

Automated medical diagnosis of COVID-19 through EfficientNet convolutional neural network

Gonçalo Marques*, Deevyankar Agarwal, Isabel de la Torre Díez

Department of Signal Theory and Communications, and Telematics Engineering University of Valladolid, Paseo de Belén, 15, 47011 Valladolid, Spain



ARTICLE INFO

Article history:

Received 17 August 2020
Received in revised form 22 August 2020
Accepted 26 August 2020
Available online 29 August 2020

Keywords:

Automated decision support system
Convolutional Neural Network (CNN)
COVID-19
Deep learning
Machine learning

ABSTRACT

COVID-19 infection was reported in December 2019 at Wuhan, China. This virus critically affects several countries such as the USA, Brazil, India and Italy. Numerous research units are working at their higher level of effort to develop novel methods to prevent and control this pandemic scenario. The main objective of this paper is to propose a medical decision support system using the implementation of a convolutional neural network (CNN). This CNN has been developed using EfficientNet architecture. To the best of the authors' knowledge, there is no similar study that proposes an automated method for COVID-19 diagnosis using EfficientNet. Therefore, the main contribution is to present the results of a CNN developed using EfficientNet and 10-fold stratified cross-validation. This paper presents two main experiments. First, the binary classification results using images from COVID-19 patients and normal patients are shown. Second, the multi-class results using images from COVID-19, pneumonia and normal patients are discussed. The results show average accuracy values for binary and multi-class of 99.62% and 96.70%, respectively. On the one hand, the proposed CNN model using EfficientNet presents an average recall value of 99.63% and 96.69% concerning binary and multi-class, respectively. On the other hand, 99.64% is the average precision value reported by binary classification, and 97.54% is presented in multi-class. Finally, the average F1-score for multi-class is 97.11%, and 99.62% is presented for binary classification. In conclusion, the proposed architecture can provide an automated medical diagnostics system to support healthcare specialists for enhanced decision making during this pandemic scenario.

© 2020 Elsevier B.V. All rights reserved.

1. Introduction

Currently, COVID-19 has led to critical consequences on the economic and social structures of developed and developing countries [1,2]. The first case of infection by coronavirus has been reported in December 2019 at Wuhan, China [3]. However, the current pandemic situation still to be resolved [4,5]. This virus critically affects several countries such as the USA [6], Brazil [7], India [8] and Italy [9]. Therefore, several research units are working to develop policies, vaccines and novel methods to control this pandemic scenario [10].

On the one hand, numerous researchers from the medicine domain are developing drugs to stop the virus prefoliation [11, 12]. First, it is necessary to develop new methods to help infected people [13–15]. Second, it is also crucial to plan sanitary policies to prevent infected patients from disseminating the virus [16,17]. On the other hand, computer science researchers have a critical

role in the development of new methods to support virus diagnostics [18,19]. Several innovations have been developed using a set of numerous technologies, such as the development of mobile applications to monitor and track the interaction between people [20,21].

Artificial intelligence (AI) is also crucial in this field and to develop solutions to support diagnosis [22]. AI methods have been used to create automated systems for COVID-19 diagnosis [23–25]. These methods will never replace human care. However, they can be a relevant solution to combat the virus. AI is widely used in the medicine domain [26]. Despite the ethical concern regarding the application of AI with patients' data in the current pandemic scenario, these methods should be used to support medical staff [27–29]. The stress factors that affect medical professionals during this pandemic scenario concerning the increase of patients in the hospitals are significantly affect their work and performance [30]. Consequently, it is necessary to create novel methods that can support their work.

Nowadays, researchers are using CNN (Convolutional Neural Network), a class of deep learning neural networks for several applications [31–33]. CNNs have an input layer, an output layer, and hidden layers [34]. The hidden layers usually consist of convolutional layers, ReLU layers, pooling layers, and fully connected

* Corresponding author.

E-mail addresses: goncalosantosmarques@gmail.com (G. Marques), deevyankar.agarwal@hct.edu.om (D. Agarwal), isator@tel.uva.es (I. de la Torre Díez).

layers [35,36]. CNN's represents a huge breakthrough in automatic image classification systems as there is no need for pre-processing the images, that was needed in traditional machine learning algorithms [37–39].

The main objective of this research is to propose a medical decision support system using CNNs. To the best of the authors' knowledge, there is no similar study that proposes an automated method to the detected COVID in CT X-ray images using EfficientNet [40,41]. Therefore, the main contribution of this paper is to present an automated medical diagnosis system implemented using EfficientNet. Numerous studies and applications have been reported in the literature. The authors believe that in the critical pandemic scenario is crucial to share all the methods and materials to allow the readers to reproduce the results. The authors share the python scripts developed in the Google Colab platform. In this way, the software is accessible to all the readers who can execute the scripts for future research activities. This architecture can be used for transfer learning, and it is more efficient than most of its predecessors such as VGG (e.g. VGG16 or VGG19), GoogLeNet (e.g. InceptionV3), and Residual Network (e.g. ResNet50) [42]. The EfficientNet model consists of 8 models from B0 to B7, with each subsequent model number referring to variants with more parameters and higher accuracy. EfficientNet architecture uses transfer learning to save time and computational power. Consequently, it provides higher accuracy values than the competitor known models. This is due to the use of a clever scaling at depth, width, and resolution. The authors have used the B4 model, as it contains 19 m parameters, that is feasible for our experimental setup, as B5, B6 and B7 include 30M, 43M and 66M params, respectively [41]. Furthermore, the authors have used separated datasets to validate the proposed CNN models using images that are not included during the testing and training phase. The proposed model has been evaluated using stratified cross-validation 10-fold stratified. This paper presents two main experiments including different datasets for testing and validation to ensure the non-occurrence of overfitting. First, the binary classification results using images from COVID-19 patients and normal patients are shown. Second, the multi-class results using images from COVID-19, pneumonia and normal patients are discussed. The source code is provided in this document as supplementary files.

The remainder of this paper is structured as follows. Section 2 introduces the related work. The materials and methods used in this research are described in Section 3. Section 4 presents the results of the proposed CNN model. The discussion and comparison of the proposed method with the related work available in state of the art are presented in Section 5. Finally, the conclusions are presented in Section 6.

