A lightweight CNN-based network on COVID-19 detection using X-ray and CT images

Mei-Ling Huang *, Yu-Chieh Liao

Department of Industrial Engineering & Management, National Chin-Yi University of Technology, 57, Sec. 2, Zhongshan Rd., Taiping Dist., Taichung, 411030, Taiwan

ABSTRACT

Background and objectives: The traditional method of detecting COVID-19 disease mainly rely on the interpretation of computer tomography (CT) or X-ray images (X-ray) by doctors or professional researchers to identify whether it is COVID-19 disease, which is easy to cause identification mistakes. In this study, the technology of convolutional neural network is expected to be able to efficiently and accurately identify the COVID-19 disease.
Methods: This study uses and fine-tunes seven convolutional neural networks including InceptionV3, ResNet50V2, Xception, DenseNet121, MobileNetV2, EfficientNet-B0, and EfficientNetV2 on COVID-19 detection. In addition, we proposes a lightweight convolutional neural network, LightEfficientNetV2, on small number of chest X-ray and CT images. Five-fold cross-validation was used to evaluate the performance of each model. To confirm the performance of the proposed model, LightEfficientNetV2 was carried out on three different datasets (NIH Chest X-rays, SARS-CoV-2 and COVID-CT).
Results: On chest X-ray image dataset, the highest accuracy 96.50% was from InceptionV3 before fine-tuning; and the highest accuracy 97.73% was from EfficientNetV2 after fine-tuning. The accuracy of the LightEfficientNetV2 model proposed in this study is 98.33% on chest X-ray image. On CT images, the best transfer learning model before fine-tuning is MobileNetV2, with an accuracy of 94.46%; the best transfer learning model after fine-tuning is Xception, with an accuracy of 96.78%. The accuracy of the LightEfficientNetV2 model proposed in this study is 97.48% on CT image.
Conclusions: Compared with the SOTA, LightEfficientNetV2 proposed in this study demonstrates promising performance on chest X-ray images, CT images and three different datasets.

1. Introduction

COVID-19 has spread around the world. Since the virus spreads through human-to-human interaction, the speed of spread is getting faster and faster. As a result, more than 200 countries and regions around the world have been affected by the virus [1]. In the process of epidemic prevention, one of the early diagnosis of COVID-19 is through the use of reverse transcription polymerase chain reaction (RT-PCR) by doctors or medical researchers to detect positive cases of COVID-19. The method is insufficient, expensive and time-consuming [2]. Computer tomography (CT) and chest X-ray (CXR) images are used as other important detection indicators [3]. Practically, chest X-ray images are often interpreted by a radiologist, a process that is time-consuming and prone to errors in subjective assessments. The operation of neural network requires large and high-speed memory access. With the rapid improvement of modern algorithms, big data, and hardware computing

capability, the use of artificial intelligence for medical image recognition can be efficient and effective. Among them, Convolutional Neural Networks (CNN) has been proven to be a powerful medical image recognition technology [4].

Some scholars use image normalization, data augmentation and image resizing for image preprocessing [5,6]. For chest X-ray image classification, related researches use methods such as deep learning, transfer learning, and ensemble learning. Science et al. [7] proposed a CNN with chaotic salt swarm algorithm, and the classification accuracy on COVID-19, Viral Pneumonia and Normal groups was 99.69%. Rashid et al. [8] proposed AutoCovNet and achieved classification accuracy of 96.45% on three groups of COVID-19, Pneumonia and Normal; classification accuracy of 99.39% on two groups of COVID-19 and Pneumonia. Based on DenseNet121, Hertel and Benlamri [9] proposed COV-SNET and achieved classification accuracy of 95.00%. Aslan et al. [10] used eight convolutional neural networks, including AlexNet,

ResNet18, ResNet50, InceptionV3, DenseNet201, InceptionResNetV2, MobileNetV2, and GoogleNet, to extract features from chest X-ray images, and combined with machine learning classifiers using 3,781 chest X-ray images after data augmentation. The results show that combination of DenseNet and SVM reaches an accuracy of 96.29% on three group classifications of COVID-19, viral pneumonia and normal.

Nigam et al. [11] applied VGG16, DenseNet121, Xception, NASNet and EfficientNet. The best classification accuracy of 93.48% on three groups of COVID-19, Pneumonia and Normal was from EfficientNet-B7. The best classification accuracy of 93.48% on two groups of COVID-19 and Pneumonia was from VGG19 [12]. EfficientNet-B4 model results in classification accuracy of 99.62% on two groups of COVID-19 and Normal, and classification accuracy of 96.70% on three groups of COVID-19, Pneumonia and Normal [13].

Gupta et al. [14] proposed InstaCovNet-19 using ensemble methods including ResNet101, Xception, InceptionV3, MobileNet and NASNet. Results showed that the accuracy on three groups of COVID-19, Pneumonia and Normal was 99.08%. Rahimzadeh and Attar [15] combined Xception and ReNet50V2 and obtained classification accuracy of 91.40% on three groups of COVID-19, Pneumonia and Normal. An ensemble learning network DNE-RL optimizing hyper parameters was proposed on two group classification of COVID-19 and Normal, the accuracy was as high as 99.14% [16]. A computer-aided detection model first combines several pre-trained networks for feature extraction, and then uses a sparse auto-encoder and a feed forward neural network (FFNN) to improve the classification performance. The results show that the binary classification accuracy of InceptionResNetV2+Xception in COVID-19 and Non-COVID is 95.78% [17].

On the classification of chest CT images, Wang et al. [18] first used the two-dimensional fractional Fourier entropy to extract features, and then proposed a deep stacked sparse auto-encoder (DSSAE). Data augmentation was used with a total of 1,164 CT images. The results show that classification accuracy, sensitivity and F1-Score for COVID-19, community-acquired pneumonia (CAP), secondary pulmonary tuberculosis (SPT) and normal (Healthy control, HC) are 92.46%, 92.46% and 92.32%, respectively. Rohila et al. [19] proposed ReCOV-101 convolutional neural network on two group classification of COVID-19 and non-COVID-19, the accuracy was as high as 94.90%. Ten convolutional neural networks including AlexNet, VGG16, VGG19, SqueezeNet, GoogleNet, MobileNetV2, ResNet18, ResNet50, ResNet101 and Xception were applied on two group classifications between COVID-19 and non-COVID-19. Results showed that the classification accuracy from ResNet101 was 99.51% [20]. Fu et al. [21] used UNet on chest segmentation, and proposed DenseANet model on classification of three groups of COVID-19, Pneumonia and Normal to obtain the accuracy of 96.06%. Transfer learning was executed on DenseNet121, DenseNet201, VGG16, VGG19, InceptionResNetV2 and Xception models. Further, the classification accuracy achieved 98.80% through combing DenseNet201 and GradCam algorithm on groups of COVID-19 and Pneumonia [22]. Arora et al. [23] applied MobileNet on groups of COVID-19 and Pneumonia to report the classification accuracy of 96.11%. Aswathy et al. [24] proposed a two-step classification method between COVID-19 and non-COVID-19, and the accuracy was 98.50%. Fan et al. [25] proposed a *Trans*-CNN Net in 2022. This method combines the existing convolutional neural network and the branch network model of the transformer structure to fuse extracted features, using a total of 194,922 CT images. The results show that *Trans*-CNN Net has an accuracy of 96.73% in three classifications of COVID-19, pneumonia and normal. Zhang et al. [26] proposed PZM-DSSAE in 2021, which first uses the Pseudo-Zernike moment (PZM) to extract the features of the image, including the image plane over unit circle and image plane inside the unit circle; and then use deep stacked sparse autoencoder (DSSAE) as the classifier with nine data augmentation methods. A total of 2960 CT images was used in this study. The results showed that PZM-DSSAE had a binary classification accuracy of 92.31% in COVID-19 and normal, a sensitivity of 92.06%, a specificity of 92.56%, and an F1-Score of 92.29%.

Cruz [27] combined ensemble learning and transfer learning on classification between COVID-19 and non-COVID-19, but the accuracy was not high (86.70%). Based on AlexNet, GoogleNet and ResNet, Zhou et al. [28] proposed EDL-COVID ensemble convolutional neural network to extract image features on classification of three groups of COVID-19, Pneumonia and Normal. The final accuracy was as high as 99.05%. Transfer learning was applied on VGG19, ResNet101, DenseNet169 and WideResNet50-2, followed by ensemble method to obtain accuracy of 93.50% on classification between COVID-19 and non-COVID-19 [29]. Abraham and Nair [30] use ensemble learning model combining five convolutional neural networks such as MobileNetV2, ShuffleNet, Xception, DarkNet53 and EfficientNet-B0 for feature extraction, and use the kernel support vector machine (Kernel support vector machine) as the classifier, using a total of 746 CT images. The results show that ensemble learning model has a binary classification accuracy of 91.60% in COVID-19 and Non-COVID.

In addition to using chest X-ray or CT images individually, some researches considered both images together in their studies. Vinod et al. [31] proposed Deep Covix-Net network on chest X-ray images of three groups of COVID-19, Pneumonia and Normal, and the classification accuracy was 96.80%; the classification accuracy was 97.00% between COVID-19 and Normal by using CT images. Li et al. [32] proposed CMT-CNN network. The classification accuracies on chest X-ray images between COVID-19 and Normal, and on three groups of COVID-19, Pneumonia and Normal were 97.23% and 93.49%, respectively; the classification accuracies on chest CT images between COVID-19 and Normal, and on three groups of COVID-19, Pneumonia and Normal were 93.46% and 91.45%, respectively. Thakur and Kumar [33] combined 6077 X-ray images and 3877 CT images to obtain high classification of 99.64% and 98.28% between COVID-19 and Normal, and on three groups of COVID-19, Pneumonia and Normal, respectively.

