pneumonia class [43,58]. All the CXR images are in 1024×1024 size. We have referred to the Kaggle dataset as Dataset-C.

To make the network learning process (training) fast, we have performed preprocessing on the CXR images of the datasets, such as images are resized to $224 \times 224 \times 3$ and pixel values of images have been normalized between 0 and 1 by dividing all pixels values by 255. Furthermore, to improve the performance of the proposed model, we

Table 2
Details of the Dataset-A.

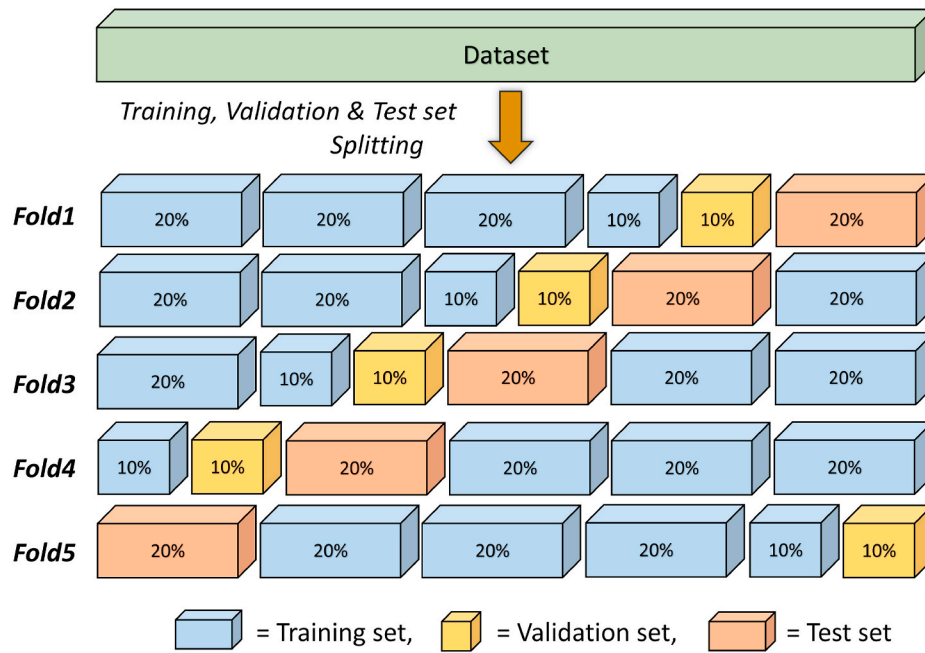| Classes	Number of Images	Number of Patients
COVID-19	546	332
NORMAL	1139	1015
PNEOMONIA	1355	583
Total	**3040**	**1930**

Fig. 3. 5-fold cross-validation strategy.

have applied data augmentation techniques, such as rotations, flips, sheers, zooms, and shifts on the training set.

4.2. Experiments and evaluation metrics

4.2.1. Experimental setup

In order to evaluate the proposed method, we have performed experiments in three scenarios. In the first scenario, the UA-ConvNet model has been trained on the Dataset-A to classify X-ray images into three categories: COVID-19, Normal, Pneumonia. In the second scenario, we have trained the UA-ConvNet model on the Dataset-B to classify images into two classes: COVID-19 and No-findings. Finally, in the third scenario, we investigated the portability (or generalizability) of the proposed model. We have trained the UA-ConvNet model on one dataset (Dataset-A/Dataset-C) and tested its performance on the other dataset (Dataset-C/Dataset-A). For making a fair comparison with the existing approaches, we have followed the same experimental protocol as in Refs. [35,41]. The performance of UA-ConvNet is evaluated using a 5-fold cross-validation strategy as shown in Fig. 3, in which the dataset has divided into a training set, a validation set, and a test set in the ratio of 70:10:20, respectively at the patient level (There are no overlapping images of patients between training, validation, and test sets). The same strategy has been repeated five times to obtained 5-folds.

For the training of the proposed UA-ConvNet model, firstly, we have performed data augmentation on the training set. Then, the proposed model has been trained on the augmented training set in a supervised learning manner, for 2000 iterations. During the training of the network, the network weights have been optimized using the RMSprop optimizer. The details of the hyper-parameters settings are given in Table 3. We experimentally find that these are the best suitable values of hyper-parameters for the proposed UA-ConvNet model and other pre-trained networks for this application.

To implement the proposed UA-ConvNet model, we have used TensorFlow and Keras framework and libraries like Numpy, Matplotlib, and Sklearn. All the experiments have been executed on the NVIDIA Tesla P100 GPU with 16 GB memory configured machine.

4.2.2. Evaluation metrics

For performance evaluation of the UA-ConvNet model, evaluation metrics like accuracy, precision, sensitivity/recall, specificity, F1-score, G-mean, Matthews correlation coefficient (MCC) (represented in Eq. (5), Eq. (6), Eq. (7), Eq. (8), Eq. (9), Eq. (10) and Eq. (11) respectively) and Area Under the receiver operating characteristic (ROC) Curve (AUC) have been used.

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \times 100\% \tag{5}$$

$$Precision = \frac{TP}{(TP + FP)} \times 100\% \tag{6}$$

$$Sensitivity\bigg/Recall = \frac{TP}{(TP + FN)} \times 100\% \tag{7}$$

$$Specificity = \frac{TN}{(TN + FP)} \times 100\% \tag{8}$$

$$F1 - Score = \frac{2TP}{(2TP + FP + FN)} \times 100\% \tag{9}$$

$$G - mean = \sqrt[3]{Sensitivity_{COVID19} \times Sensitivity_{Normal} \times Sensitivity_{Pneumonia}} \tag{10}$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TN + FN)(TP + FN)(TN + FP)}} \times 100\% \tag{11}$$

Table 3
Details of the hyper-parameter settings.

CNN (s)	Hyper-parameters			
	Optimizer	Learning Rate	Batch Size	Epochs
VGG19	RMSprob	0.0001	16	30
ResNet152	SGD	0.001	8	30
Xception	RMSprob	0.0001	16	30
DenseNet169	RMSprob	0.0001	16	30
MobileNet	SGD	0.001	8	20
NASNet-Large	SGD	0.001	8	20
Inception-V3	SGD	0.001	8	30
EfficientNet (B0–B5)	RMSprob	0.0001	32	20
UA-ConvNet	RMSprob	0.0001	32	20

Table 4
Performance of the UA-ConvNet model for multi-class classification task on the Dataset-A.

Metrics	Fold1	Fold2	Fold3	Fold4	Fold5	Mean
Prec (%)	98.49	96.82	96.74	98.60	98.68	**97.87 ± 0.99**
Sens (%)	98.26	97.16	97.46	98.76	99.13	**98.15 ± 0.84**
Spec (%)	99.00	98.21	98.41	99.12	99.42	**98.83 ± 0.51**
F1-Sc (%)	98.36	96.99	97.04	98.66	98.89	**97.99 ± 0.91**
Acc (%)	98.09	96.65	96.83	97.83	98.93	**97.67 ± 0.94**
G-mean (%)	98.15	97.07	97.35	98.34	99.18	**98.02 ± 0.75**
MCC (%)	97.14	94.78	95.14	97.43	98.18	**96.53 ± 1.33**
AUC (%)	99.47	99.65	99.40	99.84	99.89	**99.65 ± 0.19**

"Prec": Precision, "Sens": Sensitivity, "Spec": Specificity.
"F1-Sc": F1-Score, "Acc": Accuracy.