2. Related work

Numerous researchers are working at their best effort using AI technologies to develop novel systems to support COVID-19 diagnosis. These studies aim to create new automated systems for COVID-19 diagnosis. These methods should be used to support medical staff in the current pandemic scenario. Furthermore, machine learning technologies can be used to decrease the stress factors that affect medical professionals during this pandemic scenario concerning the increase of workflow in healthcare facilities.

Ozturk et al. [43] propose an automated detection system for COVID-19 cases using deep neural networks and chest X-ray images. The proposed method is based on the DarkNet model for real-time detection and implements 17 convolutional layers. This study aims to support the decision making of radiologists to validate their screening process. The heatmaps produced by

this automated system have been evaluated by radiologists. The dataset used includes 1125 images. In total, 125 samples are used for COVID-19 class, 500 for pneumonia class and for 500 normal class. The authors have used 5-fold cross-validation to validate the performance of the proposed method. The average accuracy of 98.08% and 87.02% is reported for binary and multi-class classification. The limitation on the number of samples used for COVID-19 class is reported by the authors.

A deep learning model to improve the accuracy of binary classification of COVID-19 is proposed in [44]. The proposed CNN was implemented based on the VGG-19 classifier. The dataset used includes a total of 364 X-ray scans. The model performance has been validated using random sampling. The ratio used for train, validation and testing was 80:20:20. The number of samples for normal class and COVID-19 was 233 and 115 during training, 56 and 32 during validation, and 75 and 34 during testing. The results show an accuracy of 96.3%. The limited number of samples for COVID-19 cases is stated by the authors.

Apostolopoulos et al. [45] propose a transfer learning approach using VGG-19 and MobileNet v2 for automated detection of patients with pneumonia and COVID-19. Two different datasets have been included in this study. One dataset of a total of 1427 samples that include 504, 700 and 224 images of normal, pneumonia and COVID-19, respectively, have been used. On the other hand, a different dataset of 224 samples for COVID-19, 714 samples of pneumonia patients and 504 of normal patients is also included. The 10-fold cross-validation has been used to evaluate the proposed models. The VGG-19 and MobileNet v2 reported 98.75% and 97.40% of accuracy for binary classification and 93.48% and 92.85% for multi-class concerning the first dataset. Furthermore, the MobileNet v2 has been applied in the second dataset presenting an accuracy of 96.78% for binary classification and 94.72% for multiclass. The authors state that a more in-depth analysis using more patient data concerning COVID-19 individuals is required.

The authors of [46] proposed a fast screen system for COVID-19 detection based on deep learning neural networks. The presented method is based on the nCOVnet and uses chest X-rays images. The dataset included in this study has a total of 337 samples. In total, 192 of the samples are from COVID-19 positive patients and 142 images of normal patients. The model's performance has been evaluated using random sampling using 70% for training and 30% for testing. The proposed system for binary classification provides an accuracy of 88.10%. The authors of this study state the limitations on the number of samples included in the used dataset.

A novel artificial neural network system for COVID-19 detection is proposed in [47]. The proposed method is based on Convolutional CapsNet and uses chest X-ray images. The system includes binary and multi-class classification features. The dataset used includes a total of 3150 samples, 1050 images each class (normal, pneumonia and COVID-19). The 10-fold cross-validation is used to evaluate the performance of the proposed system. The results show an accuracy of 97.24% and 84.22% for binary and multi-class classification. The limitations reported by the authors focus on the hardware resources needed to process a massive number of images and the processing time.

Nour et al. [48] propose a novel medical diagnosis model of COVID-19 to support clinical applications. The system is based on deep features and Bayesian optimization. The CNN model is applied for automated extraction of features that are often processed by different machine learning methods such as kNN, SVM and Decision Tree. The used dataset contains 2033 of samples, 135 for COVID-19, 939 for normal class and 941 for pneumonia. The authors have used data augmentation to increase the number of samples concerning COVID-19 class. The performance of the

Table 1
Related work on COVID-19 detection systems.

Reference	Model	Data used	Number of images	Classification	Evaluation method
[43]	DarkNet model	Chest X-ray	1125 – total; 125 – COVID-19; 500 – Pneumonia; 500 – Normal	Binary and Multi-class	5-fold cross-validation
[44]	VGG-19	Chest X-ray	545 – total; 181 – COVID-19; 364 - Normal	Binary	Random sampling 80:20:20 for train, validation and testing.
[45]	VGG-19 and MobileNet v2	Chest X-ray	Dataset 1: 1427 – total; 504 – Normal; 700 – Pneumonia; 224 – COVID-19 Dataset 2: 1442 – total; 504 – Normal; 714 – Pneumonia; 224 – COVID-19	Binary and Multi-class	10-fold cross-validation
[46]	nCOVnet	Chest X-ray	337 – total; 192 – COVID-19; 142- Normal	Binary	Random sampling 70% for training and 30% for testing
[47]	CapsNet	Chest X-ray	3150 – total; 1050 – Normal; 1050 – Pneumonia; 1050 – COVID-19	Binary and Multi-class	10-fold cross-validation
[48]	Proposed CNN	Chest X-ray	2033 – total; 135 – COVID-19, ; 939 – Normal; 941 – Pneumonia.	Binary and Multi-class	Random sampling 70% for training and 30% for testing
[49]	ResNet 18	Chest X-ray	746 – total 349 – COVID-19 397 – Normal.	Binary	Random sampling. 70% for training and 30% for testing.
[50]	Proposed Semi-supervised model	Chest X-ray	Dataset 1: 2482 – total; 1230 – Normal; 1252 – COVID-19 Dataset 2: 20 – COVID-19	Binary	Random sampling 70% for training and 30% for testing

proposed system has been evaluated using 70% and 30% of the dataset for training and testing, respectively. The proposed CNN presents an accuracy of 97.14%.

In [49], the authors have used augmentation for increasing the size of training dataset by using stationary wavelets and compared different transfer learning CNN architectures. The dataset used includes 349 samples for COVID-19 and 397 samples for normal class. The authors also applied data augmentation techniques to increase the number of samples for both classes. In this study, 70% of the samples are used for training, and 30% have been considered for validation. The proposed method provides 99.4% accuracy during testing for a binary classifier using the ResNet 18 model.