Munusamy et al. [34] proposed FractalCovNet network by combining Fractal module and UNet to target the CT image and perform classification task. Banerjee et al. [35] combined InceptionV3, InceptionResNetV2 and DenseNet201 to form COFE-Net on two group classifications of X-ray images between COVID-19 and non-COVID-19 and three group classification of COVID-19, Pneumonia and Normal. The accuracies were 99.49% and 96.39%, respectively; the accuracies on CT images were 99.68% and 98.93%, respectively. Zhang et al. [36] used convolutional block attention module to propose an end-to-end multiple-input deep convolutional attention network (MIDCAN) in 2021, where one input receives 3D CT images and the other receives 2D chest X-ray images. Besides, in order to overcome the problem of overfitting, the authors use a variety of data augmentation methods. The results showed that MIDCAN correctly detected COVID-19 with accuracy, sensitivity and specificity of 98.02%, 98.10%, and 97.95%, respectively.

Related researches have demonstrated the alternatives of using lung ultrasound for COVID-19 diagnosis, and mentioned the drawbacks of diagnosis using chest X-ray or CT images. Salvia et al. [37] adopted ResNet50, transfer learning, and data augmentation techniques on COVID-19 detection using ultrasound clips originating from linear and convex probes. A total of 5,400 ultrasound images were used. The results show that the accuracy of ResNet50 in four and seven classes are 98.32% and 98.72%, respectively. Ebadi et al. [38] proposed a Kinetics-I3D network, which can interpret lung ultrasound images quickly and reliably, and can classify the entire lung ultrasound images without using preprocessing or a frame -by- frame analysis. Kinetics-I3D network
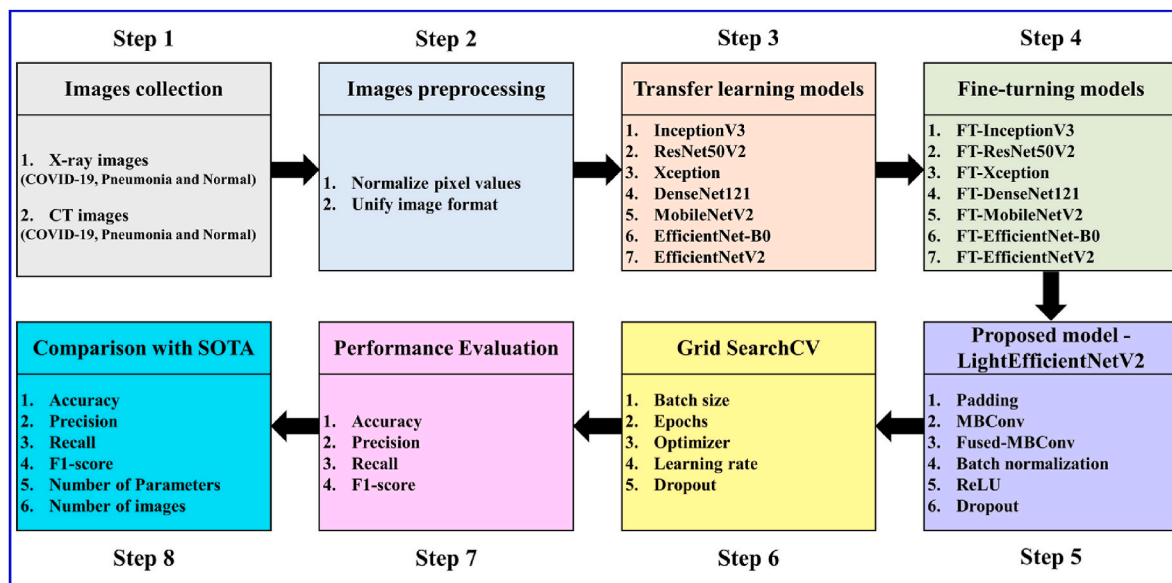
Fig. 1. Structure of this research.

Table 1
Dataset 1: Chest X-ray image dataset.

Category	Total number of images	Number of images selected in this study	train	validation	test
COVID-19	3,616	600	400	100	100
Pneumonia	5,618	600	400	100	100
Normal	11,775	600	400	100	100
Total	21,009	1800	1,200	300	300

Table 2
Dataset 2: Chest CT image dataset.

Category	Total number of images	Number of images selected in this study	train	validation	test
COVID-19	9,003	600	400	100	100
Pneumonia	1,120	600	400	100	100
Normal	7,063	600	400	100	100
Total	17,186	1800	1,200	300	300

achieves an accuracy of 90% and an average precision score of 95%. Huang et al. [39] propose a Non-local Channel Attention ResNet to analyze the lung ultrasound images and automatically score the degree of pulmonary edema of patients with COVID-19 pneumonia. 2062 effective images were used. Results show that the binary classification achieves 92.34%.

Compared with traditional machine learning methods, CNN have demonstrated great ability on COVID-19 detection by using chest X-ray or CT images jointly or separately. Most of the aforementioned development of the related studies proposed new networks aiming to obtain higher classification accuracy may not consider the number of parameters used in the network architecture or the number of images used in the training phase [40–49]. The objective of this study is to efficiently speed up the training phase with fewer parameters when using a small number of images collected for each group while maintaining favorable classification accuracy for COVID-19 detection. The classification

accuracy, Precision, Recall, and F1-Score, and the numbers of parameter used for each model were compared in this study. In addition to the pre-trained transfer learning models before and after fine-tuning, based on the architectures of AlexNet and EfficientNetV2 networks, this study proposes a lightweight convolutional neural network Light-EfficientNetV2 to faster the training time for COVID-19 detection.

The rest of this paper is organized as follows. Section 2 introduces the collection of datasets on chest X-ray and CT images; image preprocessing method, transfer learning models and the proposed model. Section 3 presents experiment results. Section 4 compares the classification results with the state-of-the-art COVID-19 classification models. The proposed model is further evaluated using three more open public datasets. Finally, Section 5 summarizes the proposed model and suggests the future research.

2. Materials and methods

2.1. Data collection

Chest X-ray images and CT images of COVID-19, Pneumonia and Normal are collected from three Kaggle datasets and one Mendeley dataset in this study. Fig. 1 shows the structure of the study.

Step 1 Image collection: Establish Dataset 1 and Dataset 2. Chest X-ray images (Dataset 1) are collected from Kaggle Chest X-Ray Images (Pneumonia) [50] and Kaggle COVID-19 Radiography Dataset [51]. CT images (Dataset 2) are collected from Kaggle Large COVID-19 CT scan slice dataset [52] and Mendeley Data Chest CT of multiple pneumonia [53]. Each dataset contains three groups of COVID-19, Pneumonia and Normal images.
Step 2 Image preprocessing: Normalize pixel values and unify image format for Dataset 1 and Dataset 2.
Step 3 Transfer learning models: InceptionV3, ResNet50V2, Xception, DenseNet121, MobileNetV2 and EfficientNet-B0.
Step 4 Fine-tuning models: the above 6 models are fine tuned.
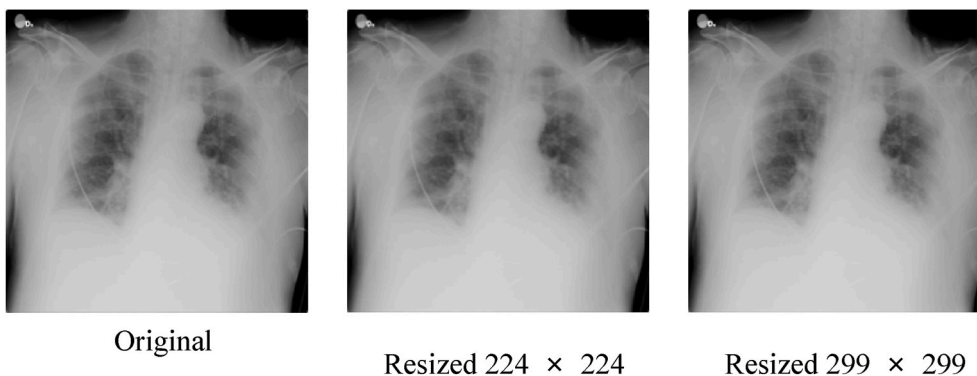Step 5 Propose a model: We propose a light-weight LightEfficientNetV2 model.
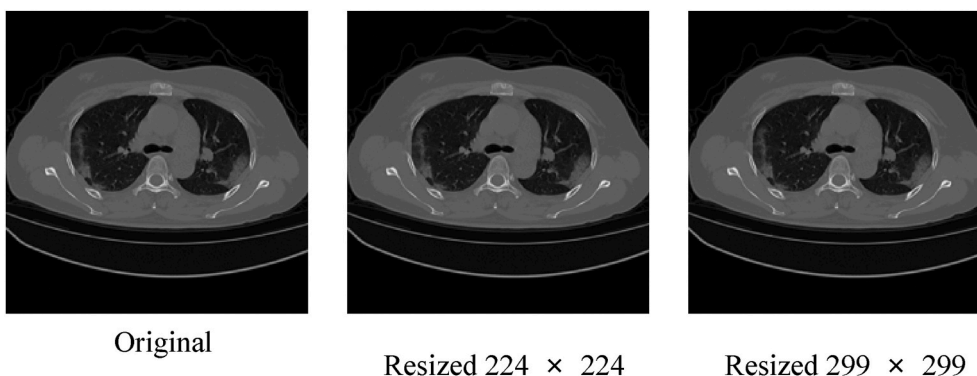
Fig. 2. Resized Chest X-ray image.

Fig. 3. Resized Chest CT image.

Step 6 Grild Search: Grid SearchCV is used to search for parameter combination.

Step 7 Perfrormance evaluation: Four performance indices are computed for the above model.

Step 8 Comparison with SOTA: The results of the above models are compared with SOTA.

Tables 1 and 2 present the number of images for each group, and 600 images are randomly selected from the total number of images for each group to be used for both datasets in this study. Among the 600 images, the number of images for training, validation and test subsets are 400, 100, and 100, respectively.

2.2. Image preprocessing

To reduce the training time and avoid memory overload, the image size was reduced to 224×224 and 299×299, and images were converted into the portable network graphics (PNG) file format in this study. Figs. 2 and 3 present the resized chest X-ray images and resized chest CT images, respectively.

2.3. Grid SearchCV

Grid SearchCV is a useful tool on fine-tuning the parameters for models. It implements grid search and cross validation through predefined hyper-parameters to fit the estimators on the training set. The parameters selected include batch size, epochs, optimizer, learning rate and dropout in this study.