Table 5
Performance of the UA-ConvNet model for binary classification task on the Dataset-B.

Metrics	Fold1	Fold2	Fold3	Fold4	Fold5	Mean
Prec (%)	99.51	98.08	96.30	100	100	**98.78 ± 1.59**
Sens (%)	98.00	99.50	99.00	100	100	**99.30 ± 0.84**
Spec (%)	98.00	99.40	99.00	100	100	**99.28 ± 0.83**
F1-Sc (%)	98.73	98.77	97.57	100	100	**99.01 ± 1.02**
Acc (%)	98.90	99.20	98.69	100	100	**99.36 ± 0.61**
G-mean (%)	97.15	99.48	99.18	100	100	**99.16 ± 1.05**
MCC (%)	97.50	97.57	95.26	100	100	**98.07 ± 1.78**
AUC (%)	99.97	99.99	99.95	100	100	**99.98 ± 0.02**

Where TP, TN, FP, and FN represent the true positive, true negative, false positive, and false negative respectively. In this study, we considered each class equally, therefore, to get the overall metric score of the method, the mean of the metric score of each class is calculated.

To quantify the predictive uncertainty of proposed model, we have used three different evaluation metrics: standard deviation (SD), confidence interval (CI) and entropy. CI is mathematically defined in Eq. (12).

$$CI = \bar{x} \pm z \frac{\sigma}{\sqrt{M}} \quad (12)$$

Where $\bar{x}$ is the mean of the predictive distribution for the given image $x^{(i)}$, M is the number of prediction, σ is the standard deviation of the predictive distribution, and z is the critical value of the distribution. In this study, we have calculated CI at the 95% confidence level. Hence, for the 95% confidence level value of z is 1.96.

We have used entropy (E) of the prediction to measure the uncertainty of the proposed model and which can be calculated over the mean predictive distribution using the Eq. (13).

$$E(p(\mathbf{y}^*|\mathbf{x}^*)) = -\sum_{i=1}^{C} p(\mathbf{y}^*|\mathbf{x}^*) log(p(\mathbf{y}^*|\mathbf{x}^*)) \quad (13)$$

4.3. Results and discussion

This section first presents the classification performances of the UA-ConvNet model, model uncertainty analysis, portability analysis and

ablation study on pre-trained CNN models, followed by the performance comparison of the proposed model with existing methods.

4.3.1. Classification performance of UA-ConvNet model

Table 4 shows the performance of the proposed model on the Dataset-A for multi-class classification task: COVID19, Normal, and Pneumonia. Similarly, Table 5 shows the performance of the model for binary class classification task: COVID-19 and No-findings. It can be observed from Tables 4 and 5, the UA-ConvNet shows good discrimination ability for the COVID-19 disease detection from the CXR images. The proposed UA-ConvNet model achieved sensitivity of 98.15%, G-mean of 98.02%, MCC of 96.53% and F1-score of 97.99% for the classification of the CXR images into COVID-19, Normal and Pneumonia categories. On the other hand, for binary classification of the CXR images into COVID-19 and No-findings it achieved sensitivity of 99.30%, G-mean of 99.16%, MCC of 98.07% and F1-score of 99.01%. The proposed model has achieved good sensitivity on both datasets. Therefore, it reduces the chances of miss classification are significantly less. However, both datasets have a data (or class) imbalanced problem. Still, the proposed UA-ConvNet achieved better performance in terms of G-mean, MCC, F1-score, and AUC for both datasets. From that, we can infer the proposed model has efficiently handled the problem of data imbalance.

Further, the UA-ConvNet model's performance is represented by the confusion matrix and the receiver operating characteristic (ROC) curve. The confusion matrices of proposed model are shown in Table 6. It can be seen in the confusion matrices that the UA-ConvNet model produces very less values of false positives and false negatives for multi-class classification tasks. Similarly, Fig. 4 and Fig. 5 represent the ROC curves for multi-class classification and binary class classification. It can be seen in Figs. 4 and 5 that the performance of the proposed model is relatively stable over the different folds of both datasets. It achieved a mean AUC of 99.65% (95% CI of 99.65 ± 0.002) and 99.98% (95% CI of 99.98 ± 0.006) on the Dataset-A and Dataset-B, respectively.

In order to analyze the effect of sample size (number of forward passes) of predictive distribution on the model classification performances, we have evaluated model performances with a different number of forwarding passes (sample size k). Here, we have calculated mean classification error and standard deviation by evaluating the UA-ConvNet for 20 times (or for 20 forward pass), 50 times, and so on. Fig. 6a and b shows the classification error Vs sample size for multi-class classification and binary classification, respectively. Fig. 6 shows that the model performance pretty stable with the sample size.