Konar et al. [50] propose a semi-supervised shallow neural network model for automated diagnostic of COVID-19. This study included two datasets. One of them consists of a total of 2482 samples, from these 1252 samples are from COVID-19 positive patients, and 1230 are from not infected patients. The second dataset includes 20 samples of COVID-19 positive patients. The proposed model has tested using random sampling with a ratio of 70% for training and 30% for testing. Moreover, the model has also been evaluated using 5 and 10-fold validation. The proposed method presents an accuracy of 93.1%.

In summary, several methods have been proposed in the literature for the automated diagnostic of COVID-19. These studies use different number of images and datasets from multiple sources. Moreover, different approaches have been used to evaluate the performance of the models such as cross-validation and random sampling. Most of the studies state the limitation associated with the number of samples to conduct the experiments. Table 1 summarizes the related work on COVID-19 detection systems.

3. Methods and materials

This section presents the methods and materials used in this study. Section 3.1 details the datasets of X-ray images used to test and train the proposed method. The proposed CNN is presented in Section 3.2. Finally, the validation method and experimental setup are presented in Section 3.3.

3.1. X-ray Image DataSet

The samples used to train and test the proposed method have been collected from public datasets. It is critical to ensure

the equal number of samples that cover the analyzed classes to properly validate the performance of the model. Consequently, the authors have used the same number of images to train and test. Table 2 presents the reference and number of images used by the authors. In total, 404 samples have been used corresponding to Normal, Pneumonia and COVID-19. These samples have been used in the stratified 10-fold cross-validation.

Furthermore, the authors have tested the model using a separate dataset for validation. The number of samples used to validate the proposed model was 96 for normal class and 100 for pneumonia and COVID-19 class. These datasets were not used in the training phase of the model, and this experiment has been conducted to test the non-occurrence of overfitting.

On the one hand, pneumonia and normal images have been retrieved from the dataset available in [51]. This dataset is public and contains validated chest X-ray images of the pneumonia patients and normal patients. It is freely available on the Kaggle website. This dataset only contains the folders named as Pneumonia and Normal, and there is no other information available. The authors have downloaded these folders directly from the Kaggle website to be included in the proposed work. On the other hand, the COVID-19 Image DataSet has been used to retrieve the COVID-19 positive samples and is available in [52]. It is a public dataset of validated chest X-ray images of COVID-19 positive patients. It is available on GitHub database repository. The images are collected from public sources as well as through indirect collection from hospitals and physicians. This project is approved by the University of Montreal's Ethics Committee (CERSES-20-058-D). The Research community is adding images continuously in this dataset. The main purpose of developing this dataset is to improve prognostic predictions to triage and manage patient care. The authors used only two attributes, the findings column to identify COVID-19 images and the name of the image.

3.2. Proposed CNN

The authors have used the EfficientNetB4 model for the transfer learning process and added a global_average_pooling2d layer to minimize overfitting by reducing the total number of parameters. In addition to this, a sequence of 3 inner dense layers with RELU activation functions and dropout layers have been added. In total, a 30% dropout rate has chosen randomly to avoid overfitting. Finally, one output dense layer contains two output units in case of binary classification, and 3 output units for multi-class classification, with softmax activation function that has been

Table 2
Dataset information.

Class	Reference	Number of images for training/ testing	Number of images for validation
NORMAL	[51]	404	96
PNEUMONIA		404	100
COVID-19	[52]	404	100

Table 3
Layer types and parameters used in the proposed model.

Layer (type)	Output shape	Param #
EfficientNetB4 (Model)	7 × 7 × 1792	17,673,816
global_average_pooling2d	1792	0
dense (Dense)	128	229,504
dropout (Dropout)	128	0
dense_1 (Dense)	64	8256
dropout_1 (Dropout)	64	0
dense_2 (Dense)	32	2080
dropout_2 (Dropout)	32	0
dense_3 (Dense)	2/3	99
Total Parameters: 17,913,755		
Trainable Parameters: 17,788,555		
Non-trainable Parameters: 125,200		

added to create the proposed automated detection system. The details of the layers and their order in the proposed model, output shape of each layer, the number of parameters (weights) in each layer, and the total number of parameters (weights) are presented in Table 3. The total number of parameters is 17,913,755.

All the software and libraries used in the proposed work are open source. To reproduce the results, the readers should use Google Colab Notebook using the GPU run time type. This software can be used without costs since it is provided by Google for research activities using a Tesla K80 GPU of 12 GB. The EfficientNet Models are pre-trained, scaled CNN models that can be used for transfer learning in image classification problems. The model is developed by Google AI in May 2019 and is available from Github repositories. The ImageDataAugmentor is a custom image data generator for Keras which supports augmentation modules. It is also developed by Google AI and is available from Github repositories. Finally, the Albumentations library is also developed by Google AI and can be installed from Github Repositories. In summary, the software used can be used without license concerns as it is free and open source.

The authors have used three different main libraries in the proposed method. These libraries include the EfficientNet module, the Albumentation module and the ImageDataAugmentor module.

On the one hand, the EfficientNet Models are based on simple and highly effective compound scaling methods. This method enables to scale up a baseline ConvNet to any target resource constraints while maintaining model efficiency, used for transfer learning datasets. In general, EfficientNet models achieve both higher accuracy and better efficiency over existing CNNs such as AlexNet, ImageNet, GoogleNet, and MobileNetV2 [41]. EfficientNet could serve as a new foundation for future computer vision tasks. There is no similar study that uses EfficientNet for transfer learning concerning COVID-19 classification to the best of authors knowledge until this date. EfficientNet includes models from B0 to B7, and each one has different parameters from 5.3M to 66M. The authors used EfficientNetB4 that contains 19M parameters, as it is suitable according to our resources and purpose.

On the other hand, the Albumentation library is widely used in industry, deep learning research, machine learning competitions, and open source projects. This module efficiently implements a variety of image transform operations that are optimized for performance. This library provides an image augmentation interface for different computer vision tasks, including object classification,

segmentation, and detection. The authors have used the Compose method of the Albumentation library. This library reduces overfitting, improve the performance of classifiers and the decrease execution time [53]. After the implementation of this library for augmentation purposes in each fold, the model accuracy of the model has increased, and the processing time decreased.

