

**IJECBE**

International Journal of Electrical, Computer and Biomedical Engineering

*IJECBE* (2023), 1, 2, 168–178  
Received (14 June 2023) / Revised (23 June 2023)  
Accepted (04 July 2023) / Published (30 December 2023)  
<https://doi.org/10.62146/ijecbe.v1i2.14>  
<https://ijecbe.ui.ac.id>  
ISSN 3026-5258

## RESEARCH ARTICLE

# Implementation of Xception and EfficientNetB3 for COVID-19 Detection on Chest X-Ray Image via Transfer Learning

Nadya Novalina and Mia Rizkinia<sup>\*</sup>

Department of Electrical Engineering, Faculty of Engineering, Universitas Indonesia, Depok, 16424, Indonesia

<sup>\*</sup>Corresponding author. Email: [mia@ui.ac.id](mailto:mia@ui.ac.id)

### Abstract

COVID-19 is a highly contagious infectious disease caused by the SARS-CoV-2 virus that can cause respiratory issues. The utilization of X-ray imaging has the potential to serve as an alternative means of detecting COVID-19 by offering insights into the condition of the lungs. Rapid and automated analysis of medical images and patterns can be achieved through deep learning techniques. In this study, we propose methods for the automatic classification of COVID-19 from Chest X-Ray images using CNN with transfer learning techniques, namely Xception and EfficientNetB3 architectures, as well as an ensemble of both architectures working in parallel. Additionally, we use Grad-CAM to visualize the regions of the X-ray image that are most important for the classification decision. The classification of COVID-19 is carried out for four types of classes: COVID-19, normal, bacterial pneumonia, and viral pneumonia. The proposed classifier models achieve an overall accuracy of 94.44% for the Xception classifier, 95.28% for the EfficientNetB3 classifier, and 94.44% for the parallel classifier. The accuracy value is higher than the other comparison classifiers accuracy values. Overall, the proposed classifiers can be recommended as tools to assist radiologists and clinical practitioners in diagnosing and following up with COVID-19 cases.

**Keywords:** COVID-19, CXR, classification, Convolutional Neural Network, transfer learning

## 1. Introduction

Coronavirus disease 2019 (COVID-19) cases were first reported in Wuhan, China, in December 2019, and on March 11, 2020, World Health Organization (WHO) declared it a pandemic [1]. With the increasing number of COVID-19 infections,

there is a need for immediate and less invasive diagnostic tools. Commonly used tests for detection include antigen, immunoenzymatic serological, and reverse-transcription polymerase chain reaction (RT-PCR) molecular tests [2]. Another alternative for the diagnosis and validation of COVID-19 is the use of non-invasive radiological imaging, such as computed tomography (CT) and chest X-ray (CXR) [3].

CXR is used because it is fast, cost-effective, and can detect pulmonary and alveolar interstitial opacities in patients with COVID-19 symptoms [1, 2]. The portable CXR can also be used for easy, timely access in remote areas where access to healthcare is limited [4]. But some lung abnormalities cannot be detected using CXR [2]. To overcome this limitation, deep learning (DL) is used for medical image analysis; it can automatically analyze, identify, and classify patterns in medical images. One of the most widely used DL models for classifying and detecting medical images is the convolutional neural network (CNN) that has been used to detect various diseases and lesions on the body automatically in recent years [3].

Several studies have been conducted to classify COVID-19 using CXR images. In one study [5], a CNN model called CoroNet with a transfer learning architecture Xception achieved an overall accuracy of 89.6% for four classes: COVID-19, normal, bacterial pneumonia, and viral pneumonia. Another study [6] used a combination of two CNN architectures, Xception and ResNet50V2, and achieved an accuracy of 91.4% for the same four classes. In a third study [7], the transfer learning architecture of EfficientNetB4 was used, and an overall accuracy of 96.70% for three classes, COVID-19, normal, and viral pneumonia, was achieved.

In this study, we use a deep learning approach with the transfer learning CNN model, namely Xception and EfficientNetB3 architectures, and the ensemble of the two architectures arranged in parallel for the classification of COVID-19 using CXR images. Xception and EfficientNetB3 architectures are used because they have better accuracy than other architectures after researching several possible CNN architectures that can be used. The classification of COVID-19 is carried out for four types of classes: COVID-19, normal, bacterial pneumonia, and viral pneumonia.