Furthermore, we have investigated the computation time of the proposed model on the system. To calculate the computation time of the proposed model, we make a prediction 200 times for each image of the test set. Then average computation time is calculated by dividing the total time elapsed in the prediction by the total number of test set images. The proposed UA-ConvNet model requires, on average, 1.46 s of computation time to detect the disease from an X-ray image with 16 GB GPU (NVIDIA Tesla P100). Based on the computation time, the developed UA-ConvNet model is observed to be computationally efficient.
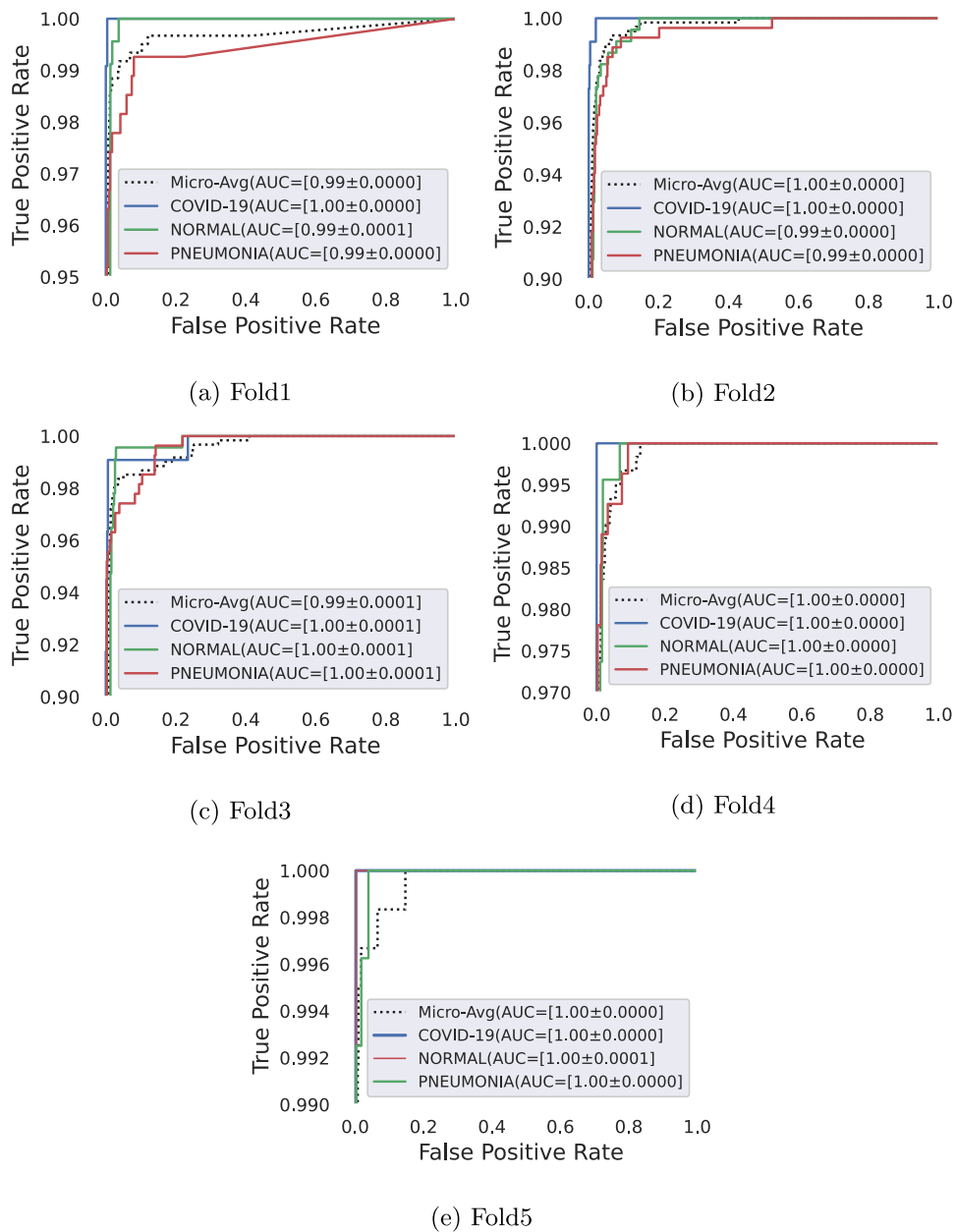
Table 6
Confusion Matrices for multi-class classification on the Dataset-A.

		Fold1			Fold2			Fold3			Fold4			Fold5		
		Predicted			Predicted			Predicted			Predicted			Predicted		
		C	N	P	C	N	P	C	N	P	C	N	P	C	N	P
Actual	C	107	0	2	110	0	1	108	0	1	110	0	0	108	0	0
	N	0	227	1	0	219	8	1	227	0	0	227	1	0	228	0
	P	0	8	263	3	8	258	3	14	254	0	9	265	2	5	260

"C": COVID19, "N": Normal, "N": Pneumonia.

(a) Fold1

(b) Fold2

(c) Fold3

(d) Fold4

(e) Fold5

Fig. 4. ROC curve for multi-class classification task on the different folds of Dataset-A.

(a) Fold1

(b) Fold2
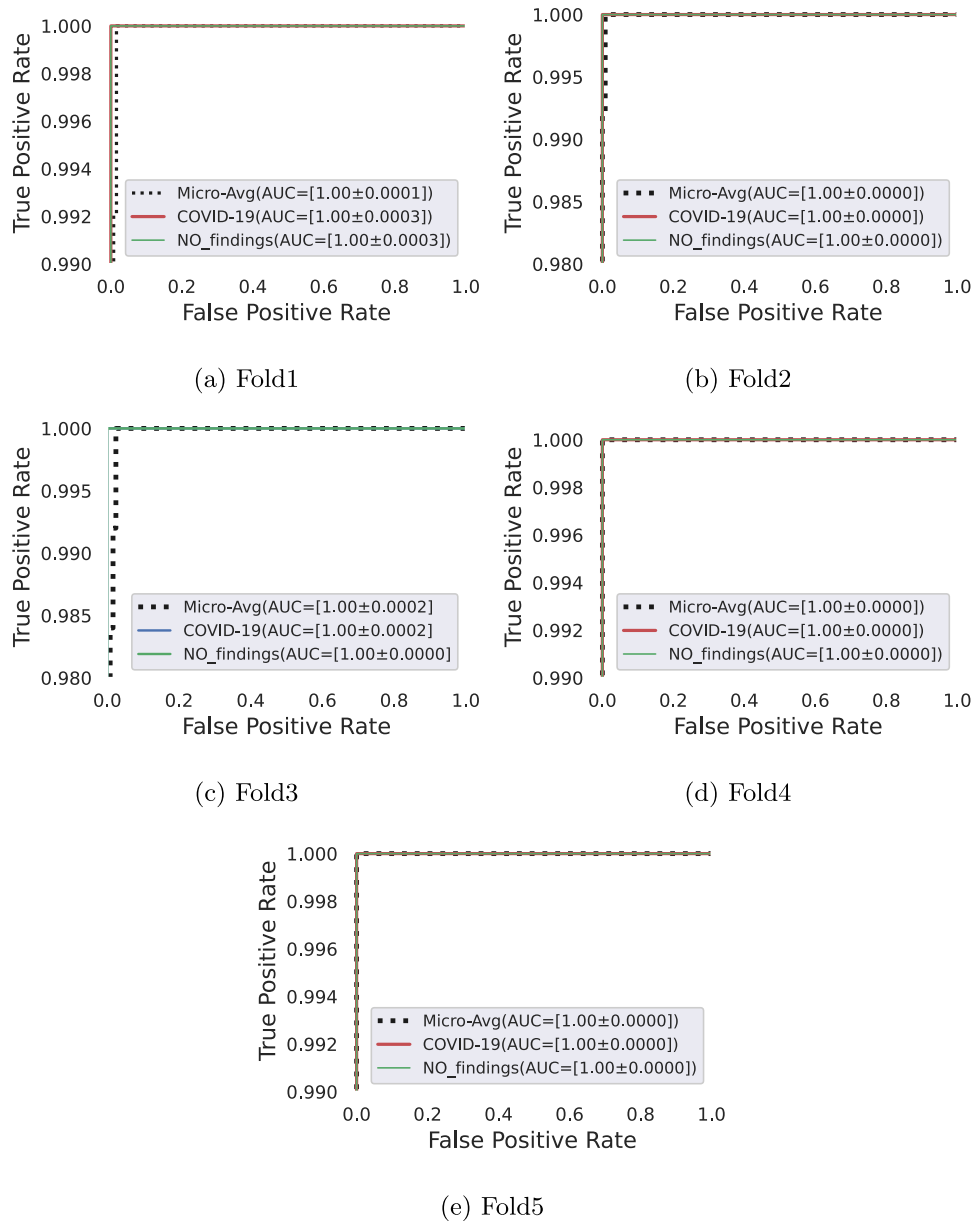
(c) Fold3

(d) Fold4

(e) Fold5

Fig. 5. ROC curve for binary classification task on the different folds of Dataset-B.