Finally, the ImageDataAugmentor is a custom image data generator for Keras supporting the use of modern augmentation modules (e.g. imgaug and albumentations) [54]. This library is used to configure the Image data generator according to the albumentations settings to decrease execution time. Data generator is created by using the constructor of ImageDataAugmentor class with two arguments. One is *rescale* whose value is set as 1/255 to transform every pixel value from range [0, 255] to [0, 1]. Another is the *augment* value that is configured as the output of the compose function of the Albumentation library. Data generator has used further to process the image datasets. Fig. 1 presents the block diagram of the proposed work.

3.3. Validation and experimental setup

The model has been validated in two different phases. On the one hand, the 10-fold cross-validation method has been using the same dataset for training and testing. On the other hand, a separate dataset which contains samples that have not been used during the training phase has been applied to validate the performance of the model. The confusion matrix has been extracted. Consequently, the precision, recall and F1-score have been computed concerning the separated classes. Finally, the average values for each fold have been calculated. The experimental setup used to conduct this study is detailed in Algorithm 1.

4. Results

The experiments were carried out on Google Colab notebook using GPU run time type. The training of the proposed CNN model was realized using stratified 10-fold cross-validation method. In total, 11 epochs were used in each fold. Moreover, 69 steps for multi-class and 46 steps for binary classification are used in each epoch. The mini-batch size used was 16.

The training of the model was completed in a total of 7590 iterations for multi-class and 5060 iterations for binary class. The time elapsed for the training of the model was 111.83 min for multi-class CNN and 79.16 min for binary class. The initial learning rate was 0.0001. The authors employed a ReduceLROn-Plateau method since it reduces the learning rate when it stops improving. This callback monitors the improvement, and if no improvement is verified for a 'patience' number of epochs, the learning rate is reduced. The authors have defined patience=3 and the min_lr=0.000001 in the proposed method. The ADAM optimization method was used as a solver. The training and validation graphs with the loss, confusion matrix, and area under the curve of receiver operating characteristics for each fold of the proposed CNN can be verified from the supplementary files. After training of all the 10-fold CNN models, the best model is identified and used for the validation testing, by using different datasets. The performance reported in the validation experiment is promising. The scripts and detailed information concerning the experiments can be consulted in the supplementary files.

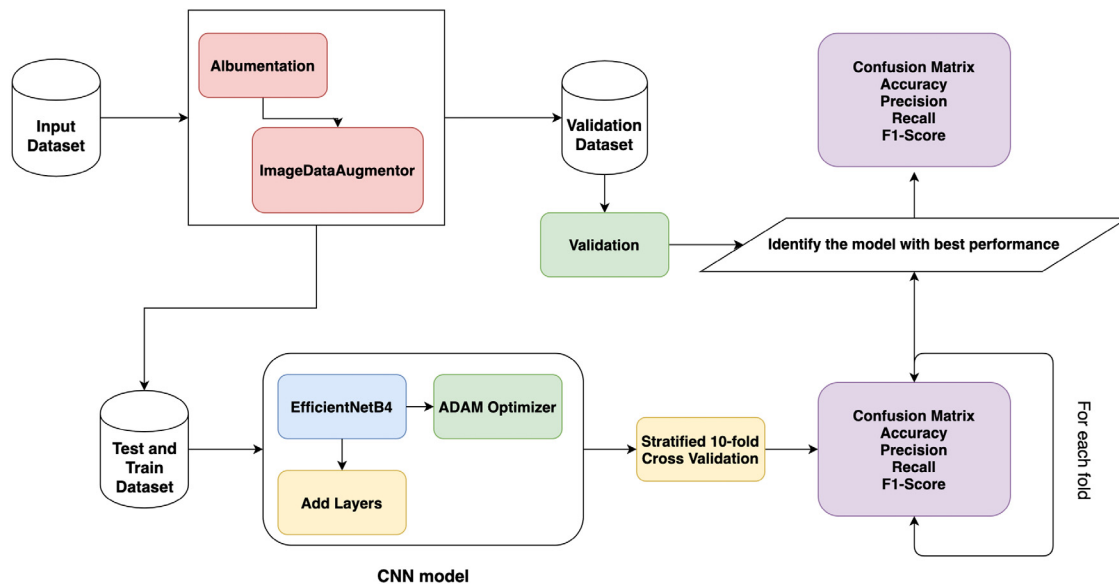


Fig. 1. Block Diagram of the proposed work.

Algorithm 1: Experimental Setup

Input	1. Download Images of three categories {COVID-19, NORMAL, PNEUMONIA}.
Environment	2. Use Google CoLab and install the required libraries
Configuration	3. Import the images
	4. Configure training, testing, and validate the model using stratified 10-fold cross-validation
Data Generator Configuration	5. Define an augmentation pipeline using the Compose function of albumentation library
	6. Create the data generator as an object of the ImageDataAugmentor library and configure the augmentation pipeline obtained in (5).
	7. Configure batch_size = 16.
Directories Configuration	8. Create two directories {DIR-1, DIR-2} of the images with their labels according the classes. DIR-1 will be used for training/testing and DIR-2 will be used for the validation.
Training and Testing Apply 10-fold stratified cross-validation.	9. Create the model using EfficientNetB4 and dense layers with relu activation function and an output layer with a softmax activation function
	10. Compile the model using the ADAM optimizer with the learning rate as .0001, Categorical_Crossentropy function for loss calculation.
	11. Model fitting using 11 epochs and ReduceLRonPlateau function to reduce the learning rate when the metrics stop improving.
	12. Save the model to be used for validation testing.
	13. Configure testing dataset.
	14. Generate performance score values for each fold. <ol style="list-style-type: none"> a. Classification Report b. Confusion Matrix c. AUC-ROC curve d. Model loss graph e. Model Accuracy graph
	<i>Note: Following steps are executed 10 times.</i>
Validation	15. Identify the best model from the outputs of training and testing phase
	16. Load the identified model
	17. Load validation datasets
	18. Perform the validation
	19. Generate performance score values <ol style="list-style-type: none"> a. Confusion Matrix b. AUC-ROC curve

Table 4
Results of binary classification for COVID-19 class.