2.4. Transfer learning models

Transfer learning is mainly to transfer the trained model to another new model and use it in other different but related fields, so that we do not need to start from scratch. It can not only reduce the dependence of a large amount of data, but also improve the classification performance in the target field [54–56]. Considering the number of parameter in the network architecture, InceptionV3, ResNet50V2, Xception, DenseNet121, MobileNetV2, EfficientNet-B0 models were selected in the study.

2.4.1. InceptionV3

Google proposed GoogLeNets in 2014, called it InceptionV1 [57], and then released InceptionV2 and InceptionV3 [58] the following year. InceptionV3 uses convolutional layers with stride = 2 in parallel with pooling layers to shrink feature maps. The difference from InceptionV2 is that the first Inception module of InceptionV3 is to change the 7×7 convolutional layer to three 3×3 convolutional layers. Through the above improvements, InceptionV3 increases the width and depth of the network to improve performance. In addition, the image was resized from 224×224 of the original InceptionV1 to 299×299. Therefore, this study selects InceptionV3, which has 311 layers and 21,808,931 parameters in its network architecture.

2.4.2. ResNet50V2

He et al. [59] proposed the ResNet model in 2015 and won the first place in the ILSVRC-2015 classification task. ResNet adds residual learning to solve the problem of gradient and precision drop in training

Table 3
Comparison of model parameters before and after fine-tuning.

Before fine-tuning	Parameters	After fine-tuning	Parameters
InceptionV3	21,808,931	FT-InceptionV3	913,299
ResNet50V2	23,570,947	FT-ResNet50V2	957,827
Xception	20,867,627	FT-Xception	925,083
DenseNet121	7,040,579	FT-DenseNet121	948,611
MobileNetV2	2,261,827	FT-MobileNetV2	890,851
EfficientNet-B0	4,053,414	FT-EfficientNet-B0	912,123
EfficientNetV2-S	20,181,331	FT-EfficientNetV2-S	863,895

sets, enabling the network to become deeper and deeper, while ensuring accuracy and controlling speed. The ResNet series of models also include ResNet50, ResNet101, and ResNet152. ResNet50V2 is a modified version of ResNet50 and is pre-trained on the ImageNet dataset, which outperforms ResNet50 and ResNet101. Therefore, this study chooses ResNet50V2, which has 190 layers and 25,613,800 parameters in its network architecture, which modifies the propagation formula of connections between convolution modules.

2.4.3. Xception

Google proposed Xception [60] in 2016, and its model was modified from InceptionV3, using depthwise separable convolution (DSC) to replace the original Inception module, and decomposed ordinary convolution into spatial convolution and point-by-point convolution. Spatial convolution is performed on each input channel, while pointwise convolution uses a 1×1 kernel to convolve point by point, which reduces not only the number of parameters but also the number of computations. The Xception network architecture in this study has 132 layers and 20,867,627 parameters, and is composed of 14 modules, including 36 convolutional layers. The input image size for this model is 299×299.

2.4.4. DenseNet121

Huang et al. [61] proposed the DenseNet model, whose core is mainly composed of DenseBlock (DB), transition layer and growth rate. DenseNet also includes a series of models such as DenseNet121, DenseNet169, DenseNet201 and DenseNet264. The advantage of DenseNet121 is that the model requires less parameters, so that a deeper model can be trained during calculation, and the connection of its fully connected layer also has a regularization effect, which can reduce the problem of overfitting on smaller datasets. DenseNet121 is selected in this study, which has 427 layers and 7,040,579 parameters in its

network architecture.

2.4.5. MobileNetV2

Google proposed the MobileNet model in 2017 [62] with a streamlined architecture. DSC is applied to MobileNetV1, and two global hyper-parameters are proposed to control the capacity of the network, which can effectively maintain a considerable high precision between latency and accuracy. MobileNetV2 was released in early 2018, and its architecture adds two new modules, reverse residual and linear bottleneck, which can achieve faster training speed and higher accuracy. MobileNetV2 is specially designed for images and can be used for classification and feature generation. With a total of 154 layers and 2,261,827 parameters, which is much fewer than other commonly used CNN models, MobileNetV2 is selected in this study.

2.4.6. EfficientNet-B0

In 2019, Google proposed the EfficientNet model [63], which is a new network scaling method. It uses compound coefficients to uniformly scale the depth, width and image resolution of the network. In addition, Google also developed a new baseline network using neural architecture search (NAS), and then extended it to a series of models: EfficientNet-B0 ~ EfficientNet-B7, which is somewhat similar to MnasNet Both use the same search space, but EfficientNet has a larger floating-point operations per second (FLOPS) because the main module used is MBConv. This study chooses EfficientNet-B0, which has 237 layers and 4,053,414 parameters in its network architecture.
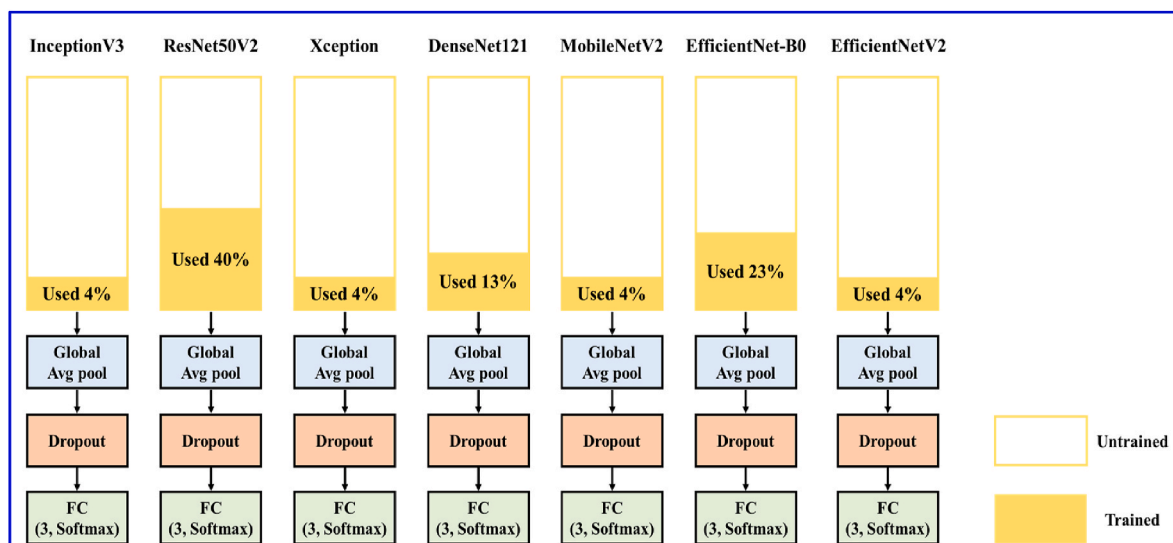
2.4.7. EfficientNetV2

EfficientNetV2, proposed by Tan & Le [64] in 2021, possesses faster training speed and better parameter efficiency. Compared to original EfficientNet, EfficientNetV2 uses both MBConv and the newly added fused-MBConv in the early layers. The expansion ratio for MBConv in EfficientNetV2 is 4, which is smaller than the expansion ratio used in EfficientNet. The advantage of smaller expansion ratio is to reduce the memory access overhead. The number of parameter of EfficientNetV2 used in this study is 20,181,331.

2.5. The proposed COVID-19 detection model

2.5.1. Fine-tuning transfer learning models

The most common ways of fine-tuning transfer learning are (1) using the pre-trained model as feature extraction, that is, shaving the output layer, and then using the remaining network as a feature extractor and

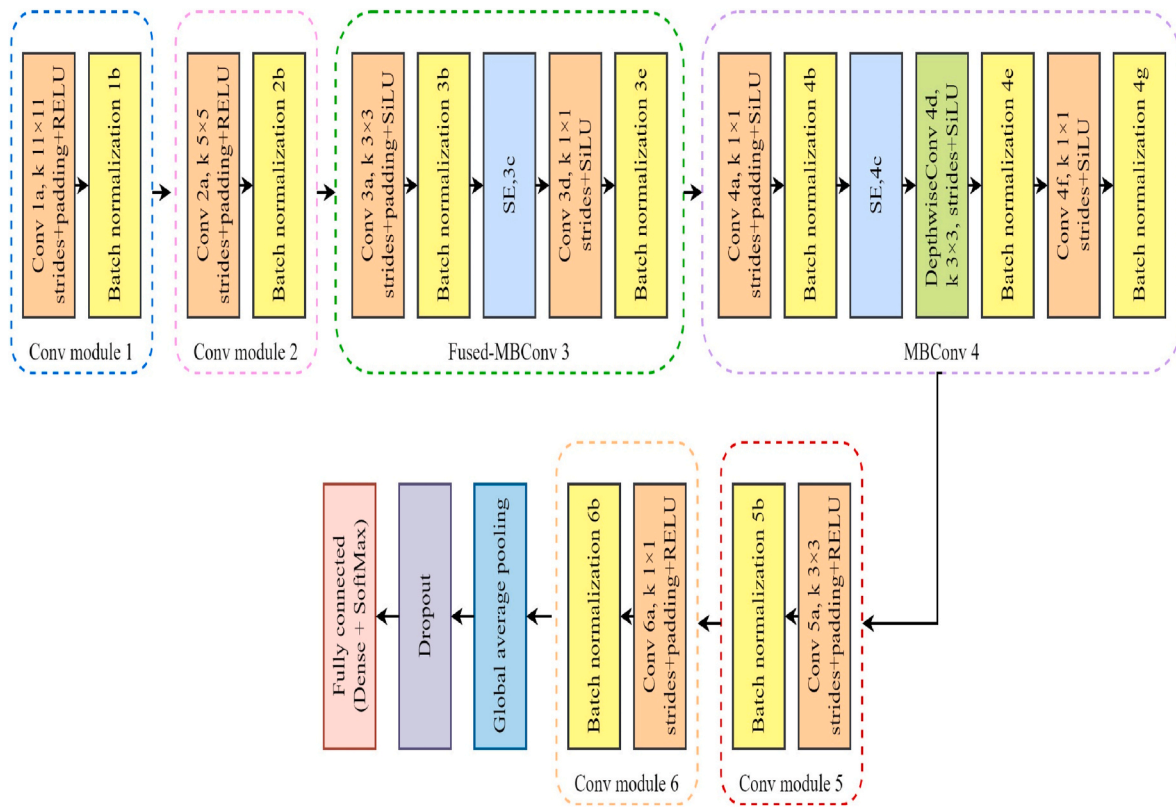**Fig. 4.** Architecture of fine-tuning transfer learning models.