4.3.2. Uncertainty estimation

In order to estimate the model uncertainty, first, we have selected the appropriate MC dropout rate. The dropout rate should be in such a way, it not be too large and too small. If the dropout rate is too large, then predictive distribution will be very diverse, and so the estimated confidence intervals for distribution will be too large. If it is too small, then the predictive distribution will be too similar, and the confidence intervals will be small. We experimentally find that the dropout rate of 0.425 is optimal for this application. Second, we have obtained posterior predictive distribution by evaluating the UA-ConvNet model 200 times for each image of the test set. Finally, model uncertainty has been measured by calculating confidence interval, standard deviation (SD) and entropy on the obtained predictive distribution.

Fig. 7 depict the predictive distribution of the UA-ConvNet of the UA-ConvNet model. Tables 7 and 8 represent the model uncertainty for multi-class classification and binary classification, respectively. Where SD shows the variations in the prediction of the proposed model for *M* forward passes, and entropy shows the uncertainty in the prediction. It

can be seen in Tables 7 and 8 that the proposed model has shown low entropy and SD for both datasets. It can also be observed from Tables 7 and 8 that the proposed model has a confidence interval of 97.99–98.07% and 98.81–99.19% on Dataset-A and Dataset-B, respectively. Hence, we can conclude that the proposed model has high confidence and minimal variance in its prediction.

Fig. 8 shows a visual inspection of the proposed model uncertainty estimation. We have obtained the predictive distribution corresponding to each image of the test set by evaluating the UA-ConvNet model 200 times. Then, we have calculated the mean predictive probability for each class and measured the uncertainty as to the entropy of the mean predictive probability. Thereafter nine images (three images of each class) were displayed in Fig. 8, for which the proposed model shows low, high, and average predictive confidence (or uncertainty).

4.3.3. Portability analysis

The portability (or generalizability) analysis of the machine learning model is an important aspect when applying a model to other datasets.
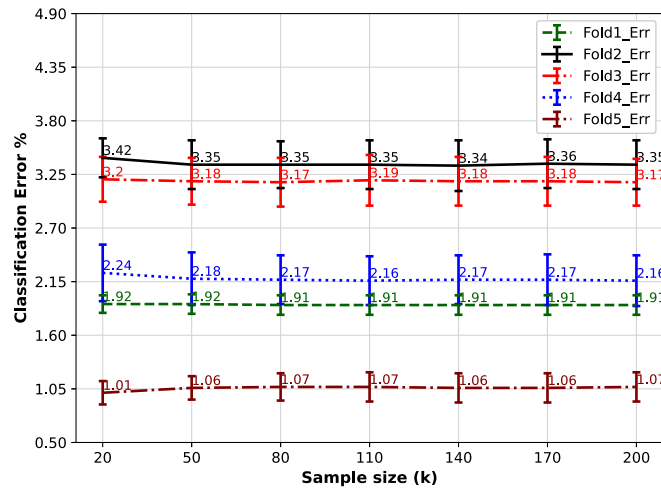
(a) Multi-class Classification
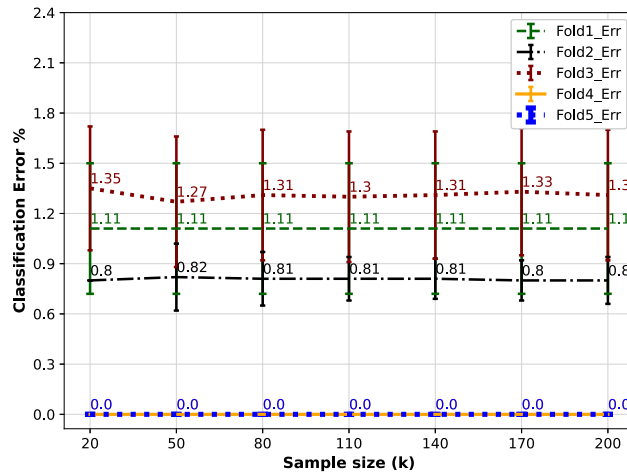
(b) Binary class classification

Fig. 6. Model classification error for different number of forward passes (or sample size (k)); where (a) represent classification error on the Dataset-A for multi-class classification task, and (b) represent classification error on the Dataset-B for binary classification task.
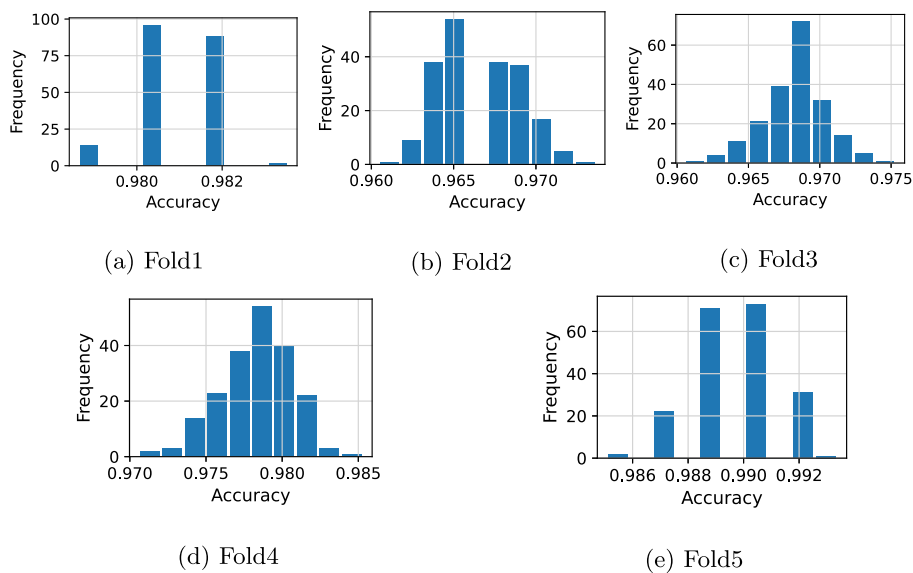

(a) Fold1

(b) Fold2

(c) Fold3

(d) Fold4

(e) Fold5

Fig. 7. UA-ConvNet model posterior predictive distribution for *M* forward passes corresponding to different folds of Dataset-A.

Table 7

G-mean, standard deviation, Entropy and confidence interval for multi-class classification task on Dataset-A.

Fold (s)	G-mean (%)	CI(%) @ 95%	SD	Entropy
Fold1	98.15	[98.13–98.17]	0.0014	0.0054
Fold2	97.07	[97.04–97.11]	0.0023	0.0327
Fold3	97.35	[97.32–97.38]	0.0022	0.0281
Fold4	98.34	[98.31–98.36]	0.0020	0.0326
Fold5	99.18	[99.17–99.20]	0.0013	0.0089
Mean	**98.02**	**[97.99 - 98.07]**	**0.0018**	**0.0215**

Table 8

G-mean, standard deviation, Entropy and confidence interval binary classification task on Dataset-B.