Fold	Precision	Recall	F1 score
1	100%	100%	100%
2	97.61%	100%	98.79%
3	100%	97.56%	98.76%
4	100%	100%	100%
5	100%	100%	100%
6	100%	100%	100%
7	100%	100%	100%
8	97.56%	100%	98.76%
9	100%	100%	100%
10	100%	100%	100%
Average	99.51%	99.75%	99.63%

Table 5
Results of binary classification for normal class.

Fold	Precision	Recall	F1 score
1	100%	100%	100%
2	97.61%	100%	98.79%
3	97.56%	100%	98.76%
4	100%	100%	100%
5	100%	100%	100%
6	100%	100%	100%
7	100%	100%	100%
8	100%	97.56%	98.76%
9	100%	100%	100%
10	100%	100%	100%
Average	99.51%	99.75%	99.63%

It is crucial to provide all the details about the methods and materials used to allow the readers to reproduce the results. Therefore, the python scripts developed in a Google Colab platform are included as supplementary files. The supplementary file A presents the results for binary classification, and the supplementary file B contains the results for multi-class classification.

4.1. Experimental results of binary classification

In this section, the results for the binary classification are presented. In total, 728 images have been used for training, and 80 images have been used for testing. The results are presented for each fold, and the average value is also reported. The accuracy, precision, recall and f1-score results are presented for each class and for both classes.

Table 4 presents the results for the binary classification considering the COVID-19 class. The lower performance values of the model concerning the COVID-19 class samples are reported for the 2, 3 and 8-fold. The lower precision values of 97.61% and 97.56% occurred in the 2-fold and 3-fold, respectively. Moreover, the minimum recall value is 97.56% in 3-fold. The F1-score is 98.79% at 2-fold and 98.79% at 3-fold and 8-fold. The average precision, recall, and F1-score are 99.51%, 99.75% and 99.63%, respectively.

The performance result for binary classification concerning the normal class is presented in Table 5. The lowest recall value is 97.56% reported for the 3-fold. Furthermore, the precision is 97.61% and 97.56% for 2-fold and 3-fold, respectively. Finally, the F1-score is 98.79% at 2-fold and 98.76% for 3-fold and 8-fold.

Table 6 presents the accuracy, precision, recall and F1-score results for binary classification for both classes. The average accuracy reported is 99.62%. Moreover, 99.64%, 99.63% and 99.62% are reported concerning precision, recall, and F1-score, respectively.

4.2. Experimental validation results of classification

An external dataset which contains samples that have not been used during the training phase has been applied to validate

Table 6
Results of binary classification between classes.

Fold	Accuracy	Precision	Recall	F-1 Score
1	100%	100%	100%	100%
2	98.76%	98.88%	98.75%	98.76%
3	98.76%	98.78%	98.78%	98.76%
4	100%	100%	100%	100%
5	100%	100%	100%	100%
6	100%	100%	100%	100%
7	100%	100%	100%	100%
8	98.76%	98.78%	98.78%	98.76%
9	100%	100%	100%	100%
10	100%	100%	100%	100%
Average	99.62%	99.64%	99.63%	99.62%

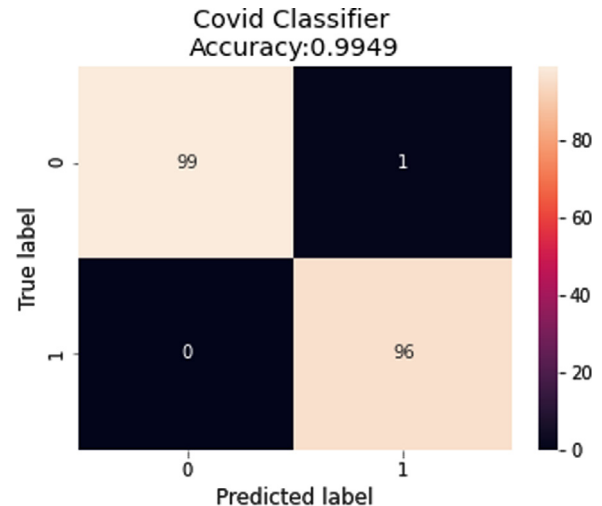


Fig. 2. Confusion matrix of the validation testing for the binary classifier.

the performance of the model and ensure that the model is not overfitted. This external dataset contains 96 samples for normal class and 100 samples for COVID-19 class. To the best of the author's knowledge, there is no similar study that has used this method to validate the proposed CNN model. The trained model of 10-fold has been used for validation. The confusion matrix for the cross-validation test is presented in Fig. 2.

The precision value is 100% for COVID-19 class and 98.96% for normal class. Moreover, the reported recall value is 99% for COVID-19 class and 100% for normal class. Finally, the F1-score is 99.49% for COVID-19 class and 99.48% for normal class. The average accuracy is 99.49%. Therefore, this experiment ensures the efficiency of the proposed method. The receiver operating characteristic for binary-class data is presented in Fig. 3.

4.3. Experimental results of multi-class classification

The classification results for multi-class are presented in this section. In total, 1092 samples have been used for training, and 122 images have been used for testing. Three different classes are analyzed, such as COVID-19, normal and pneumonia. Table 7 presents the classification results concerning the COVID-19 class. On the one hand, the lower precision value of 97.56% has been reported for 8, 9 and 10-fold. On the other hand, the lower F1-score of 98.76% is reported for 8, 9 and 10-fold. The average precision recall and F1-score values are 99.26% 100% and 99.62%, respectively.

The results for normal class are presented in Table 8. The maximum precision value is 100% reported at the 6-fold, and the minimum precision value is 92.50% in 10-fold. Moreover,

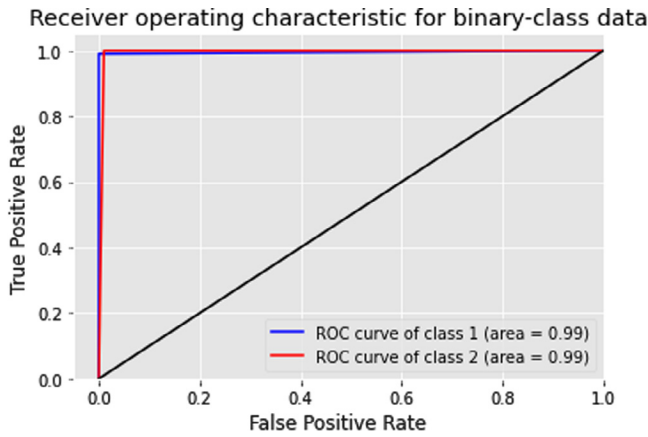


Fig. 3. ROC curve of the validation testing for the binary classifier.