Fig. 5. Architecture of LightEfficientNetV2.

Table 4
Architecture of LightEfficientNetV2.

Layer (type)		Filter	Kernel Size	Stride	Padding	Output shape
Image Input		–	–	–	–	$224 \times 224 \times 1$
Conv 1a, k 11×11		96	11×11	4	1	$56 \times 56 \times 96$
Batch normalization 1b		–	–	–	–	$56 \times 56 \times 96$
Conv 2a, k 5×5		128	5×5	1	0	$56 \times 56 \times 128$
Batch normalization 2b		–	–	–	–	$56 \times 56 \times 128$
Fused-MBConv3	Conv 3a	24	3×3	1	0	$56 \times 56 \times 24$
	Batch normalization 3b	–	–	–	–	$56 \times 56 \times 24$
	SE 3c	24	–	–	–	$56 \times 56 \times 24$
		8	–	–	–	$56 \times 56 \times 8$
	Conv 3d	24	1×1	2	–	$28 \times 28 \times 24$
	Batch normalization 3e	–	–	–	–	$28 \times 28 \times 24$
MBConv4	Conv 4a	256	1×1	1	0	$28 \times 28 \times 256$
	Batch normalization 4b	–	–	–	–	$28 \times 28 \times 256$
	SE 4c	256	–	–	–	$28 \times 28 \times 256$
		8	–	–	–	$28 \times 28 \times 8$
	DepthwiseConv 4d	128	3×3	2	–	$14 \times 14 \times 128$
	Batch normalization 4e	–	–	–	–	$14 \times 14 \times 128$
	Conv 4f	512	1×1	1	–	$14 \times 14 \times 512$
	Batch normalization 4g	–	–	–	–	$7 \times 7 \times 512$
Conv 5a, k 3×3		64	3×3	1	0	$7 \times 7 \times 64$
Batch normalization 5b		–	–	–	–	$7 \times 7 \times 64$
Conv 6a, k 1×1		64	1×1	1	0	$7 \times 7 \times 64$
Batch normalization 6b		–	–	–	–	$7 \times 7 \times 64$
Global Average Pooling		–	–	–	–	$1 \times 1 \times 64$
Dropout		–	–	–	–	–
FC		–	–	–	–	$1 \times 1 \times 3$
Softmax		–	–	–	–	$1 \times 1 \times 3$
Classification Output		–	–	–	–	–

Table 5
Parameter setting.

Parameter	Value
Batch size	8, 16, 32
Epochs	20, 50, 100, 150, 200, 250, 300
Optimizer	SGD, RMSprop, Adagrad, Adadelta, Adam, Adamax, Nadam
Learning rate	0.00001, 0.0001, 0.001, 0.01, 0.1
Dropout	0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9

applying it to new dataset. (2) The architecture of the pre-trained model is adopted, but all weights are randomized before training for its own dataset. (3) Keep the weights of the initial layers of the model unchanged and retrain the latter layers to obtain new weights. During this process, many attempts can be made to find the best match between frozen and retrained layers. The dataset used in this study is small and the image similarity is high, so we freeze the weights of the first a few layers in the pre-trained model, and then retrain the latter layers to reduce the number of parameters of each model to be less than one million. Table 3 compares the amount of parameters used before and after fine-tuning. In addition, we use a global average pooling layer to replace the original fully connected layer after the pre-training model, and finally add dropout before the classification layer. Fig. 4 shows the fine-tuned transfer learning model.

2.5.2. Proposed model - LightEfficientNetV2

The newly introduced convolutional neural network EfficientNetV2 [64] uses training-aware neural architecture search and scaling to jointly optimize training speed and parameter efficiency. Compared with EfficientNetV2, each layer of EfficientNet adopts a compound scaling method. For example, when the depth is 2, the number of layers of each layer is doubled, but the training speed and parameter amount of each layer are not consistent. At the same time, EfficientNet continues to increase the image size, which also leads to large memory consumption, which in turn affects the training speed. EfficientNetV2 uses a non-uniform scaling strategy, which can increase the number of layers in more subsequent layers.

Although many existing deep learning models have been proven to be powerful classification networks, there are still problems with a large number of parameters, which in turn leads to excessive training time. This study proposes a lightweight convolutional neural network LightEfficientNetV2 with reference to the AlexNet and EfficientNetV2 network architectures. The reasons are as follows:

1. AlexNet achieves favorable classification results when the number of network layers is small, mainly because the size of the convolution kernel in the first two convolutional layers is relatively large, which makes it possible to extract features more accurately. In addition, the activation function ReLU is added to each convolution layer as a regularization method, which helps to improve the generalization ability of the model.
2. Compared with previous models, EfficientNetV2 not only has faster training speed but also higher parameter efficiency. The related research points out that replacing the MBConv in the first to third layers of EfficientNet-B4 with Fused-MBConv will significantly improve the training speed. But if it is replaced in all layers, the amount of parameters and FLOPs will be greatly increased at this time. This shows that a proper combination of MBConv and Fused-MBConv can achieve beteer results.

Based on the above two points, we proposed a lightweight new model LightEfficientNetV2 in this study. First use the first two convolution layers of AlexNet, and then select convolution modules of MBConv and Fused-MBConv from the original EfficientNetV2, and finally add two convolution modules we constructed to form LightEfficientNetV2 in this study. With a total parameter amount of 798,539, which is significantly reduced from the original EfficientNetV2 parameter amount of 24,000,000. This not only reduces the computational cost, but also reduces the computational complexity, enabling accurate feature extraction with a small number of parameters. Fig. 5 and Table 4 present the layer-by-layer architecture of LightEfficientNetV2 used in this study, and the model is detailed as follows:

1. Padding was added to the first convolutional layer in MBConv and Fused-MBConv in this study.
2. In order to pursue higher accuracy, MBConv is the convolution module used in the Efficientnet network.
3. Fused-MBConv performs faster.
4. Batch normalization replaces the max pooling layer in the AlexNet convolutional module after each convolutional layer.
5. ReLU can retain important information in the image and is added to the last two convolution modules added by this study.
6. Adding batch normalization to the convolutional layer can make training more stable [65]. This study adds a dropout before the classification layer.

Table 6
Confusion matrix.

$$Accuracy_c = \frac{(TP_c + TN_c)}{(TP_c + FP_c + FN_c + TN_c)} \tag{1}$$

$$Precision_c = \frac{TP_c}{(TP_c + FP_c)} \tag{2}$$

$$Recall_c = \frac{TP_c}{(TP_c + FN_c)} \tag{3}$$

$$F1 - score = \frac{2}{3} \sum_{c=1}^{3} \frac{Recall_c \times Precision_c}{Recall_c + Precision_c} \tag{4}$$

Confusion matrix		Actual situation		
		COVID-19	Pneumonia	Normal
Model classification	COVID-19	COVID-19 category that are correctly classified as COVID-19.	Pneumonia category that are wrongly classified as COVID-19.	Normal category that are wrongly classified as COVID-19.
	Pneumonia	COVID-19 category that are wrongly classified as Pneumonia.	Pneumonia category that are correctly classified as Pneumonia.	Normal category that are wrongly classified as Pneumonia.
	Normal	COVID-19 category that are wrongly classified as Normal.	Pneumonia category that are wrongly classified as Normal.	Normal category that are correctly classified as Normal.

Table 7
Model accuracy on Dataset 1.

	Train	Valid	Test	Train time	Test time
InceptionV3	99.71 ± 0.11%	96.55 ± 0.54%	96.50 ± 0.82%	00:14:27	00:00:03
ResNet50V2	99.65 ± 0.25%	95.51 ± 0.99%	95.10 ± 1.93%	00:10:40	00:00:02
Xception	99.77 ± 0.14%	96.32 ± 0.50%	95.89 ± 0.35%	00:33:03	00:00:02
DenseNet121	98.45 ± 0.45%	95.14 ± 0.70%	94.67 ± 1.28%	00:14:30	00:00:03
MobileNetV2	99.56 ± 0.41%	94.87 ± 1.01%	96.36 ± 1.02%	00:08:47	00:00:01
EfficientNetB0	95.09 ± 1.11%	94.40 ± 0.60%	94.80 ± 1.68%	00:16:14	00:00:02
EfficientNetV2	99.75 ± 0.14%	97.13 ± 0.87%	94.81 ± 0.58%	00:21:32	00:00:03
FT-InceptionV3	99.75 ± 0.11%	98.10 ± 0.66%	97.72 ± 0.38%	00:12:23	00:00:03
FT-ResNet50V2	99.88 ± 0.07%	97.62 ± 0.64%	97.13 ± 0.86%	00:09:20	00:00:02
FT-Xception	99.86 ± 0.06%	97.94 ± 0.47%	97.31 ± 1.56%	00:25:25	00:00:02
FT-DenseNet121	99.90 ± 0.07%	98.05 ± 0.60%	95.41 ± 1.25%	00:12:35	00:00:03
FT-MobileNetV2	99.68 ± 0.32%	97.16 ± 0.60%	97.71 ± 1.31%	00:05:21	00:00:01
FT-EfficientNet-B0	99.13 ± 0.43%	97.32 ± 0.59%	96.77 ± 1.18%	00:12:22	00:00:02
FT-EfficientNetV2	99.78 ± 0.13%	96.65 ± 1.14%	96.73 ± 1.19%	00:09:29	00:00:02
LightEfficientNetV2	99.80 ± 0.16%	95.17 ± 0.74%	98.33 ± 0.99%	00:04:30	00:00:01

2.6. Experimental setup and implementation

In the experiment, we extract 100 images for each category as the test set, and then divide the remaining images into 80% training and 20% validation. Five-fold cross validation is used to evaluate the performance of each model. Furthermore, we use grid search to obtain the best parameters, and the values selected for each parameter are shown in Table 5. The final result is batch size of 16; epochs count of 50; optimizer of Adam, learning rate of 1e-4 and dropout of 0.2. The equipment used in the experiment is an Intel(R) Core(TM) i9-10900F

Table 8
Model performance on Dataset 1.