Fold(s)	G-mean (%)	CI(%) @ 95%	SD	Entropy
Fold1	97.15	[96.27–98.04]	0.0101	0.0075
Fold2	99.48	[99.44–99.51]	0.0025	0.0029
Fold3	99.18	[99.15–99.21]	0.0024	0.0054
Fold4	100	[100−100]	0.0000	0.0000
Fold5	100	[100−100]	0.0000	0.0000
Mean	**99.16**	**[98.81 - 99.19]**	**0.0030**	**0.0032**

We have performed two sets of experiments to analyze the portability of the proposed UA-ConvNet model for classification of CXR images into COVID-19, Normal, and Pneumonia classes. In the first set of experiments, the UA-ConvNet model has been trained on the training set of Dataset-A and tested its performance on the test set of Dataset-C. Similarly, in the second set of experiments, we have trained the UA-ConvNet model on the training set of Dataset-C and tested its performance on the test set of Dataset-A. The results are shown in Table 9.

On the test set of Dataset-A, the UA-ConvNet model trained on the Dataset-C (Dataset-C ⇒Dataset-A) has achieved a G-mean of 94.85% and there is an efficiency loss of 3.3% compared to the model trained on the Dataset-A (Dataset-A ⇒Dataset-A). On the other hand, for the test set of Dataset-C, the UA-ConvNet trained on Dataset-A (Dataset-A ⇒Dataset-C) has achieved a G-mean of 96.17% and there is an efficiency loss of 1.66% compared to the model trained on Dataset-C (Dataset-C ⇒Dataset-C). The results illustrate the good portability performance of our UA-ConvNet model on different CXR image datasets.

4.3.4. Ablation study on pre-trained CNN models

This section presents the ablation study on the EfficientNet models and existing pre-trained CNN models. First, we have investigated the performance of existing pre-trained CNNs, namely VGG19, ResNet,
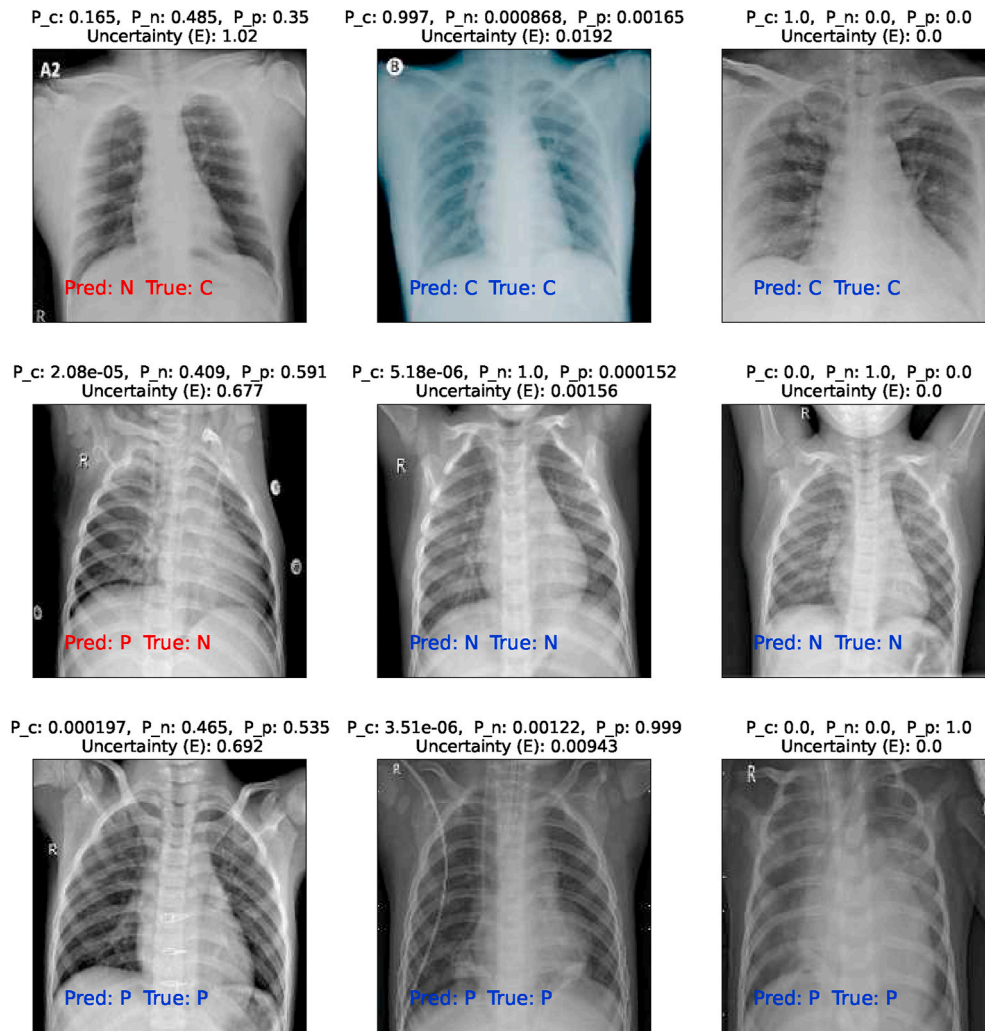
Fig. 8. X-ray images and their predicted labels, with class probabilities (COVID-19: P_c, Normal: P_n, Pneumonia: P_p) and uncertainty produced by UA-ConvNet model. Where, first, middle and last columns represent images with low confidence (or high uncertainty (E)), average confidence, and high confidence, respectively.

Table 9
Portability performance of the proposed UA-ConvNet model on Dataset-A and Dataset-C.

Training dataset ⇒Test dataset	Prec (in %)	Sens (in %)	Spec (in %)	F1-Sc (in %)	MCC (in %)	G-mean (in %)	CI(%) @ 95%	SD	Entropy
Dataset-C ⇒Dataset-C	99.04	98.27	99.17	98.64	97.64	97.83	97.75–97.91	0.0040	0.0299
Dataset-C ⇒Dataset-A	97.42	95.30	98.11	96.24	94.78	94.85	94.78–94.92	0.0036	0.0212
Dataset-A ⇒Dataset-A	98.49	98.26	99.00	98.36	97.14	98.15	98.13–98.17	0.0014	0.0054
Dataset-A ⇒Dataset-C	97.63	96.01	97.89	96.74	94.07	96.17	96.11–96.23	0.0030	0.0097