Table 7
Results of multi-class classification for COVID-19 class.

Fold	Precision	Recall	F1 Score
1	100%	100%	100%
2	100%	100%	100%
3	100%	100%	100%
4	100%	100%	100%
5	100%	100%	100%
6	100%	100%	100%
7	100%	100%	100%
8	97.56%	100%	98.76%
9	97.56%	100%	98.76%
10	97.56%	100%	98.76%
Average	99.26%	100%	99.62%

Table 8
Results of multi-class classification for normal class.

Fold	Precision	Recall	F1 Score
1	92.68%	95.00%	93.82%
2	95.00%	95.00%	95.00%
3	97.50%	97.50%	97.50%
4	95.00%	95.00%	95.00%
5	97.50%	97.50%	97.50%
6	100%	97.56%	98.76%
7	95.34%	100%	97.61%
8	97.61%	100%	98.79%
9	97.50%	95.12%	96.29%
10	92.50%	92.50%	92.50%
Average	96.06%	96.51%	96.27%

the lower recall result is 92.50% at 10-fold. Finally, the F1-score ranges from 92.50% at 10-fold to 98.79% at 8-fold. The average results of 96.06%, 96.51% and 96.27% are presented for precision, recall and F1-score, respectively.

The results for pneumonia class are presented in Table 9. These results show that precision ranges from 95% at the 1-fold to 100% at 6,7, and 8-fold. The lowest value for recall is 87.80% at 10-fold. Finally, the better F1-score value is reported at 6-fold, and the lower value is 92.30% presented in 10-fold. The average values reported for precision, recall and f1-score concerning pneumonia samples are 97.47%, 93.58% and 95.45%, respectively.

The average score values between classes such as COVID-19, normal and pneumonia are presented in Table 10. The average accuracy, precision, recall and F1-score are 96.70%, 97.59%, 96.69% and 97.11%, respectively. On the one hand, the highest average precision is reported at 6-fold, and the lowest value is 95.78% presented at 10-fold. On the other hand, the lower result concerning recall is 93.43% presented at 10-fold, and the higher value is reported at 99.18% at 6-fold. Furthermore, the average F1-score ranges from 94.52% at 10-fold to 99.58% at fold-6-fold. The results

Table 9
Results of multi-class classification for pneumonia class.

Fold	Precision	Recall	F1 Score
1	95.00%	92.68%	93.82%
2	95.12%	95.12%	95.12%
3	97.5%	95.12%	96.29%
4	95.12%	95.12%	95.12%
5	97.43%	95.00%	96.20%
6	100%	100%	100%
7	100%	92.50%	96.10%
8	100%	92.50%	96.10%
9	97.29%	90.00%	93.50%
10	97.29%	87.80%	92.30%
Average	97.47%	93.58%	95.45%

Table 10
Results of multi-class classification between all classes.

Fold	Accuracy	Precision	Recall	F-1 Score
1	95.90%	95.89%	95.89%	95.88%
2	96.72%	96.70%	96.70%	96.70%
3	97.54%	98.33%	97.54%	97.93%
4	96.72%	96.70%	96.70%	96.70%
5	97.52%	98.31%	97.50%	97.90%
6	99.17%	100%	99.18%	99.58%
7	97.52%	98.44%	97.50%	97.90%
8	97.52%	98.39%	97.50%	97.88%
9	95.04%	97.45%	95.04%	96.18%
10	93.38%	95.78%	93.43%	94.52%
Average	96.70%	97.59%	96.69%	97.11%

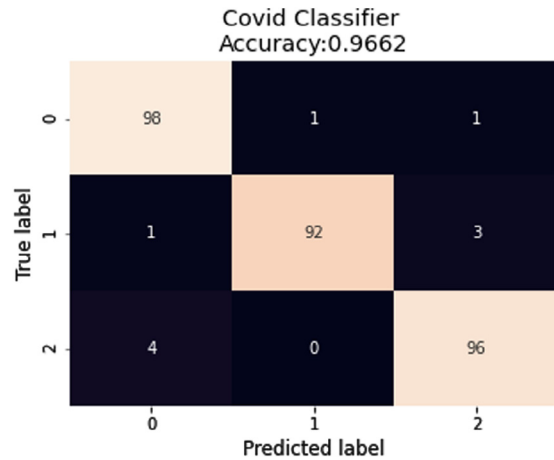


Fig. 4. Confusion matrix of the validation testing for the multi-class classifier.

presented in Table 9 ensure the performance of the proposed method for automated medical diagnosis.

4.4. Experimental validation results of multi-class classification

Similarly to the experiment presented in Section 4.2, an external dataset has been applied to validate the performance of the model. This external dataset contains 96 samples for normal class, 100 samples for COVID-19 class and 100 samples for pneumonia class. The trained model of 10-fold has been used for validation. The extracted confusion matrix for the cross-validation test for multi-class classification is presented in Fig. 4.

The average reported accuracy is 96.62%. The results for precision, recall and F1-score for COVID-19 class is 98.00%. Moreover, 98.92%, 95.83% and 97.35% are reported for precision, recall and F1-score, respectively, concerning samples for normal class. Finally, the precision, recall and F1-score for pneumonia class reported are 96.00%. The receiver operating characteristic for multi-class data is presented in Fig. 5.

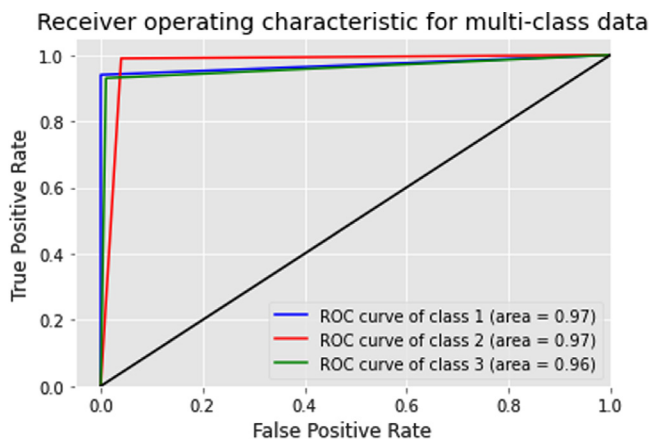


Fig. 5. ROC curve of the validation testing for the multi-class classifier.