	Precision	Recall	F1-score
InceptionV3	96.73 ± 0.83%	96.77 ± 0.90%	96.66 ± 0.83%
ResNet50V2	95.19 ± 1.87%	95.10 ± 1.95%	95.13 ± 1.82%
Xception	96.54 ± 0.73%	96.04 ± 0.45%	96.50 ± 0.76%
DenseNet121	94.74 ± 1.45%	94.63 ± 1.33%	95.01 ± 1.29%
MobileNetV2	96.30 ± 0.99%	96.33 ± 1.01%	96.04 ± 0.80%
EfficientNetB0	94.54 ± 1.43%	94.49 ± 1.51%	94.46 ± 1.49%
EfficientNetV2	94.77 ± 0.57%	94.83 ± 0.59%	94.63 ± 0.39%
FT-InceptionV3	97.66 ± 0.46%	97.69 ± 0.40%	97.78 ± 0.58%
FT-ResNet50V2	96.77 ± 0.86%	96.89 ± 1.06%	96.87 ± 0.81%
FT-Xception	97.46 ± 1.62%	97.35 ± 1.59%	97.24 ± 2.18%
FT-DenseNet121	95.14 ± 1.11%	95.26 ± 1.31%	95.19 ± 0.77%
FT-MobileNetV2	97.56 ± 1.46%	97.65 ± 1.47%	97.44 ± 1.48%
FT-EfficientNet-B0	96.71 ± 1.00%	96.60 ± 1.10%	96.63 ± 1.13%
FT-EfficientNetV2	96.81 ± 1.12%	96.84 ± 1.12%	96.70 ± 1.18%
LightEfficientNetV2	98.22 ± 0.88%	98.30 ± 0.96%	98.22 ± 0.90%

2.81 GHz CPU, NVIDIA GeForce RTX 3070 8G GPU. The whole experiment process is performed using Python 3.8 [Python Software Foundation, Fredericksburg, Virginia, USA], which contains keras 2.6 and tensorflow 2.6.

2.7. Evaluation of model performance

This study uses a confusion matrix to analyze the performance of the models used, as shown in Table 6. The four evaluation metrics used for model evaluation are accuracy, precision, recall, and F1-score, and the mathematical formulas of these indicators are shown in Eqs. (1)–(4), respectively, where TP (true positive) represents the number of positive categories that are correctly classified as positive, FP (false positive) represents the number of negative categories that are incorrectly classified as positive, TN (true negative) refers to the number of negative categories that are correctly classified as negative, and FN (false negative) refers to the number of positive categories that are incorrectly classified as negative.

3. Results

3.1. Performance results on dataset 1

Table 7 reports the average and standard deviation of accuracies in the training, validation and test set for InceptionV3, ResNet50V2, Xception, DenseNet121, MobileNetV2, EfficientNet-B0 and EfficientNetV2 models before and after fine-tuning, and the proposed LightEfficientNetV2. The accuracy of the model after fine-tuning are higher than the accuracy of the model before fine-tuning for InceptionV3, ResNet50V2, Xception, DenseNet121, MobileNetV2, EfficientNet-B0 and EfficientNetV2 models. The training time of the model after fine-tuning is less than the training time of the model before fine-tuning for all models. LightEfficientNetV2 achieved the highest accuracy 98.33% with the least training time in the test set.

Table 8 reports the average and standard deviation of precision, recall, and F1-score in the test set for models of InceptionV3, ResNet50V2, Xception, DenseNet121, MobileNetV2, EfficientNet-B0 and EfficientNetV2 models before and after fine-tuning, and the proposed LightEfficientNetV2. The precision, recall, and F1-score of the model after fine-tuning are higher than the precision, recall, and F1-score of the model before fine-tuning for all models. The highest precision, recall and F1-score from the proposed LightEfficientNetV2 model are 98.22%, 98.30% and 98.22%, respectively.

Since models after fine-tuning have better performance than models before fine-tuning, the four performance metrics for models of InceptionV3, ResNet50V2, Xception, DenseNet121, MobileNetV2, EfficientNet-B0 and EfficientNetV2 models after fine-tuning, and the proposed LightEfficientNetV2 were compared for dataset 1 in Table 9. With the best performance, the accuracy, precision, recall and F1-score from the proposed LightEfficientNetV2 are significantly different from the other seven models. InceptionV3 model after fine-tuning scores the second place. The least standard deviations for four metrics are all from InceptionV3 model after fine-tuning, which has the best model stability.

3.2. Performance results on dataset 2

Table 10 reports the average and standard deviation of accuracies in the training, validation and testing set for InceptionV3, ResNet50V2, Xception, DenseNet121, MobileNetV2, EfficientNet-B0 and EfficientNetV2 models before and after fine-tuning, and the proposed LightEfficientNetV2. The accuracy of the model after fine-tuning is

Table 9
Performance comparison on dataset 1.

Models	Dataset 1			
	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
FT-InceptionV3	97.72 (±0.38)[b]	97.66 (±0.46)[d]	97.69 (±0.40)[e]	97.78 (±0.58)[g]
FT-ResNet50V2	97.13 (±0.86)	96.77 (±0.86)	96.89 (±1.06)	96.87 (±0.81)
FT-Xception	97.31 (±1.56)	97.46 (±1.62)	97.35 (±1.59)	97.24 (±2.18)
FT-DenseNet121	95.41 (±1.25)[b,c]	95.14 (±1.11)[d]	95.26 (±1.31)[e,f]	95.19 (±0.77)[g]
FT-MobileNetV2	97.71 (±1.31)[c]	97.56 (±1.46)	97.65 (±1.47)[f]	97.44 (±1.48)
FT-EfficientNet-B0	96.77 (±1.18)	96.71 (±1.00)	96.60 (±1.10)	96.63 (±1.13)
FT-EfficientNetV2	96.73 (±1.19)	96.81 (±1.12)	96.84 (±1.12)	96.70 (±1.18)
LightEfficientNetV2	98.33 (±0.99)[a]	98.22 (±0.88)[a]	98.30 (±0.96)[a]	98.22 (±0.90)[a]
P-value	0.00**	0.00**	0.00**	0.00**

**p < 0.01.
a Means are statistically significant from the others.
b-g Means that share a letter are significantly different.

Table 10
Model accuracy on Dataset 2.

	Train	Valid	Test	Train time	Test time
InceptionV3	99.84 ± 0.10%	92.53 ± 1.35%	93.02 ± 1.15%	00:14:31	00:00:03
ResNet50V2	99.88 ± 0.10%	93.65 ± 0.85%	93.43 ± 1.91%	00:10:33	00:00:02
Xception	98.20 ± 1.59%	93.41 ± 1.25%	93.46 ± 0.80%	00:33:06	00:00:02
DenseNet121	97.61 ± 0.88%	93.68 ± 1.30%	93.12 ± 2.21%	00:14:33	00:00:03
MobileNetV2	99.77 ± 0.13%	94.36 ± 0.59%	94.46 ± 1.44%	00:08:49	00:00:01
EfficientNetB0	94.51 ± 1.59%	92.50 ± 1.16%	93.40 ± 2.17%	00:17:32	00:00:02
EfficientNetV2	99.81 ± 0.07%	93.46 ± 0.37%	93.42 ± 0.31%	00:21:37	00:00:03
FT-InceptionV3	99.46 ± 0.35%	94.70 ± 0.45%	94.51 ± 0.76%	00:12:20	00:00:03
FT-ResNet50V2	99.84 ± 0.06%	95.64 ± 0.54%	96.01 ± 1.83%	00:09:21	00:00:02
FT-Xception	99.88 ± 0.08%	96.01 ± 0.67%	96.78 ± 1.19%	00:25:45	00:00:02
FT-DenseNet121	99.88 ± 0.05%	96.35 ± 1.03%	96.58 ± 1.07%	00:12:40	00:00:03
FT-MobileNetV2	99.87 ± 0.09%	95.58 ± 1.10%	95.82 ± 1.42%	00:05:24	00:00:01
FT-EfficientNet-B0	99.44 ± 0.16%	95.67 ± 0.39%	95.73 ± 1.05%	00:12:08	00:00:02
FT-EfficientNetV2	99.82 ± 0.09%	95.11 ± 0.83%	96.67 ± 1.35%	00:09:27	00:00:02
LightEfficientNetV2	99.89 ± 0.08%	96.65 ± 0.56%	**97.48 ± 0.85%**	00:04:30	00:00:01

higher than the accuracy of the model before fine-tuning for InceptionV3, ResNet50V2, Xception, DenseNet121, MobileNetV2, EfficientNet-B0 and EfficientNetV2 models. The training time of the model after fine-tuning is less than the training time of the model before fine-tuning for all models. LightEfficientNetV2 achieved the highest accuracy 97.48% with the least training time in the test set. Table 11 reports the average and standard deviation of precision, recall, and F1-score in the test set for models of InceptionV3, ResNet50V2, Xception, DenseNet121, MobileNetV2, EfficientNet-B0 and EfficientNetV2 models

Table 11
Model performance on Dataset 2.

	Precision	Recall	F1-score
InceptionV3	93.20 ± 1.02%	92.86 ± 1.02%	92.83 ± 0.98%
ResNet50V2	93.61 ± 2.07%	93.56 ± 2.00%	93.46 ± 2.07%
Xception	94.09 ± 0.82%	94.06 ± 0.90%	93.89 ± 0.77%
DenseNet121	93.11 ± 2.25%	93.07 ± 2.23%	93.06 ± 2.31%
MobileNetV2	94.51 ± 1.40%	94.45 ± 1.40%	94.41 ± 1.36%
EfficientNetB0	93.31 ± 2.20%	93.30 ± 2.10%	93.15 ± 2.20%
EfficientNetV2	93.46 ± 0.22%	93.48 ± 0.24%	93.34 ± 0.34%
FT-InceptionV3	94.77 ± 1.00%	94.60 ± 0.85%	94.57 ± 1.17%
FT-ResNet50V2	95.67 ± 1.90%	95.67 ± 1.86%	95.66 ± 2.07%
FT-Xception	96.43 ± 1.15%	96.52 ± 1.28%	96.48 ± 1.35%
FT-DenseNet121	96.59 ± 1.02%	96.53 ± 1.09%	96.48 ± 1.18%
FT-MobileNetV2	95.74 ± 1.22%	95.83 ± 1.47%	95.06 ± 0.67%
FT-EfficientNet-B0	95.48 ± 0.23%	95.74 ± 1.01%	95.55 ± 0.74%
FT-EfficientNetV2	96.11 ± 1.30%	96.09 ± 1.38%	95.98 ± 1.35%
LightEfficientNetV2	97.41 ± 1.03%	97.40 ± 0.96%	97.49 ± 0.92%

Table 12
Performance comparison on dataset 2.