"Dataset-A ⇒Dataset-C": UA-ConvNet model has been trained on Dataset-A and tested on Dataset-C.
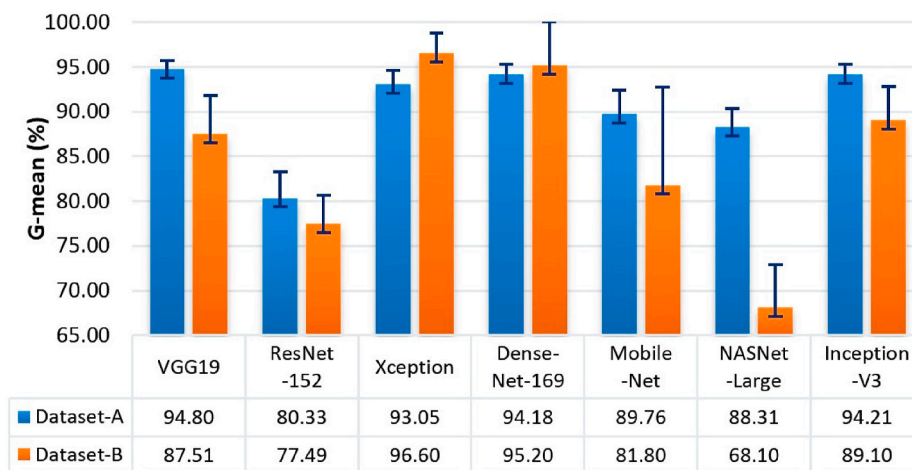

	VGG19	ResNet-152	Xception	Dense-Net-169	Mobile-Net	NASNet-Large	Inception-V3
Dataset-A	94.80	80.33	93.05	94.18	89.76	88.31	94.21
Dataset-B	87.51	77.49	96.60	95.20	81.80	68.10	89.10

Fig. 9. Performances of pre-trained CNNs on the Dataset-A and Dataset-B.

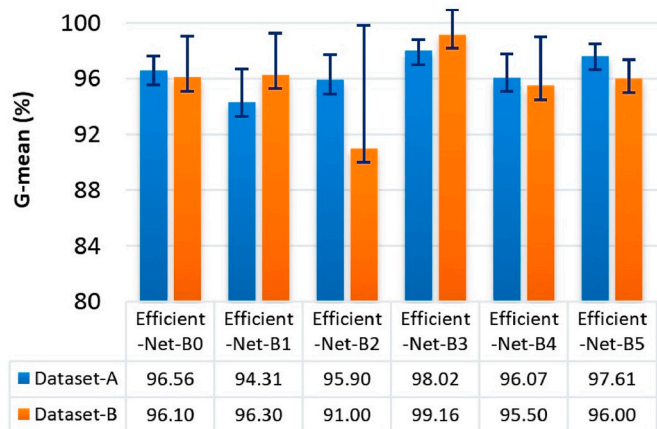|	Efficient-Net-B0	Efficient-Net-B1	Efficient-Net-B2	Efficient-Net-B3	Efficient-Net-B4	Efficient-Net-B5
Dataset-A	96.56	94.31	95.90	98.02	96.07	97.61
Dataset-B	96.10	96.30	91.00	99.16	95.50	96.00

Fig. 10. Performances of EfficientNet models on the Dataset-A and Dataset-B.

InceptionV3, DenseNet, Xception, MobileNet, and NASNetLarge, for COVID-19 disease detection from CXR images. The performances of the pre-trained networks on Dataset-A and Dataset-B in terms of G-mean are shown in Fig. 9. Among the pre-trained CNN models, the VGG19 model achieved a slightly higher G-mean of 94.80% on Dataset-A, and the Xception model achieved the highest G-mean of 96.60% on Dataset-B.

Secondly, we have tested the performances of EfficientNet models (EfficientNet-B0 to B5) for the CXR image classification on Dataset-A and Dataset-B, with the same set of hyperparameters as used in the UA-ConvNet model. Their performances in terms of G-mean are depicted

in Fig. 10. The proposed approach (EfficientNet-B3 model) achieved the best G-mean of 98.02% and 99.16% on Dataset-A and Dataset-B, respectively. It can be observed from Figs. 10 and 9 that the EfficientNet-B3 model outperformed the pre-trained CNN models and EfficientNet models on both datasets.

4.3.5. Performance comparison with existing methods

Several studies have been proposed recently on the COVID-19 detection from the X-ray. The performance comparison of our approach with existing approaches is shown in Table 10. We can see in Table 10 that the studies in Refs. [28,29,31,33,34,36–38] have evaluated on the different datasets. Most of the existing studies in Table 10 have used less than 400 COVID-19 images for developing their methods, except the studies in Refs. [31,37]. We have used 546 COVID-19 images to develop our proposed model, which is the relatively larger number of COVID-19 images among most of the studies presented in Table 10.

Since studies in Table 10 have tested for the different datasets, it is not fair to compare the performance of proposed method with their performances. However, for the binary classification of X-ray images, we can compare the performance of our method with the one proposed by Ozturk et al. in Ref. [41] as they have evaluated their method on the Dataset-B. As can be seen in Table 10, the proposed UA-ConvNet provides an improvement of 1.28% in the accuracy over the existing method [41]. Similarly, for the multi-class classification task, we can compare the proposed method with the method in studies [35,43], as they have evaluated their method on the Dataset-A and Dataset-C, respectively. It is observed from Table 10 that the proposed method shows a significant performance improvement of 4.93% and 1.77% in the accuracy over the existing methods [35,43], respectively.

Table 10
Performance comparison with the existing methods.

Author (s)	Method	Dataset\Subjects	Accuracy (%)
Narin et al. [28]	ResNet50, InceptionV3, Inception-ResNetV2,	COVID-19: 50, Normal: 50	Binary: 98
Zhang et al. [29]	Two-stage transfer: learning, ResNet34	COVID-19: 189, Normal: 235, Other Pneumonia: 63	Multi-class: 91.08
Waheed et al. [31]	Auxiliary Classifier Generative Adversarial Network	COVID-19: 403, Normal: 721	Binary: 95
Togacar et al. [33]	Fuzzy color, Stacking, MobileNetV2, SqueenzeNet SVM	COVID-19: 295, Normal: 65, Pneumonia: 98	Multi-class: 99.27
Oh et al. [34]	ResNet18	Normal:191, Bacterial: 54, Tuberculosis:57, Viral:20, COVID-19:180	Multi-class: 91.9
Gianchandani et al. [37]	Ensemble models: Vgg16, ResNet152, DenseNet201	COVID-19: 423, Normal: 1579, Viral Pneumonia: 1485	Binary: 99.21 Multi-class: 96.15
Wang et al. [38]	COVID-Net	COVID-19: 358, Normal: 8066, non-COVID19: 5538	Multi-class: 93.3
Ozturk et al. [41]	DarkCovidNet: DarkNet19, YOLO	**Dataset-B:** COVID-19: 127, No-findings: 500	Binary: 98.08
Maheshwari et al. [43]	LBP, image-based features SVM	**Dataset-C:** COVID-19: 219, Normal: 1341, Pneumonia: 1345	Multi-class: 96.99
Gour et al. [35]	Stacked CNN model: VGG19, CovNet30	**Dataset-A:** COVID-19: 546, Normal: 1139, Pneumonia: 1355	Multi-class: 92.74
Proposed	UA-ConvNet: EfficientNet-B3, MC-dropout	**Dataset-A** [35]	Multi-class: **97.67**
		Dataset-C [43]	Multi-class: 98.76
		Dataset-B [41]	Binary: **99.36**

5. Conclusion

In this study, we developed an uncertainty-aware CNN model for automated screening of COVID-19 disease from the CXR images along with model uncertainty estimations. Specifically, we have included MC dropout with the EfficientNet-B3 network to make it Bayesian and estimated the model uncertainty by modeling the posterior predictive distribution of M samples (obtained by running M forward passes of the trained network). We demonstrated how uncertainty could be modeled in the CAD system for COVID-19 detection from the CXR images. Our experimental results on the Dataset-A and Dataset-B show the effectiveness of the UA-ConvNet model for the detection of COVID-19 disease from the CXR images. More, importantly, the proposed UA-ConvNet model has achieved an G-mean of 98.02% (with CI of 97.99–98.07) and 99.16% (with CI of 98.81–99.19) for multi-class classification task and binary classification task, respectively. For the classification of CXR images proposed approach outperforms the existing approaches. In the future, we would like to explore the UA-ConvNet model to detect COVID-19 disease from the chest CT images.