In summary, the results of the proposed CNN model for automated medical diagnostics support are promising. The reported average accuracy values for binary and multi-class are 99.62% and 96.70%, respectively. On the one hand, the proposed CNN model using EfficientNet architecture presents an average recall value of 99.63% and 96.69% concerning binary and multi-class, respectively. On the other hand, the average precision of 99.64% is reported by binary classification, and 97.54% is presented in multi-class. Finally, the average F1-score value for multi-class is 97.11%, and 99.62% is presented for binary classification.

5. Discussion

The proposed model is compared with the related work. Indeed, it is crucial to mention this comparison is limited concerning the differences in the samples used and the parameters of the machine learning methods. Furthermore, since most of the studies do not provide the software, it is not possible to compare the methods applied by the studies available in the literature using the samples included in our research or vice-versa.

The number of new methods proposed in the literature by computer science researchers increases every day. Currently, the focus on these systems to support health professionals is a trending topic. Different architectures such as DarkCovidNET, VGG19, MobileNet v2, CapsNet and nCOVnet have been proposed for automated medical diagnosis of COVID-19. This section aims to compare the proposed model results with similar studies available in state of the art. Table 11 presents the results presented for binary classification studies available in the literature.

The proposed method outperforms all the works concerning binary classification. Nevertheless, the authors of [45] proposed an architecture with a relevant accuracy of 98.75%. Moreover, the method proposed in [43] also presents a significant accuracy of 98.08%. Consequently, the implementation of the EfficientNet architecture presents promising results for automated medical diagnosis of COVID-19 concerning binary classification. Different methods for multi-class classification of COVID-19 patients are presented in Table 12.

The accuracy levels proposed by the methods in Table 11 range from 84.22% [47] to 97.14% [48]. When compared with the system proposed by the authors of [48] our method provides less accuracy but higher recall and F1-score. The authors of [48] present a recall of 94.61% and an F1-score of 95.75%. The proposed method provides 96.69% and 97.11% concerning recall and F1-score, respectively.

The authors have used stratified 10-fold cross-validation to evaluate the proposed models. On the one hand, Table 3 contains

the precision, recall, and F1 score of each fold for the binary classification considering the COVID-19 class. From the analysis of Table 3 the authors identified the best model, and consequently choose the model trained in the 10-fold for validation using samples that have not been including in the training process. On the other hand, Table 6 presents the precision, recall, and F1 score of each fold for the multi-class classification considering the COVID-19 class. Based on the results of Table 6 the authors selected the model trained in the 6-fold for validation using samples that have not been included in the training process. The output of validation testing is detailed in Sections 4.2 and 4.4 for binary and multi-class, respectively. Fig. 2 presents the confusion matrix and Fig. 3 shows the AUC-ROC curve concerning binary classification. The results presented an accuracy of 99.49% on the validation process is similar to the presented testing accuracy of 99.62%, which proves the model is not overfitted. Fig. 4 shows the confusion matrix, and Fig. 5 presents the AUC-ROC curve. The achieved accuracy during the validation process of multi-class is 96.62%, that is similar to the 96.70% accuracy reported during testing and proves that the proposed model is not overfitted.

To the best of the authors' knowledge, there is no automated system for COVID-19 diagnosis in the literature that includes the combination of the following features. On the one hand, the proposed model uses EfficientNet for transfer learning. On the other hand, the proposed methods are evaluated using 10-fold stratified cross-validation method. This method is used for selecting the images for testing and training. It reduces bias and ensures that all the images are used 9 times for training and 1 time for testing. On the other hand, the proposed model includes validation using an external dataset of images. In total, 296 images that have been not used during the training of the model are used for cross-validation. The validation results state similar accuracy as expected, which proves the model is not overfitted. This is detailed in Sections 4.2 and 4.4. Furthermore, the Albumentation library is used pre-processing images during each fold. This library is not used to increase the size of the datasets as presented in the related work. Instead, Albumentation is implemented to increase transfer learning performance, reduce overfitting, and improve execution time. Finally, the source code of all the experiments is available as supplementary files to allow the readers to reproduce the experiments. The proposed models have been developed and testing using Google Colab. Therefore, the software can be executed on the Google's cloud servers.

The proposed study includes ADAM optimization method. ADAM benefits of AdaGrad and RMSProp methods. Most of the similar studies also used ADAM optimizer. Furthermore, Adam is currently recommended as the default algorithm as it usually presents better results than RMSProp. Nevertheless, it is often also worth trying SGDNesterov Momentum as an alternative. The authors aim to integrate ADAM optimizer with SOM (Self-Organization Map) and PCA (Principal Component Analysis) to improve performance as proposed by the authors of [55].

In the proposed work the authors have used 10-fold stratified cross-validation and an Albumentation library for performing augmentations to pre-processing images in each fold, and not for increasing the size of training datasets as proposed the authors of [48,49]. The authors are able to use all the images at least once for training and testing both and also increasing the learning of the model. Augmentation is used for two purposes such as increasing the training set size, and another is the k-fold cross validation for pre-processing images in each fold to improve the performance of the model as proposed in [53].

In summary, the authors state the promising results of the EfficientNet architecture for automated diagnosis of COVID-19 for binary and multi-class classification. Moreover, the authors recommend the use of Albumentation and ImageDataAugmentor

Table 11
Comparison of the state-of-art models for binary classification.

Ref.	Architecture	Accuracy	Recall	Specificity	Precision	F1-Score
[43]	DarkCovidNet	98.08%	95.13%	95.3%	98.03%	96.51%
[44]	VGG19	96.33%	97.05%	96.0%	91.6%	94.24%
[45]	VGG19	98.75%	92.85%	98.75%	-	-
[45]	MobileNet v2	97.40%	99.10%	97.09%	-	-
[46]	nCOVnet	88.10%	82.00%	97.06%	97.62%	89.13%
[47]	CapsNet	97.24%	97.42%	97.04%	97.08%	97.24%
[49]	RestNet 18	99.4%	100%	98.6%	99.00%	99.5%
[50]	Semi-supervised model	93.1%	83.5%	-	89.0%	82.6%
Proposed	EfficientNet	99.62%	99.63%	-	99.64%	99.62%

Table 12
Comparison of the state-of-art models for multi-class classification.