## 2. Materials and Methods

### 2.1 Dataset

The COVID-19 dataset comes from the public data “Curated Dataset for COVID-19 Posterior–Anterior Chest Radiography Images (X–Rays)” Version 3 [8] with a Postero–Anterior view of 4000 CXR images. The amount of data per class has the same amount so that the data used is balanced. From the total data of 4000 CXR images, then the data is divided by a ratio of 90:10 for training–validation data (3600 CXR images) and test data (400 CXR images). A summary of the data used is presented in TABLE I.

### 2.2 Data Splitting

Data splitting is done by splitting the training–validation data into several types of data splitting: data ratios of 70:30, 80:20, 90:10, and 5-fold cross-validation. Split data 70:30, 80:20, and 90:10 were performed by adding the validation of the split parameter to the data augmentation settings using the ImageDataGenerator function from the Keras

**Table 1.** Summary of COVID-19 Datasets

<b>Dataset</b>	<b>Classes</b>	<b>Images</b>
Curated Dataset for COVID-19 Posterior-Anterior Chest Radiography Images (X-Rays) Version 3	COVID-19	1000
	Normal	1000
	Bacterial Pneumonia	1000
	Viral Pneumonia	1000
	<b>Total</b>	4000

library. 5-fold cross-validation was performed using the StratifiedKFold function from the sklearn library so that the data will automatically be divided into five-fold.

### 2.3 Pre-processing

Pre-processing is the transformation of raw data before it is entered or used in the model. The process is done by resizing the data size to  $224 \times 224 \times 3$  pixels to ensure consistent dimensions across all samples. This resizing step helps in achieving uniformity in data representation. The data augmentation is done to overcome overfitting and increase the accuracy of the proposed model. This technique introduces synthetic variations to the dataset, increasing its diversity and reducing the risk of the model memorizing specific patterns. Data augmentation operations include rescale, rotation\_range, height\_shift\_range, width\_shift\_range, shear\_range, zoom\_range, horizontal\_flip, vertical\_flip, and fill\_mode. The data is then randomized to generalize the model and reduce overfitting [9, 10].

For the training process, this study used 60 epochs and 16 batch sizes, which were determined after experimenting on several different epochs and batch sizes combinations. To prevent overfitting and ensure effective training, an early stopping callback is employed [11]. Loss function categorical cross-entropy for multi-class classification is used in accordance with the classification carried out in the study [9]. The optimizer used is the Adam optimizer, with a learning rate of 0.0001 [12].

By following these pre-processing steps and training configurations, the model is better equipped to handle the classification task, leading to improved accuracy and generalization capabilities.

### 2.4 Architectural Model Selection

Experiments were conducted to determine the best CNN architecture for the proposed research model. Experiments were carried out in Kaggle kernels [13] with the proposed layer modification and 60 epochs on the CXR dataset that will be used for this study. Data is divided by a ratio of 80:20 for training and validation data. The experimental results obtained the highest average validation accuracy value of all test architectures on the Xception and EfficientNetB3 architectures at 94.31%. Based on this experiment, we decided to use CNN transfer learning using the Xception and EfficientNetB3 architectures and the ensemble transfer learning of the two architectures arranged in parallel.

## 2.5 Model Architecture and Development

Three classifier models are developed based on the Xception, EfficientNetB3, and an ensemble of the two architectures arranged in parallel. The base model has been pre-trained on ImageNet [14] for Xception and with noisy-student weights [15] for EfficientNetB3. Xception or Extreme Inception is a deep separable convolution layer with residual connections with 36 convolution layers organized into 14 modules. The residual connections are "skip connections" that allow the gradient to flow through the network directly, avoiding the problem of missing gradients [16]. EfficientNet is a CNN architecture developed using a scaling method that uniformly scales all width, depth, and resolution dimensions using compound coefficients. The compound coefficient method is based on the idea of balancing the dimensions of width, depth, and resolution with constant-ratio scaling [17].

To enhance the performance of our models, we introduced several innovative modifications to the base architectures, taking inspiration from the work of G. Marques et al. [7]. Their research showcased the effectiveness of incorporating global average pooling, dense layers, and dropout layers after the transfer learning model. Building upon their findings, we tailored our models by incorporating these modifications alongside additional enhancements. Specifically, we integrated global average pooling, fully connected layer (dense layer) with the rectified linear unit (ReLU) activation function, dropout layer, batch normalization, and output layer (dense layer) utilizing the softmax activation function for multi-class classification. The model architecture, illustrating these modifications, is depicted in Fig. 1.