Declaration of competing interest

None.

References

[1] F. Wu, S. Zhao, B. Yu, Y.-M. Chen, W. Wang, Z.-G. Song, Y. Hu, Z.-W. Tao, J.-H. Tian, Y.-Y. Pei, et al., A new coronavirus associated with human respiratory disease in China, Nature 579 (7798) (2020) 265–269.
[2] Z. Wu, J.M. McGoogan, Characteristics of and important lessons from the coronavirus disease 2019 (covid-19) outbreak in China: summary of a report of 72 314 cases from the Chinese center for disease control and prevention, Jama 323 (13) (2020) 1239–1242.
[3] T. Singhal, A review of coronavirus disease-2019 (covid-19), Indian J. Pediatr. (2020) 1–6.
[4] Who, Novel Coronavirus (2019-ncov) Technical Guidance: Laboratory Testing for 2019-ncov in Humans, World Health Organization, Geneva, Switzerland, https://www.who.int/emergencies/diseases/novel-coronavirus-2019/technical-guidance/laboratoryguidance.
[5] Y. Fang, H. Zhang, J. Xie, M. Lin, L. Ying, P. Pang, W. Ji, Sensitivity of chest ct for covid-19: comparison to rt-pcr, Radiology (2020) 200432.
[6] T. Ai, Z. Yang, H. Hou, C. Zhan, C. Chen, W. Lv, Q. Tao, Z. Sun, L. Xia, Correlation of chest ct and rt-pcr testing in coronavirus disease 2019 (covid-19) in China: a report of 1014 cases, Radiology (2020) 200642.
[7] M.-Y. Ng, E.Y. Lee, J. Yang, F. Yang, X. Li, H. Wang, M.M.-s. Lui, C.S.-Y. Lo, B. Leung, P.-L. Khong, et al., Imaging profile of the covid-19 infection: radiologic findings and literature review, Radiology: Cardiothoracic Imaging 2 (1) (2020), e200034.
[8] J.P. Kanne, B.P. Little, J.H. Chung, B.M. Elicker, L.H. Ketai, Essentials for Radiologists on Covid-19: an Update Radiology Scientific Expert Panel, 2020.
[9] X. Huang, J. Shan, V. Vaidya, Lung nodule detection in ct using 3d convolutional neural networks, in: 2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017), IEEE, 2017, pp. 379–383.
[10] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya, et al., Chexnet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning, arXiv preprint arXiv:1711.05225.
[11] M. Talo, U.B. Baloglu, Ö. Yıldırım, U.R. Acharya, Application of deep transfer learning for automated brain abnormality classification using mr images, Cognit. Syst. Res. 54 (2019) 176–188.
[12] M. Gour, S. Jain, R. Agrawal, Deeprnnetseg: deep residual neural network for nuclei segmentation on breast cancer histopathological images, in: International Conference on Computer Vision and Image Processing, Springer, 2019, pp. 243–253.
[13] M. Gour, S. Jain, T.S. Kumar, Residual learning based cnn for breast cancer histopathological image classification, Int. J. Imag. Syst. Technol. 30 (3) (2020) 621–635.
[14] H. Zhao, X. Qiu, W. Lu, H. Huang, X. Jin, High-quality retinal vessel segmentation using generative adversarial network with a large receptive field, Int. J. Imag. Syst. Technol. 30 (3) (2020) 828–842.

[15] R. Shwartz-Ziv, N. Tishby, Opening the Black Box of Deep Neural Networks via Information, arXiv preprint arXiv:1703.00810.

[16] S. Yu, J.C. Principe, Understanding autoencoders with information theoretic concepts, Neural Network. 117 (2019) 104–123.

[17] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.

[18] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR), 2016, pp. 770–778, https://doi.org/10.1109/CVPR.2016.90.

[19] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2818–2826.

[20] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4700–4708.

[21] F. Chollet, Xception: deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258.

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

[23] B. Zoph, V. Vasudevan, J. Shlens, Q.V. Le, Learning transferable architectures for scalable image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8697–8710.

[24] M. Anthimopoulos, S. Christodoulidis, L. Ebner, A. Christe, S. Mougiakakou, Lung pattern classification for interstitial lung diseases using a deep convolutional neural network, IEEE Trans. Med. Imag. 35 (5) (2016) 1207–1216.

[25] S. Christodoulidis, M. Anthimopoulos, L. Ebner, A. Christe, S. Mougiakakou, Multisource transfer learning with convolutional neural networks for lung pattern analysis, IEEE.J. Biomed.Health Inf. 21 (1) (2017) 76–84.

[26] J. Shuja, E. Alanazi, W. Alasmary, A. Alashaikh, Covid-19 Open Source Data Sets: A Comprehensive Survey, Applied Intelligence, 2020, pp. 1–30.

[27] D. Dong, Z. Tang, S. Wang, H. Hui, L. Gong, Y. Lu, Z. Xue, H. Liao, F. Chen, F. Yang, et al., The role of imaging in the detection and management of covid-19: a review, IEEE.Rev. Biomed. Eng. 14 (2021) 16–29, https://doi.org/10.1109/RBME.2020.2990959.

[28] A. Narin, C. Kaya, Z. Pamuk, Automatic Detection of Coronavirus Disease (Covid-19) Using X-Ray Images and Deep Convolutional Neural Networks, Pattern Analysis and Applications, 2021, pp. 1–14.

[29] R. Zhang, Z. Guo, Y. Sun, Q. Lu, Z. Xu, Z. Yao, M. Duan, S. Liu, Y. Ren, L. Huang, et al., Covid19xraynet: a two-step transfer learning model for the covid-19 detecting problem based on a limited number of chest x-ray images, Interdiscipl. Sci. Comput. Life Sci. 12 (4) (2020) 555–565.

[30] A.A. Ardakani, A.R. Kanafi, U.R. Acharya, N. Khadem, A. Mohammadi, Application of deep learning technique to manage covid-19 in routine clinical practice using ct images: results of 10 convolutional neural networks, Comput. Biol. Med. (2020) 103795.

[31] A. Waheed, M. Goyal, D. Gupta, A. Khanna, F. Al-Turjman, P.R. Pinheiro, Covidgan: data augmentation using auxiliary classifier gan for improved covid-19 detection, IEEE Access 8 (2020) 91916–91923.