Models	Dataset 2			
	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
FT-InceptionV3	94.51 (±0.76)[b,c,d]	94.77 (±1.00)[e]	94.60 (±0.85)[f,g]	94.57 (±1.17)[h,i]
FT-ResNet50V2	96.01 (±1.83)	95.67 (±1.90)	95.67 (±1.86)	95.66 (±2.07)
FT-Xception	96.78 (±1.19)[b]	96.43 (±1.15)	96.52 (±1.28)[g]	96.48 (±1.35)[i]
FT-DenseNet121	96.58 (±1.07)[c]	96.59 (±1.02)[e]	96.53 (±1.09)[f]	96.48 (±1.18)[h]
FT-MobileNetV2	95.82 (±1.42)	95.74 (±1.22)	95.83 (±1.47)	95.06 (±0.67)
FT-EfficientNet-B0	95.73 (±1.05)	95.48 (±0.23)	95.74 (±1.01)	95.55 (±0.74)
FT-EfficientNetV2	96.67 (±1.35)[d]	96.11 (±1.30)	96.09 (±1.38)	95.98 (±1.35)
LightEfficientNetV2	97.48 (±0.85)[a]	97.41 (±1.03)[a]	97.40 (±0.96)[a]	97.49 (±0.92)[a]
P-value	0.00**	0.00**	0.00**	0.00**

**p < 0.01.
[a]Means are statistically significant from the others.
b-g Means that share a letter are significantly different.

before and after fine-tuning, and the proposed LightEfficientNetV2. The precision, recall, and F1-score of the model after fine-tuning are higher than the precision, recall, and F1-score of the model before fine-tuning for all models. The highest precision, recall and F1-score from the proposed LightEfficientNetV2 model are 97.41%, 97.40% and 97.49%, respectively.

Since models after fine-tuning have better performance than models before fine-tuning, the four performance metrics for models of InceptionV3, ResNet50V2, Xception, DenseNet121, MobileNetV2, EfficientNet-B0 and EfficientNetV2 models after fine-tuning, and the proposed LightEfficientNetV2 were compared for dataset 2 in Table 12. With the best performance, the accuracy, precision, recall and F1-score from the proposed LightEfficientNetV2 are significantly different from the other seven models. Unlike the results found in Dataset 1, InceptionV3 model after fine-tuning does not score the second place. The performance of InceptionV3 model after fine-tuning was unfavorable among eight models.

4. Discussion

We compare seven models after fine-tuning and the proposed LightEfficientNetV2 model against the related state-of-the-art models in this section. Table 13 shows quantitative comparison results including

Table 13
Comparison of the proposed model with SOTA.

Study(s)	Dataset	Architecture	Class	Accuracy	Parameter
Panwar et al. (2020) [67]	Total: 5863 X-ray images.	nCOVnet	2	88.00%	131,842
Das et al. (2021) [40]	219 COVID-19, 1345 viral pneumonia, 1341 normal X-ray images.	TLCoV + VGG16	3	97.67%	12,410,021
Tahir et al. (2021) [41]	11596 COVID-19, 11263 Non-COVID-19, 10701 normal X-ray images.	FPN + InceptionV4	3	99.23%	4,357,000
Sheykhivand et al. (2021) [42]	2842 COVID-19, 2840 viral pneumonia, 2778 bacterial pneumonia, 2923 normal X-ray images.	Proposed DNN	4	99.50%	23,070,232
Saha et al. (2021) [5]	2300 COVID-19, 2300 normal X-ray images.	EMCNet	2	98.91%	3,955,009
Ibrahim et al. (2021) [43]	Total: 33676 (X-ray + CT images)	VGG19+ CNN	4	98.05%	22,337,604
Hussain et al. (2021) [44]	500 COVID-19, 400 viral pneumonia, 400 bacterial pneumonia, 800 normal X-ray images.	CoroDet	3	94.20%	2,873,609
Panwar et al. (2020) [45]	Total: 5856 X-ray images. Total: 3008 CT images.	Proposed CNN	2	95.61%	5,244,098
Moghaddam et al. (2021) [46]	4001 COVID-19, 5705 non-informative, 9979 normal CT images.	WCNN4	3	99.03%	4,610,531
Ahamed et al. (2021) [66]	1143 covid-19, 1150 bacterial pneumonia, 1150 viral pneumonia, 1150 normal X-ray images.	Modified & Tuned ResNet50V2	4	96.45%	49,210,756
	1000 covid-19, 1000 normal, 1000 CAP CT images.		3	99.01%	
Proposed	X-ray images 600 COVID-19, 600 pneumonia, 600 normal.	FT-EfficientNet-B0	3	96.77%	912,123
		FT-DenseNet121		95.41%	948,611
		FT-ResNet50V2		97.13%	957,827
		FT-MobileNetV2		97.71%	890,851
		FT-InceptionV3		97.81%	913,299
		FT-Xception		97.31%	925,083
		FT-EfficientNetV2		96.73%	863,895
		LightEfficientNetV2		98.33%	798,539
	CT images 600 COVID-19, 600 pneumonia, 600 normal.	FT-EfficientNet-B0	3	95.73%	912,123
		FT-DenseNet121		96.58%	948,611
		FT-ResNet50V2		96.01%	957,827
		FT-MobileNetV2		95.82%	890,851
		FT-InceptionV3		94.51%	913,299
		FT-Xception		96.78%	925,083
		FT-EfficientNetV2		96.00%	863,895
		LightEfficientNetV2		97.48%	798,539

the metrics of accuracy and parameters in its network architecture. In fact, the accuracy from our proposed LightEfficientNetV2 model was slightly inferior or close to some of the studies [5,41,42,46,66]. However, in terms of the total number of parameters used in the model, the model proposed in this study is less than those in most other studies. The least amount of parameters 131,842 used is from Ref. [67], which is less than that in our proposed LightEfficientNetV2 model in this study, but the model [55] only achieved an accuracy of 88.00% in the chest X-ray image dataset. In addition, the number of images selected for each category in our study is 600, which is much fewer than most studies. With fewer images, the training time can be greatly reduced in our models.

To verify the performance of the proposed LightEfficientNetV2 model, three more open public datasets NIH Chest X-rays (https://www.kaggle.com/nih-chest-xrays/data/code), SARS-CoV-2 (https://www.kaggle.com/plameneduardo/sarscov2-ctscan-dataset) and COVID-CT (https://github.com/UCSD-AI4H/COVID-CT) were tested and the performance metrics were displayed in Table 14. The first dataset NIH Chest X-rays is a larger dataset, which contains 15 categories, 112,120 chest X-ray images. The classification accuracies reported from related studies were around 73%–81%, and the highest accuracy 81.60% was from Guan and Huang [68] in 2020. The accuracy from our proposed LightEfficientNetV2 model is 83.42%, which dominates the related studies.

The second dataset SARS-CoV-2 contains 2 categories with 2,482 CT images in total. With accuracies higher than 95%, the performance of related studies are favorable. The accuracy of 98.99% was from Shaik and Cherukuri [69] using ensemble learning model. Even with fewer parameter in the network architecture, our proposed Light-EfficientNetV2 model scores the highest accuracy of 99.47%. The third dataset COVID-CT contains 2 categories with 746 CT images in total. Most of the accuracies from related studies are less than 90%, except the one 93.33% from Shaik and Cherukuri [69]. Although the accuracy from our proposed LightEfficientNetV2 model is 88.67%, which is lower than that in Ref. [69], the parameter in LightEfficientNetV2 model is much less than those in Refs. [70,71].

5. Conclusion

Based on AlexNet and EfficientNetV2, a light-weight convolutional neural network LightEfficientNetV2 is introduced on classification of COVID-19, Pneumonia and Normal images. With less number of parameters used in the network during the training phase as compared to state-of-the-art models, the proposed LightEfficientNetV2 model achieves high accuracies of 98.33% and 97.48% on chest X-ray image and CT image, respectively. LightEfficientNetV2 saves computation and running time as compared with the original models. LightEfficientNetV2

Table 14

Comparisons on NIH Chest X-rays, SARS-CoV-2 and COVID-CT datasets.

Dataset	Study(s)	Accuracy	Precision	Recall	F1-score	Parameter
NIH Chest X-rays	Wang et al. (2017) [72]	74.50%	–	–	–	–
	Yao et al. (2018) [73]	76.10%	–	–	–	–
	Guendel et al. (2018) [74]	80.70%	–	–	–	–
	Li et al. (2018) [75]	75.50%	–	–	–	–
	Shen and Gao (2018) [76]	77.50%	–	–	–	–
	Tang et al. (2018) [77]	80.30%	–	–	–	–
	Liu et al. (2019) [3]	81.50%	–	–	–	–
	Bharati et al. (2020) [78]	73.00%	69.00%	63.00%	68.00%	15,488,051
	Guan and Huang (2020) [68]	81.60%	–	–	–	–
	Proposed (LightEfficientNetV2)	**83.42%**	**81.94%**	**82.76%**	**82.39%**	**798,539**
SARS-CoV-2	Özkaya et al. (2019) [79]	96.00%	–	–	–	–
	Soares et al. (2020) [80]	97.30%	99.10%	95.50%	97.30%	–
	Panwar et al. (2020) [45]	95.00%	95.30%	94.00%	94.30%	5,244,098
	Jaiswal et al. (2021) [81]	96.20%	–	–	–	20,242,984
	Shaik and Cherukuri (2021) [69]	98.99%	98.98%	99.00%	98.99%	–
	Proposed (LightEfficientNetV2)	**99.47%**	**99.19%**	**99.25%**	**99.04%**	**798,539**
COVID-CT	Mishra et al. (2020) [82]	88.30%	–	–	86.70%	–
	Saqib et al. (2020) [83]	80.30%	78.20%	85.70%	81.80%	–
	He et al. (2020) [70]	86.00%	–	–	85.00%	14,149,480
	Mobiny et al. (2020) [84]	–	84.00%	–	–	–
	Polsinelli et al. (2020) [71]	85.00%	85.00%	87.00%	86.00%	12,600,000
	Yang et al. (2020) [85]	–	–	89.10%	–	25,600,000
	Cruz (2021) [24]	86.00%	–	89.00%	85.00%	–
	Shaik and Cherukuri (2021) [69]	**93.33%**	**93.17%**	**93.54%**	**93.29%**	–
	Proposed (LightEfficientNetV2)	88.67%	87.28%	87.43%	87.55%	**798,539**

excels favorable performance on three other public datasets NIH Chest X-rays, SARS-CoV-2 and COVID-CT.