Global average pooling was chosen as it can summarize spatial information without introducing additional parameters, which can help prevent overfitting [18]. The dense layer with 256 nodes was selected as a hidden layer with the ReLU activation function after experimenting with different numbers of nodes. It was found that using 256 nodes resulted in better accuracy compared to other node variations. The ReLU activation function is commonly used in CNN because it has a lower computational load and is faster than sigmoid and tanh [9]. Dropout and batch normalization are also utilized for regularization to avoid overfitting. A dropout rate of 0.5 is used as it provides the highest regularization rate [19]. The output layer with softmax activation function is used with the number of nodes depending on each class to be classified. The design of this layer modification is determined after selecting several different layer arrangements and selecting the layer combined with the highest accuracy among other layer modification designs.

These modifications represent a novel contribution to the field by combining established techniques with innovative adaptations and optimizations. The CNN model architecture we developed introduces a new design that sets it apart from existing state-of-the-art models. Through extensive experimentation, we fine-tuned the layer configuration to achieve superior accuracy in classifying COVID-19, pneumonia virus, pneumonia bacterial, and normal cases. These enhancements showcase a distinct layer design, setting our models apart from other state-of-the-art models.

The design and training of the proposed classifier model will be carried out on the Kaggle kernels [13] using a GPU NVIDIA TESLA P100 VGA, 13GB RAM, and the Python programming language.

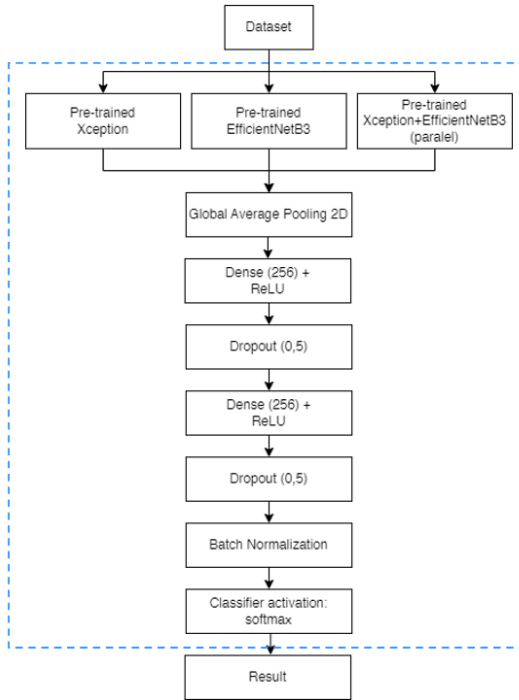


Figure 1. Overview of the proposed model architectures

## 2.6 Evaluation Metrics

The model is evaluated by calculating the accuracy, precision, recall, and F1-score on validation and test data. The evaluation results were compared using a confusion matrix and classification report to calculate the accuracy, precision, recall, and F1-score level. Gradient-weighted Class Activation Mapping (Grad-CAM) [20] is also used to provide color visualization of lung areas infected by COVID-19 and pneumonia using test data.

The values of accuracy, precision, recall, and f1-score are calculated using the following equation:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$recall (sensitivity) = \frac{TP}{TP+FN} \quad (2)$$

$$precision = \frac{TP}{TP+FP} \quad (3)$$

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (4)$$

The true positive (TP) and true negative (TN) represent the correct classification results, while the false positive (FP) and false negative (FN) represent the incorrect classification results. TP is the number of data of a class that is classified correctly. FP

is the number of misclassified data from a class, and FN is the number of data from a class detected as another. TN is the amount of data that does not belong to a class and is not classified as belonging to that class [21, 22].

### 3. Result and Discussion

The COVID-19 classification experiments were carried out using three proposed models: Xception, EfficientNetB3, and parallel classifier. Different data splitting techniques, including 70:30, 80:20, and 90:10 ratios for training and validation data, as well as 5-fold cross-validation, were used in the experiments. The evaluation and analysis focused on the positive COVID-19 class, as the study aimed to detect positive cases of COVID-19 from the non-COVID-19 classes, including normal, bacterial pneumonia, and viral pneumonia.