[32] J. Wang, Y. Bao, Y. Wen, H. Lu, H. Luo, Y. Xiang, X. Li, C. Liu, D. Qian, Prior-attention residual learning for more discriminative covid-19 screening in ct images, IEEE Trans. Med. Imag. 39 (8) (2020) 2572–2583, https://doi.org/10.1109/TMI.2020.2994908.

[33] M. Toğaçar, B. Ergen, Z. Cömert, 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. Biol. Med. 121 (2020) 103805, https://doi.org/10.1016/j.compbiomed.2020.103805.

[34] Y. Oh, S. Park, J.C. Ye, Deep learning covid-19 features on cxr using limited training data sets, IEEE Trans. Med. Imag. 39 (8) (2020) 2688–2700, https://doi.org/10.1109/TMI.2020.2993291.

[35] M. Gour, S. Jain, Stacked Convolutional Neural Network for Diagnosis of Covid-19 Disease from X-Ray Images, arXiv preprint arXiv:2006.13817.

[36] P. Autee, S. Bagwe, V. Shah, K. Srivastava, Stacknet-denvis: a multi-layer perceptron stacked ensembling approach for covid-19 detection using x-ray images, Phys. Eng.Sci. Med. (2020) 1–16.

[37] N. Gianchandani, A. Jaiswal, D. Singh, V. Kumar, M. Kaur, Rapid covid-19 diagnosis using ensemble deep transfer learning models from chest radiographic images, J. Ambient Intell.humanized computing (2020) 1–13.

[38] L. Wang, Z.Q. Lin, A. Wong, Covid-net: a tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images, Sci. Rep. 10 (1) (2020) 1–12.

[39] M. Kuchana, A. Srivastava, R. Das, J. Mathew, A. Mishra, K. Khatter, Ai aiding in diagnosing, tracking recovery of covid-19 using deep learning on chest ct scans, Multimed. Tool. Appl. (2020) 1–15.

[40] X. Zhao, P. Zhang, F. Song, G. Fan, Y. Sun, Y. Wang, Z. Tian, L. Zhang, G. Zhang, D2a u-net: automatic segmentation of covid-19 ct slices based on dual attention and hybrid dilated convolution, Comput. Biol. Med. 135 (2021) 104526, https://doi.org/10.1016/j.compbiomed.2021.104526.

[41] T. Ozturk, M. Talo, E.A. Yildirim, U.B. Baloglu, O. Yildirim, U. Rajendra Acharya, Automated detection of covid-19 cases using deep neural networks with x-ray images, Comput. Biol. Med. 121 (2020) 103792, https://doi.org/10.1016/j.compbiomed.2020.103792.

[42] N.K. Mishra, P. Singh, S.D. Joshi, Automated detection of covid-19 from ct scan using convolutional neural network, Biocybern.Biomed. Eng. 41 (2) (2021) 572–588, https://doi.org/10.1016/j.bbe.2021.04.006.

[43] S. Maheshwari, R.R. Sharma, M. Kumar, Lbp-based information assisted intelligent system for covid-19 identification, Comput. Biol. Med. 134 (2021) 104453, https://doi.org/10.1016/j.compbiomed.2021.104453.

[44] A. Shamsi, H. Asgharnezhad, S.S. Jokandan, A. Khosravi, P.M. Kebria, D. Nahavandi, S. Nahavandi, D. Srinivasan, An uncertainty-aware transfer learning-based framework for covid-19 diagnosis, IEEE. Trans. Neural Network. Learn. Syst. 32 (4) (2021) 1408–1417, https://doi.org/10.1109/TNNLS.2021.3054306.

[45] S. Calderon-Ramirez, S. Yang, A. Moemeni, S. Colreavy-Donnelly, D.A. Elizondo, L. Oala, J. Rodríguez-Capitán, M. Jiménez-Navarro, E. López-Rubio, M.A. Molina-Cabello, Improving uncertainty estimation with semi-supervised deep learning for covid-19 detection using chest x-ray images, IEEE Access 9 (2021) 85442–85454, https://doi.org/10.1109/ACCESS.2021.3085418.

[46] S. Dong, Q. Yang, Y. Fu, M. Tian, C. Zhuo, Rconet: Deformable mutual information maximization and high-order uncertainty-aware learning for robust covid-19 detection, IEEE. Trans. Neural Network. Learn. Syst. 32 (8) (2021) 3401–3411, https://doi.org/10.1109/TNNLS.2021.3086570.

[47] J. Postels, F. Ferroni, H. Coskun, N. Navab, F. Tombari, Sampling-free epistemic uncertainty estimation using approximated variance propagation, in: 2019 IEEE/CVF International Conference on Computer Vision, ICCV), 2019, pp. 2931–2940, https://doi.org/10.1109/ICCV.2019.00302.

[48] M. Tan, Q. Le, Efficientnet, Rethinking model scaling for convolutional neural networks, in: International Conference on Machine Learning, PMLR, 2019, pp. 6105–6114.

[49] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., Imagenet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (3) (2015) 211–252.

[50] Y. Huang, Y. Cheng, A. Bapna, O. Firat, D. Chen, M. Chen, H. Lee, J. Ngiam, Q. V. Le, Y. Wu, et al., Gpipe: efficient training of giant neural networks using pipeline parallelism, Adv. Neural Inf. Process. Syst. 32 (2019) 103–112.

[51] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (1) (2014) 1929–1958.

[52] Y. Gal, Z. Ghahramani, Dropout as a bayesian approximation: representing model uncertainty in deep learning, in: International Conference on Machine Learning, 2016, pp. 1050–1059.

[53] J. P. Cohen, P. Morrison, L. Dao, K. Roth, T. Q. Duong, M. Ghassemi, Covid-19 Image Data Collection: Prospective Predictions Are the Future, arXiv preprint arXiv:2006.11988.

[54] L. Wang, A. Wong, Z.Q. Lin, P. McInnis, A. Chung, H. Gunraj, Figure 1 covid-19 chest x-ray data initiative [Accessed 15 June 2020] (2020). URL, https://github.com/agchung/Figure 1-COVID-chestxray-dataset.

[55] D. Kermany, K. Zhang, M. Goldbaum, Labeled optical coherence tomography (oct) and chest x-ray images for classification, Mendeley data 2.

[56] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, R.M. Summers, Chestx-ray8: hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2097–2106.

[57] T. Rahman, M. Chowdhury, A. Khandakar, Covid-19 radiography database, URL, https://www.kaggle.com/tawsifurrahman/covid19-radiography-database, 2021. (Accessed 15 August 2021).

[58] M.E.H. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M.A. Kadir, Z. B. Mahbub, K.R. Islam, M.S. Khan, A. Iqbal, N.A. Emadi, M.B.I. Reaz, M.T. Islam, Can ai help in screening viral and covid-19 pneumonia? IEEE Access 8 (2020) 132665–132676, https://doi.org/10.1109/ACCESS.2020.3010287.