In addition to classification of three groups of COVID-19, Pneumonia and Normal images in this study, Bacterial Pneumonia, Tuberculosis, or Chest Abdomen Pelvis are encouraged to be considered in the future study. Although the model parameter of the proposed LightEfficientNetV2 is much less than those used in the related study, the accuracy achieved from LightEfficientNetV2 is not the best as compared to the related works. Using the advanced ensemble learning network which reduces the misclassification rate from single models might be a direction in the future research.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] K.H. Shibly, S.K. Dey, M.T.U. Islam, M.M. Rahman, COVID faster R–CNN: a novel framework to Diagnose Novel Coronavirus Disease (COVID-19) in X-Ray images, Inform. Med. Unlocked 20 (2020), 100405, https://doi.org/10.1016/j.imu.2020.100405.

[2] M.F. Aslan, M.F. Unlersen, K. Sabanci, A. Durdu, CNN-based transfer learning–BiLSTM network: a novel approach for COVID-19 infection detection, Appl. Soft Comput. 98 (2021), 106912, https://doi.org/10.1016/j.asoc.2020.106912.

[3] H. Liu, L. Wang, Y. Nan, F. Jin, Q. Wang, J. Pu, SDFN: segmentation-based deep fusion network for thoracic disease classification in chest X-ray images, Comput. Med. Imag. Graph. 75 (2019) 66–73, https://doi.org/10.1016/j.compmedimag.2019.05.005.

[4] A. Bhattacharyya, D. Bhaik, S. Kumar, P. Thakur, R. Sharma, R.B. Pachori, A deep learning based approach for automatic detection of COVID-19 cases using chest X-ray images, Biomed. Signal Process Control 71 (PB) (2022), 103182, https://doi.org/10.1016/j.bspc.2021.103182.

[5] P. Saha, M.S. Sadi, M.M. Islam, EMCNet: automated COVID-19 diagnosis from X-ray images using convolutional neural network and ensemble of machine learning classifiers, Inform. Med. Unlocked 22 (2021), 100505, https://doi.org/10.1016/j.imu.2020.100505.

[6] D. Arias-Garzón, et al., COVID-19 detection in X-ray images using convolutional neural networks, Mach. Learning Appl. 6 (April) (2021), 100138, https://doi.org/10.1016/j.mlwa.2021.100138.

[7] N. Science, C. Phenomena, A. Altan, S. Karasu, Chaos , Solitons and Fractals Recognition of COVID-19 disease from X-ray images by hybrid model consisting of

2D curvelet transform, Chaotic Salp Swarm Algorithm Deep Learn. Techniq. 140 (2020), https://doi.org/10.1016/j.chaos.2020.110071.

[8] N. Rashid, M.A.F. Hossain, M. Ali, M. Islam Sukanya, T. Mahmud, S.A. Fattah, AutoCovNet: unsupervised feature learning using autoencoder and feature merging for detection of COVID-19 from chest X-ray images, Biocybern. Biomed. Eng. (2021), https://doi.org/10.1016/j.bbe.2021.09.004.

[9] R. Hertel, R. Benlamri, COV-SNET: a deep learning model for X-ray-based COVID-19 classification, Inform. Med. Unlocked 24 (April) (2021), 100620, https://doi.org/10.1016/j.imu.2021.100620.

[10] M.F. Aslan, K. Sabanci, A. Durdu, M.F. Unlersen, COVID-19 diagnosis using state-of-the-art CNN architecture features and Bayesian Optimization, in: Computers in Biology and Medicine, 142, Elsevier Ltd, 2022, https://doi.org/10.1016/j.compbiomed.2022.105244. Mar. 01.

[11] B. Nigam, A. Nigam, R. Jain, S. Dodia, N. Arora, B. Annappa, COVID-19: automatic detection from X-ray images by utilizing deep learning methods, Expert Syst. Appl. 176 (2021), 114883, https://doi.org/10.1016/j.eswa.2021.114883. January.

[12] I.D. Apostolopoulos, T.A. Mpesiana, Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks, Phys. Eng. Sci. Med. 43 (2) (2020) 635–640, https://doi.org/10.1007/s13246-020-00865-4.

[13] G. Marques, D. Agarwal, I. de la Torre Díez, Automated medical diagnosis of COVID-19 through EfficientNet convolutional neural network, Appl. Soft CompuT. J. 96 (2020), 106691, https://doi.org/10.1016/j.asoc.2020.106691.

[14] A. Gupta, Anjum, S. Gupta, R. Katarya, InstaCovNet-19: a deep learning classification model for the detection of COVID-19 patients using Chest X-ray, Appl. Soft Comput. 99 (2021), https://doi.org/10.1016/j.asoc.2020.106859. Feb.

[15] M. Rahimzadeh, A. Attar, New modified deep convolutional neural network for detecting covid-19 from X-ray images, a 19, 2020, 100360, https://doi.org/10.1016/j.imu.2020.100360. *arXiv*.

[16] S.M.J. Jalali, M. Ahmadian, S. Ahmadian, A. Khosravi, M. Alazab, S. Nahavandi, An oppositional-Cauchy based GSK evolutionary algorithm with a novel deep ensemble reinforcement learning strategy for COVID-19 diagnosis, Appl. Soft Comput. 111 (March 2020) (2021), 107675, https://doi.org/10.1016/j.asoc.2021.107675.

[17] J.L. Gayathri, B. Abraham, M.S. Sujarani, M.S. Nair, A computer-aided diagnosis system for the classification of COVID-19 and non-COVID-19 pneumonia on chest X-ray images by integrating CNN with sparse autoencoder and feed forward neural network, Comput. Biol. Med. 141 (Feb) (2022), https://doi.org/10.1016/j.compbiomed.2021.105134.

[18] S.-H. Wang, X. Zhang, and Y.-D. Zhang, "DSSAE: deep stacked sparse autoencoder analytical model for COVID-19 diagnosis by fractional fourier entropy," ACM Trans. Manag. Inf. Syst., vol. 13, no. 1, pp. 1–20, Mar. 2022, doi: 10.1145/3451357.

[19] V.S. Rohila, N. Gupta, A. Kaul, D.K. Sharma, Deep learning assisted COVID-19 detection using full CT-scans, Internet of Things (Netherlands) 14 (2021), 100377, https://doi.org/10.1016/j.iot.2021.100377.

[20] 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. 121 (March) (2020), 103795, https://doi.org/10.1016/j.compbiomed.2020.103795.

[21] Y. Fu, P. Xue, E. Dong, Densely connected attention network for diagnosing COVID-19 based on chest CT, Comput. Biol. Med. 137 (June) (2021), https://doi.org/10.1016/j.compbiomed.2021.104857.

[22] I. Lahsaini, M. El Habib Daho, M.A. Chikh, Deep transfer learning based classification model for covid-19 using chest CT-scans, Pattern Recogn. Lett. 152 (2021) 122–128, https://doi.org/10.1016/j.patrec.2021.08.035.

[23] V. Arora, E. Y, K. Ng, R.S. Leekha, M. Darshan, A. Singh, Transfer learning-based approach for detecting COVID-19 ailment in lung CT scan, Comput. Biol. Med. 135 (April) (2021), 104575, https://doi.org/10.1016/j.compbiomed.2021.104575.

[24] A.L. Aswathy, A.H. S, V.C. Vinod, COVID-19 diagnosis and severity detection from CT-images using transfer learning and back propagation neural network, J. Infection Publ. Health 14 (10) (2021) 1435–1445, https://doi.org/10.1016/j.jiph.2021.07.015.

[25] X. Fan, X. Feng, Y. Dong, H. Hou, COVID-19 CT Image Recognition Algorithm Based on Transformer and CNN," *Displays*, 72, Elsevier B.V., 2022, https://doi.org/10.1016/j.displa.2022.102150. Apr. 01.

[26] Y.D. Zhang, M.A. Khan, Z. Zhu, S.H. Wang, Pseudo zernike moment and deep stacked sparse autoencoder for COVID-19 diagnosis, Comput. Mater. Continua (CMC) 69 (3) (2021) 3146–3162, https://doi.org/10.32604/cmc.2021.018040.

[27] J.F. Hernández Santa Cruz, An ensemble approach for multi-stage transfer learning models for COVID-19 detection from chest CT scans, Intelligence-Based Med. 5 (February) (2021), 100027, https://doi.org/10.1016/j.ibmed.2021.100027.

[28] T. Zhou, H. Lu, Z. Yang, S. Qiu, B. Huo, Y. Dong, The ensemble deep learning model for novel COVID-19 on CT images, Appl. Soft Comput. 98 (2021) 106885, https://doi.org/10.1016/j.asoc.2020.106885.

[29] E. Jangam, C.S.R. Annavarapu, A stacked ensemble for the detection of COVID-19 with high recall and accuracy, Comput. Biol. Med. 135 (June) (2021), 104608, https://doi.org/10.1016/j.compbiomed.2021.104608.

[30] B. Abraham, M.S. Nair, Computer-aided detection of COVID-19 from CT scans using an ensemble of CNNs and KSVM classifier, Signal, Image and Video Processing 16 (3) (Apr. 2022) 587–594, https://doi.org/10.1007/s11760-021-01991-6.