#### 3.1 Proposed Xception Classifier

The validation results of the COVID-19 classification on the proposed Xception classifier are presented in TABLE II. which details the value of precision, recall, F1-score, and overall accuracy for all classes. Based on TABLE II. , the overall accuracy for all classes with the best results is generated in the split data with a ratio of 90:10, which is 94.44%. Then the precision value, recall, and F1-score for the COVID-19 class resulted in a matter of 100% in the split data ratio of 90:10 and fold 1. As for other data splitting, the COVID-19 class still produced good scores, ranging from 97.81% to 100% for precision, recall, and F1-scores. The evaluation value that does not have a value of 100% is because there are FP and FN in the model validation results.

The model also performed well in identifying non-COVID-19 classes, although not as well as COVID-19. In the 90:10 split data, the normal class had a high precision, recall, and F1-score, achieving a recall score of 100%, a precision of 95.74%, and an F1-score of 97.83%. Similarly, the bacterial pneumonia class achieved a precision value of 96.20%, recall of 84.44%, and F1-score of 89.94%, and the viral pneumonia class achieved a precision of 86.60%, recall of 93.33%, and F1-score of 89.84% in the same split data ratio. Overall, the proposed Xception classifier model effectively identifies COVID-19 cases, and the 90:10 split data ratio is the best for achieving high accuracy and good precision, recall, and F1-score values in all classes.

**Table 2.** The Results of The COVID-19 Classification Evaluation Parameters on The Xception Classifier per Class

Splitting data	Precision (%)				Recall (%)				F1-score (%)				Accuracy (%)	
	C	N	B	V	C	N	B	V	C	N	B	V		
Split data 70:30	100,00	97,00	84,85	80,99	98,15	95,93	82,96	85,19	99,07	96,46	83,90	83,03	90,56	
Split data 80:20	100,00	98,90	91,12	86,84	99,44	100,00	85,56	91,67	99,72	99,45	88,25	89,19	94,31	
Split data 90:10	100,00	95,74	96,20	86,60	100,00	100,00	84,44	93,33	100,00	97,83	89,94	89,84	94,44	
5-fold cross-validation	Fold 1	100,00	98,34	89,27	87,91	100,00	98,89	87,78	88,89	100,00	98,61	88,52	88,40	93,89
	Fold 2	98,90	98,27	84,21	85,23	99,44	94,44	88,89	83,33	99,17	96,32	86,49	84,27	91,53
	Fold 3	97,81	96,09	85,29	80,85	99,44	95,56	80,56	84,44	98,62	95,82	82,86	82,61	90,00
	Fold 4	100,00	98,88	88,62	83,08	99,44	98,33	82,22	90,00	99,72	98,61	85,30	86,40	92,50
	Fold 5	100,00	97,81	86,53	91,52	99,44	99,44	92,78	83,89	99,72	98,62	89,54	87,54	93,89
	Average	99,34	97,88	86,78	85,72	99,55	97,33	86,45	86,11	99,45	97,60	86,54	85,84	92,36

Note: C indicates COVID-19 class, N indicates normal class, B indicates bacterial pneumonia class, and V indicates viral pneumonia class. The blue color indicates the best overall model.

### 3.2 Proposed EfficientNetB3 Classifier

The validation results of the COVID-19 classification on the proposed EfficientNetB3 classifier are presented in TABLE III. , which details the value of precision, recall, F1-score, and overall accuracy or accuracy for all classes. Based on TABLE III. , the accuracy values for all classes with the best results are generated in the split data with a ratio of 90:10, which is 95.28%. Then the precision value, recall, and F1-score for the COVID-19 class resulted in a value of 100% in the split data ratio of 90:10. As for other data splitting, the COVID-19 class still produced good scores, ranging from 97.78% to 100% for precision, recall, and F1 scores. The evaluation value that does not have a value of 100% is because there are FP and FN in the model validation results.

The model also performed well in identifying non-COVID-19 classes, although not as well as COVID-19. In the 90:10 split data, the normal class had a high precision, recall, and F1-score, achieving scores of 100% for all. The bacterial pneumonia class achieved a precision value of 91.95%, recall of 88,89%, and F1-score of 90,40%, and the viral pneumonia class achieved a precision of 89,25%, recall of 92.22%, and F1-score of 90,71% in the same split data ratio. Overall, the proposed EfficientNetB3 classifier model effectively identifies COVID-19 cases, and the 90:10 split data ratio is the best for achieving high accuracy and good precision, recall, and F1-score values in all classes.

**Table 3.** The Result of The COVID-19 Classification Evaluation Parameters on The EfficientNetB3 Classifier per Class

Splitting data		Precision (%)				Recall (%)				F1-score (%)				Accuracy (%)
		C	N	B	V	C	N	B	V	C	N	B	V	
Split data 70:30		99,63	96,24	83,39	84,70	99,26	94,81	85,56	84,07	99,44	95,52	84,46	84,39	90,93
Split data 80:20		98,89	97,27	<b>93,94</b>	87,50	98,89	98,89	86,11	93,33	98,89	98,07	89,86	90,32	94,31
Split data 90:10		<b>100,00</b>	<b>100,00</b>	91,95	89,25	<b>100,00</b>	<b>100,00</b>	88,89	<b>92,22</b>	<b>100,00</b>	<b>100,00</b>	<b>90,40</b>	<b>90,71</b>	<b>95,28</b>
5-fold cross-validation	Fold 1	<b>100,00</b>	97,80	88,17	<b>90,29</b>	98,33	98,89	<b>91,11</b>	87,78	<b>100,00</b>	<b>100,00</b>	<b>90,40</b>	<b>90,71</b>	94,03
	Fold 2	<b>100,00</b>	98,84	85,48	83,24	98,33	94,44	88,33	85,56	99,16	98,34	89,62	89,01	91,67
	Fold 3	99,44	96,24	81,36	82,49	99,44	99,44	80,00	81,11	99,16	96,59	86,89	84,38	90,00
	Fold 4	<b>100,00</b>	96,72	90,64	86,17	98,89	98,33	86,11	90,00	99,44	97,81	80,67	81,79	93,33
	Fold 5	<b>100,00</b>	94,74	86,17	89,16	97,78	<b>100,00</b>	90,00	82,22	99,44	97,52	88,32	88,04	92,50
	Average	99,89	96,87	86,36	86,27	98,55	98,22	87,11	85,33	98,88	97,30	88,04	85,55	92,31

Note: C indicates COVID-19 class, N indicates normal class, B indicates bacterial pneumonia class, and V indicates viral pneumonia class. The blue color indicates the best overall model.

### 3.3 Proposed Parallel Classifier

The validation results of the COVID-19 classification on the proposed parallel classifier are presented in TABLE IV. , which details the value of precision, recall, F1-score, and overall accuracy or accuracy for all classes. Based on TABLE IV. , the accuracy values for all classes with the best results are generated in the split data with a ratio of 90:10, which is 94.44%. Then the precision value, recall, and F1-score for the COVID-19 class resulted in a value of 100% in the split data ratio of 90:10 and fold 4. As for other data splitting, the COVID-19 class still produced good scores, ranging from 98,33% to 100% for precision, recall, and F1 scores. The evaluation value that does not have a value of 100% is because there are FP and FN in the model validation results.

The model also performed well in identifying non-COVID-19 classes, although not as well as COVID-19. In the 90:10 split data, the normal class had a high precision, recall, and F1-score, achieving a recall score of 100%, a precision of 98,90%, and an F1-score of 99,45%. The bacterial pneumonia class achieved a precision value of 90.70%, recall of 86.67%, and F1-score of 88.64%, and the viral pneumonia class achieved a precision of 88.17%, recall of 91.11%, and F1-score of 89.62% in the same split data ratio. Overall, the proposed parallel classifier model effectively identifies COVID-19 cases, and the 90:10 split data ratio is the best for achieving high accuracy and good precision, recall, and F1-score values in all classes.