[31] D.N. Vinod, B.R. Jeyavadhanam, A.M. Zungeru, S.R.S. Prabaharan, Fully automated unified prognosis of Covid-19 chest X-ray/CT scan images using Deep Covix-Net model, Comput. Biol. Med. 136 (August) (2021), 104729, https://doi.org/10.1016/j.compbiomed.2021.104729.

[32] J. Li, et al., Multi-task contrastive learning for automatic CT and X-ray diagnosis of COVID-19, Pattern Recogn. 114 (2021), 107848, https://doi.org/10.1016/j.patcog.2021.107848.

[33] S. Thakur, A. Kumar, X-ray and CT-scan-based automated detection and classification of covid-19 using convolutional neural networks (CNN), Biomed. Signal Process Control 69 (March) (2021), 102920, https://doi.org/10.1016/j.bspc.2021.102920.

[34] H. Munusamy, J.M. Karthikeyan, G. Shriram, S. Thanga Revathi, S. Aravindkumar, FractalCovNet architecture for COVID-19 chest X-ray image classification and CT-scan image segmentation, Biocybern. Biomed. Eng. 41 (3) (2021) 1025–1038, https://doi.org/10.1016/j.bbe.2021.06.011.

[35] A. Banerjee, R. Bhattacharya, V. Bhateja, P.K. Singh, A. Lay-Ekuakille, R. Sarkar, COFE-Net: an ensemble strategy for Computer-Aided Detection for COVID-19, Measurement 187 (October 2021) (2021), 110289, https://doi.org/10.1016/j.measurement.2021.110289.

[36] Y.D. Zhang, Z. Zhang, X. Zhang, S.H. Wang, MIDCAN: a multiple input deep convolutional attention network for Covid-19 diagnosis based on chest CT and chest X-ray, Pattern Recogn. Lett. 150 (Oct. 2021) 8–16, https://doi.org/10.1016/j.patrec.2021.06.021.

[37] M. la Salvia, et al., Deep learning and lung ultrasound for Covid-19 pneumonia detection and severity classification, Comput. Biol. Med. 136 (Sep. 2021), https://doi.org/10.1016/j.compbiomed.2021.104742.

[38] S. Erfanian Ebadi, et al., Automated detection of pneumonia in lung ultrasound using deep video classification for COVID-19, Inform. Med. Unlocked 25 (Jan. 2021), https://doi.org/10.1016/j.imu.2021.100687.

[39] Q. Huang, et al., Evaluation of pulmonary edema using ultrasound imaging in patients with COVID-19 pneumonia based on a non-local Channel attention ResNet, Ultrasound Med. Biol. 48 (5) (May 2022) 945–953, https://doi.org/10.1016/j.ultrasmedbio.2022.01.023.

[40] A. K. Das, S. Kalam, C. Kumar, and D. Sinha, "TLCoV- an automated Covid-19 screening model using Transfer Learning from chest X-ray images," Chaos, Solit. Fractals, vol. 144, 2021, doi: 10.1016/j.chaos.2021.110713.

[41] A.M. Tahir, et al., COVID-19 infection localization and severity grading from chest X-ray images, Comput. Biol. Med. 139 (October) (2021), 105002, https://doi.org/10.1016/j.compbiomed.2021.105002.

[42] S. Sheykhivand, et al., Developing an efficient deep neural network for automatic detection of COVID-19 using chest X-ray images, Alex. Eng. J. 60 (3) (2021) 2885–2903, https://doi.org/10.1016/j.aej.2021.01.011.

[43] D.M. Ibrahim, N.M. Elshennawy, A.M. Sarhan, Deep-chest: multi-classification deep learning model for diagnosing COVID-19, pneumonia, and lung cancer chest diseases, Comput. Biol. Med. 132 (2021), 104348, https://doi.org/10.1016/j.compbiomed.2021.104348.

[44] E. Hussain, M. Hasan, M.A. Rahman, I. Lee, T. Tamanna, M.Z. Parvez, CoroDet: a deep learning based classification for COVID-19 detection using chest X-ray images, Chaos, Solit. Fractals 142 (2021), 110495, https://doi.org/10.1016/j.chaos.2020.110495.

[45] H. Panwar, P.K. Gupta, M.K. Siddiqui, R. Morales-Menendez, P. Bhardwaj, V. Singh, 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 (2020), 110190, https://doi.org/10.1016/j.chaos.2020.110190.

[46] S.M. JavadiMoghaddam, H. Gholamalinejad, A novel deep learning based method for COVID-19 detection from CT image, Biomed. Signal Process Control 70 (December 2020) (2021), 102987, https://doi.org/10.1016/j.bspc.2021.102987.

[47] K. Hu, et al., Deep supervised learning using self-adaptive auxiliary loss for COVID-19 diagnosis from imbalanced CT images, Neurocomputing 458 (2021) 232–245, https://doi.org/10.1016/j.neucom.2021.06.012.

[48] K.U. Ahamed, et al., A deep learning approach using effective preprocessing techniques to detect COVID-19 from chest CT-scan and X-ray images, Comput. Biol. Med. (2021), 105014, https://doi.org/10.1016/j.compbiomed.2021.105014.

[49] P. Saha, M.S. Sadi, M. Islam, Informatics in Medicine Unlocked EMCNet : automated COVID-19 diagnosis from X-ray images using convolutional neural network and ensemble of machine learning classifiers, Inform. Med. Unlocked 22 (2021), 100505, https://doi.org/10.1016/j.imu.2020.100505.

[50] Chest X-ray images (pneumonia) | Kaggle. https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia. (Accessed 4 November 2021).

[51] COVID-19 Radiography database | Kaggle. https://www.kaggle.com/tawsifurrahman/covid19-radiography-database. (Accessed 4 November 2021).

[52] Large COVID-19 CT scan slice dataset | Kaggle. https://www.kaggle.com/maedemaftouni/large-covid19-ct-slice-dataset. (Accessed 9 November 2021).

[53] Mendeley data - chest CT of multiple pneumonia. https://data.mendeley.com/datasets/38hk7dcp47/2. (Accessed 9 November 2021).

[54] F. Zhuang, et al., A comprehensive survey on transfer learning, Proc. IEEE 109 (1) (2021) 43–76, https://doi.org/10.1109/JPROC.2020.3004555.

[55] Z. Jiang, Z. Dong, W. Jiang, Y. Yang, Recognition of rice leaf diseases and wheat leaf diseases based on multi-task deep transfer learning, Comput. Electron. Agric. 186 (December 2020) (2021), 106184, https://doi.org/10.1016/j.compag.2021.106184.

[56] ImageNet - Wikipedia. https://en.wikipedia.org/wiki/ImageNet. (Accessed 1 April 2022).

[57] C. Szegedy, et al., Going deeper with convolutions, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9, https://doi.org/10.1109/CVPR.2015.7298594, 07-12-June.

[58] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016, pp. 2818–2826, https://doi.org/10.1109/CVPR.2016.308, 2016-Decem.

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

[60] F. Chollet, Xception: deep learning with depthwise separable convolutions, 2017, in: Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2017, pp. 1800–1807, https://doi.org/10.1109/CVPR.2017.195, 2017-Janua.

[61] G. Huang, Z. Liu, L. van der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, 2017, pp. 2261–2269, https://doi.org/10.1109/CVPR.2017.243, 2017-Janua.

[62] A.G. Howard, et al., MobileNets: efficient convolutional neural networks for mobile vision applications [Online]. Available: http://arxiv.org/abs/1704.04861, 2017.

[63] M. Tan, Q.v. Le, EfficientNet: rethinking model scaling for convolutional neural networks, 2019, in: 36th International Conference on Machine Learning, ICML, 2019, pp. 10691–10700, 2019-June.

[64] M. Tan, Q.v. Le, EfficientNetV2: smaller models and faster training [Online]. Available: http://arxiv.org/abs/2104.00298, Apr. 2021.

[65] Dropout in (deep) machine learning | by amar Budhiraja | Medium. https://medium.com/@amarbudhiraja/https-medium-com-amarbudhiraja-learning-less-to-learn-better-dropout-in-deep-machine-learning-74334da4bfc5. (Accessed 3 May 2022).

[66] K.U. Ahamed, et al., A deep learning approach using effective preprocessing techniques to detect COVID-19 from chest CT-scan and X-ray images, Comput. Biol. Med. 139 (November) (2021), 105014, https://doi.org/10.1016/j.compbiomed.2021.105014.

[67] H. Panwar, P.K. Gupta, M.K. Siddiqui, R. Morales-Menendez, V. Singh, Application of deep learning for fast detection of COVID-19 in X-Rays using nCOVnet, Chaos, Solit. Fractals 138 (2020) 109944, https://doi.org/10.1016/j.chaos.2020.109944.

[68] Q. Guan, Y. Huang, Multi-label chest X-ray image classification via category-wise residual attention learning, Pattern Recogn. Lett. 130 (2020) 259–266, https://doi.org/10.1016/j.patrec.2018.10.027.

[69] N.S. Shaik, T.K. Cherukuri, Transfer learning based novel ensemble classifier for COVID-19 detection from chest CT-scans, Comput. Biol. Med. 141 (June 2021) (2021), 105127, https://doi.org/10.1016/j.compbiomed.2021.105127.

[70] X. He, et al., Sample-efficient deep learning for COVID-19 diagnosis based on CT scans, IEEE Trans. Med. Imag. (2020), https://doi.org/10.1101/2020.04.13.20063941 vol. XX, no. Xx.

[71] M. Polsinelli, L. Cinque, G. Placidi, A light CNN for detecting COVID-19 from CT scans of the chest, Pattern Recogn. Lett. 140 (2020) 95–100, https://doi.org/10.1016/j.patrec.2020.10.001.

[72] 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 - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2017, pp. 3462–3471, https://doi.org/10.1109/CVPR.2017.369, 2017-Janua.

[73] L. Yao, J. Prosky, E. Poblenz, B. Covington, K. Lyman, Weakly supervised medical diagnosis and localization from multiple resolutions [Online]. Available: http://arxiv.org/abs/1803.07703, 2018.

6

{