**Table 4.** Results of COVID-19 Classification Evaluation on Parallel Classifier per Class

Splitting data	Precision (%)				Recall (%)				F1-score (%)				Accuracy (%)	
	C	N	B	V	C	N	B	V	C	N	B	V		
Split data 70:30	99,25	93,06	86,36	79,43	98,52	99,26	77,41	82,96	98,88	96,06	81,64	81,16	89,54	
Split data 80:20	98,90	99,43	89,60	86,84	<b>100,00</b>	96,67	86,11	<b>91,67</b>	99,45	98,03	87,82	89,19	93,61	
Split data 90:10	<b>100,00</b>	98,90	<b>90,70</b>	88,17	<b>100,00</b>	<b>100,00</b>	86,67	91,11	<b>100,00</b>	<b>99,45</b>	88,64	<b>89,62</b>	<b>94,44</b>	
5-fold cross-validation	Fold 1	<b>100,00</b>	<b>99,44</b>	85,57	<b>90,59</b>	99,44	97,78	<b>92,22</b>	85,56	99,72	98,60	<b>88,77</b>	88,00	93,75
	Fold 2	98,89	97,09	83,68	83,71	98,89	92,78	88,33	82,78	98,89	94,89	85,95	83,24	90,69
	Fold 3	98,34	95,21	81,22	84,12	98,89	99,44	81,67	79,44	98,61	97,28	81,44	81,71	89,86
	Fold 4	<b>100,00</b>	94,65	87,43	89,41	<b>100,00</b>	98,33	88,89	84,44	<b>100,00</b>	96,46	88,15	86,86	92,92
	Fold 5	99,44	95,24	84,74	88,34	98,33	<b>100,00</b>	89,44	80,00	98,88	97,56	87,03	83,97	91,94
	Average	99,33	96,33	84,53	87,23	99,11	97,67	88,11	82,44	99,22	96,96	86,27	84,76	91,83

Note: C indicates COVID-19 class, N indicates normal class, B indicates bacterial pneumonia class, and V indicates viral pneumonia class. The blue color indicates the best overall model.

### 3.4 Comparison with Other Classifiers

Comparisons were made to evaluate the performance of the proposed classifier model against state-of-the-art classifiers. It was carried out using the same dataset, number of classes, epochs, batch size, and a data distribution or split ratio of 90:10, as used in the proposed classifier. The results of the classifier comparison are presented in TABLE V. Based on TABLE V. , the proposed classifier outperformed the other four classifiers with an accuracy of 94.44% for the Xception and parallel classifier and 95.28% for the EfficientNetB3 classifier. The proposed classifier's precision, recall, and F1-score values were also higher than all comparison classifiers, with a value of 100% for the COVID-19 class.

In addition to classifying COVID-19 classes effectively, the proposed classifier also performed well in classifying non-COVID-19 classes. The high accuracy results of the proposed classifier, especially EfficientNetB3, were due to parameter settings and layer modifications that outperformed other classifiers. Overall, the proposed classifier model outperformed the comparison classifiers regarding higher accuracy, precision, recall, and F1 scores for the COVID-19 class. The proposed EfficientNetB3 classifier had the highest F1-score for all classes compared to other classifiers and had the best recall and precision values for almost all classes.

### 3.5 Testing in Test Data

Based on TABLE VI. , the proposed classifier models have good accuracy in testing using test data; the Xception classifier has an accuracy of 93.50% for all classes with a



**Table 5.** Evaluation Comparison of The Proposed Classifiers with Other Classifier for COVID-19 Classification

Classifiers	Precision (%)				Recall (%)				F1-score (%)				Accuracy (%)
	C	N	B	V	C	N	B	V	C	N	B	V	
CoroNet [5]	98,88	97,80	81,44	84,34	97,78	98,89	87,78	77,78	98,32	98,34	84,49	80,92	90,56
Parallel (Xception + ResNet50V2) [6]	<b>100,00</b>	96,77	89,29	85,11	98,89	<b>100,00</b>	83,33	88,89	99,44	98,36	86,21	86,96	92,78
VGG16 [23]	97,75	87,88	76,14	78,57	96,67	96,67	74,44	73,33	97,21	92,06	75,28	75,86	85,28
EfficientNetB4 [7]	<b>100,00</b>	96,77	87,78	<b>89,77</b>	98,89	<b>100,00</b>	87,78	87,78	99,44	98,36	87,78	88,76	93,61
Proposed Xception	<b>100,00</b>	95,74	<b>96,20</b>	86,60	<b>100,00</b>	<b>100,00</b>	84,44	<b>93,33</b>	<b>100,00</b>	97,83	89,94	89,84	94,44
Proposed EfficientNetB3	<b>100,00</b>	<b>100,00</b>	91,95	89,25	<b>100,00</b>	<b>100,00</b>	<b>88,89</b>	92,22	<b>100,00</b>	<b>100,00</b>	<b>90,40</b>	<b>90,71</b>	<b>95,28</b>
Proposed Parallel (Xception + EfficientNetB3)	<b>100,00</b>	98,90	90,70	88,17	<b>100,00</b>	<b>100,00</b>	86,67	91,11	<b>100,00</b>	99,45	88,64	89,62	94,44

Note: C indicates COVID-19 class, N indicates normal class, B indicates bacterial pneumonia class, and V indicates viral pneumonia class. The blue color indicates the best overall model.

precision value of 100%, recall of 98.00%, and an F1-score of 98.99 for COVID-19 positive class. The EfficientNetB3 classifier has an accuracy of 95.50% for all classes, with a precision value of 100.00%, a recall of 99.00%, and an F1-score of 99.50% for the COVID-19 positive class. Furthermore, the parallel classifier has accuracy for all classes of 94.25%, with the same precision, recall, and F1-score as the EfficientNetB3 classifier for the COVID-19 positive class.

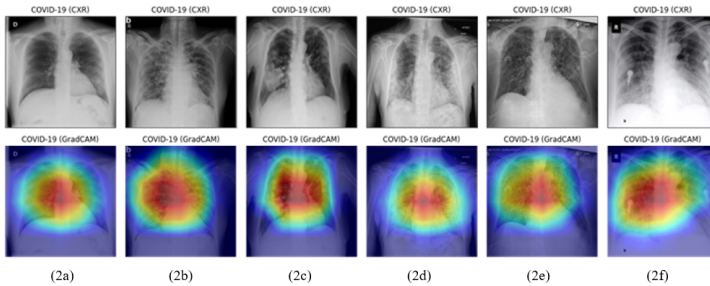
**Table 6.** Evaluation Comparison using Test Data on The Best Model of Each Proposed Classifier per Class

Classifier	Evaluation metrics	C	N	B	V
Xception	precision	<b>100,00</b>	94,34	91,67	88,00
	recall	98,00	<b>100,00</b>	88,00	88,00
	F1-score	98,99	97,09	89,80	88,00
	accuracy	93,50			
EfficientNetB3	precision	<b>100,00</b>	<b>98,04</b>	<b>92,08</b>	<b>91,84</b>
	recall	<b>99,00</b>	<b>100,00</b>	<b>93,00</b>	<b>90,00</b>
	F1-score	<b>99,50</b>	<b>99,01</b>	<b>92,54</b>	<b>90,91</b>
	accuracy	<b>95,50</b>			
Parallel	precision	<b>100,00</b>	98,02	89,32	89,69
	recall	<b>99,00</b>	99,00	92,00	87,00
	F1-score	<b>99,50</b>	98,51	90,64	88,32
	accuracy	94,25			

Note: C indicates COVID-19 class, N indicates normal class, B indicates bacterial pneumonia class, and V indicates viral pneumonia class. The blue color indicates the best overall model.

Tests on unit data are also carried out using the proposed EfficientNetB3 classifier. The test was carried out using four test data for each class. The following evaluation used the Grad-CAM visualization method to provide a visual explanation regarding the model’s decision to detect COVID-19 on CXR images. This is done as an evaluation so the doctor can decide whether to trust the model. This visualization also determines the image’s important features, so the model classifies an image into a particular class. The resulting visualization is a heatmap with different color intensities depending on the importance of features in the image.

Fig 2. shows the results of the Grad-CAM visualization using the proposed Xception classifier in classifying COVID-19 positive classes. The image shows a heatmap from the CXR COVID-19 image, highlighting an important area of the



**Figure 2.** Grad-CAM shows the part of the input image that triggers a positive prediction of COVID-19

image located in the interior of the lung. This means the model can detect the positive class of COVID-19 based on the important features present in the lungs' interior on the CXR image.

#### 4. Conclusion

In summary, this study developed three classifiers, namely Xception, EfficientNetB3, and parallel transfer learning models, for automatic COVID-19 classification from chest X-ray images. These proposed classifiers achieved higher accuracy compared to state-of-the-art models. With a data splitting ratio of 90:10, the Xception classifier achieved an overall accuracy of 94.44%, the EfficientNetB3 classifier achieved 95.28%, and the parallel classifier achieved 94.44%. The COVID-19 class exhibited perfect precision, recall, and F1-score, with a score of 100%. The proposed classifiers can serve as valuable tools to assist radiologists and clinical practitioners in diagnosing and following up on COVID-19 cases.

In future work, this research can be continued to improve accuracy in the classification of COVID-19, building upon the findings of this study. Additionally, further investigations can be conducted to develop methods for classifying the severity level of COVID-19. This expanded approach will enable more comprehensive support in disease management and provide valuable insights for healthcare professionals.

#### References

- [1] Chengdi Wang et al. "COVID-19 in early 2021: current status and looking forward". In: *Signal Transduction and Targeted Therapy* 6.1 (2021), p. 114.
- [2] Luca Falzone et al. "Current and innovative methods for the diagnosis of COVID-19 infection". In: *International journal of molecular medicine* 47.6 (2021), pp. 1–23.
- [3] Mustafa Ghaderzadeh, Farkhondeh Asadi, et al. "Deep learning in the detection and diagnosis of COVID-19 using radiology modalities: a systematic review". In: *Journal of healthcare engineering* 2021 (2021).
- [4] Maria Dietz Toppenberg et al. "Mobile X-ray outside the hospital: a scoping review". In: *BMC Health Services Research* 20.1 (2020), pp. 1–12.
- [5] Asif Iqbal Khan, Junaid Latief Shah, and Mohammad Mudasir Bhat. "CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest x-ray images". In: *Computer methods and programs in biomedicine* 196 (2020), p. 105581.

- [6] Mohammad Rahimzadeh and Abolfazl Attar. “A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet50V2”. In: *Informatics in medicine unlocked* 19 (2020), p. 100360.
- [7] Gonçalo Marques, Deevyankar Agarwal, and Isabel De la Torre Díez. “Automated medical diagnosis of COVID-19 through EfficientNet convolutional neural network”. In: *Applied soft computing* 96 (2020), p. 106691.
- [8] Unais Sait *et al.* “Curated dataset for COVID-19 posterior-anterior chest radiography images (X-Rays)”. In: *Mendeley Data* 1 (2020).
- [9] Laith Alzubaidi *et al.* “Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions”. In: *Journal of big Data* 8 (2021), pp. 1–74.
- [10] Connor Shorten and Taghi M Khoshgofaar. “A survey on image data augmentation for deep learning”. In: *Journal of big data* 6.1 (2019), pp. 1–48.
- [11] Keras. *EarlyStopping*. [https://keras.io/api/callbacks/early\\_stopping/](https://keras.io/api/callbacks/early_stopping/). Accessed: 2023-02-19.
- [12] Diederik P Kingma and Jimmy Ba. “Adam: A method for stochastic optimization. CoRR, vol. abs/1412.6980 (2014)”. In: *arXiv preprint arxiv:1412.6980* (2014).
- [13] Kaggle. *Introduction to Kaggle Kernels*. <https://www.kaggle.com/getting-started/44939>. Accessed: 2021-12-03.
- [14] Jia Deng *et al.* “Imagenet: A large-scale hierarchical image database”. In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.
- [15] Qizhe Xie *et al.* “Self-training with noisy student improves imagenet classification”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 10687–10698.
- [16] François Chollet. “Xception: Deep learning with depthwise separable convolutions”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1251–1258.
- [17] Mingxing Tan and Quoc Le. “Efficientnet: Rethinking model scaling for convolutional neural networks”. In: *International conference on machine learning*. PMLR. 2019, pp. 6105–6114.
- [18] Min Lin, Qiang Chen, and Shuicheng Yan. “Network in network”. In: *arXiv preprint arXiv:1312.4400* (2013).
- [19] Pierre Baldi and Peter J Sadowski. “Understanding dropout”. In: *Advances in neural information processing systems* 26 (2013).
- [20] Ramprasaath R Selvaraju *et al.* “Grad-cam: Visual explanations from deep networks via gradient-based localization”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 618–626.
- [21] Jasmina Dj Novaković *et al.* “Evaluation of classification models in machine learning”. In: *Theory and Applications of Mathematics & Computer Science* 7.1 (2017), p. 39.
- [22] M. Kamber J. Han and J. Pei. *Data Mining*. Third Edit. Elsevier, 2012.