Ref.	Architecture	Accuracy	Recall	Specificity	Precision	F1-Score
[43]	DarkCovidNet	87.02%	85.35%	92.18%	89.96%	87.37%
[45]	VGG19	93.48%	92.85%	98.75%	-	-
[45]	MobileNet v2	92.85%	99.10%	97.09%	-	-
[47]	CapsNet	84.22%	84.22%	91.79%	84.61%	84.21%
[48]	Proposed CNN	97.14%	94.61%	98.29%	-	95.75%
Proposed	EfficientNet	96.70%	96.69%	-	97.59%	97.11%

modules. The presented work contributes to the actual body of knowledge since it provides an effective solution for automated diagnosis of COVID-19. Systems such as the proposed will never aim to replace the medical professionals. Instead, these methods will support them and also reduce their exposure during the current pandemic scenario.

6. Conclusion

This paper has presented an automated system to support the diagnosis of COVID-19 patients. The proposed method implements EfficientNet architecture and has been tested using 10-fold cross-validation. Furthermore, an external dataset has been used for validation. On the one hand, the average accuracy, recall, precision and F1-score for binary classification is of 99.62%, 99.63%, 99.64% and 99.62%, respectively. On the one hand, the proposed model presents an average accuracy of 96.70% for multi-class. The average recall, precision and F1-score reported by our method is 96.69%, 97.59% and 97.11%. To the best of the authors' knowledge, there is no similar study that proposes an automated method to the detected COVID-19 using EfficientNet. The authors state relevant limitations transversal to all the methods available in the literature. First, although the critical number of individuals that have been affected by COVID-19, the datasets available are not yet robust. Nevertheless, the empirical knowledge regarding CNN applications state that the performance will increase with the proliferation of the number of samples available and used for training. Moreover, it is critical to study in detail the evaluation of this type of methods considering the evolution of the disease in the patient. The proposed methods can be effective to detect the disease in the advanced stage, but it is essential to focus on the initial stage where the methods can have lower performance. The coding scripts are provided as supplementary files. It is crucial to share all the methods and materials to allow the readers to reproduce the results. Moreover, in this way, it is possible to support future research activates. The readers can consult, update and implement different parameters to improve the results.

CRedit authorship contribution statement

Gonçalo Marques: Literature analysis, Interpretation of results, Preparation of the manuscript. **Deevyankar Agarwal:** Literature analysis, Interpretation of results, Preparation of the manuscript. **Isabel de la Torre Díez:** Literature analysis, Interpretation of results, Preparation of the manuscript, Project administration.

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.

Acknowledgments

This research has been partially supported by European Commission and the Ministry of Industry, Energy and Tourism under the project AAL-20125036 named "Wetake Care: ICT-based Solution for (Self-) Management of Daily Living".

Appendix A. Supplementary data

Supplementary material related to this article can be found online at <https://doi.org/10.1016/j.asoc.2020.106691>.

References

- [1] C. Lodigiani, G. Iapichino, L. Carenzo, M. Cecconi, P. Ferrazzi, T. Sebastian, N. Kucher, J.-D. Studt, C. Sacco, B. Alexia, M.T. Sandri, S. Barco, Venous and arterial thromboembolic complications in COVID-19 patients admitted to an academic hospital in Milan, Italy, *Thrombosis Res.* 191 (2020) 9–14, <http://dx.doi.org/10.1016/j.thromres.2020.04.024>.
- [2] C. Huang, Y. Wang, X. Li, L. Ren, J. Zhao, Y. Hu, L. Zhang, G. Fan, J. Xu, X. Gu, Z. Cheng, T. Yu, J. Xia, Y. Wei, W. Wu, X. Xie, W. Yin, H. Li, M. Liu, Y. Xiao, H. Gao, L. Guo, J. Xie, G. Wang, R. Jiang, Z. Gao, Q. Jin, J. Wang, B. Cao, Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China, *Lancet* 395 (2020) 497–506, [http://dx.doi.org/10.1016/S0140-6736\(20\)30183-5](http://dx.doi.org/10.1016/S0140-6736(20)30183-5).
- [3] Y.-Y. Zheng, Y.-T. Ma, J.-Y. Zhang, X. Xie, COVID-19 and the cardiovascular system, *Nat. Rev. Cardiol.* 17 (2020) 259–260, <http://dx.doi.org/10.1038/s41569-020-0360-5>.
- [4] Imperial College COVID-19 Response Team, S. Flaxman, S. Mishra, A. Gandy, H.J.T. Unwin, T.A. Mellan, H. Coupland, C. Whittaker, H. Zhu, T. Berah, J.W. Eaton, M. Monod, A.C. Ghani, C.A. Donnelly, S. Riley, M.A.C. Vollmer, N.M. Ferguson, L.C. Okell, S. Bhatt, Estimating the effects of non-pharmaceutical interventions on COVID-19 in Europe, *Nature* 584 (2020) 257–261, <http://dx.doi.org/10.1038/s41586-020-2405-7>.
- [5] F.-X. Lescure, L. Bouadma, D. Nguyen, M. Parisey, P.-H. Wicky, S. Behillil, A. Gaymard, M. Bouscambert-Duchamp, F. Donati, Q. Le Hingrat, V. Enouf, N. Houhou-Fidouh, M. Valette, A. Mailles, J.-C. Lucet, F. Mentre, X. Duval, D. Descamps, D. Malvy, J.-F. Timsit, B. Lina, S. van-der Werf, Y. Yazdanpanah, Clinical and virological data of the first cases of COVID-19 in Europe: A case series, *Lancet Infectious Dis.* 20 (2020) 697–706, [http://dx.doi.org/10.1016/S1473-3099\(20\)30200-0](http://dx.doi.org/10.1016/S1473-3099(20)30200-0).
- [6] A.P. Kansagra, M.S. Goyal, S. Hamilton, G.W. Albers, Collateral effect of Covid-19 on stroke evaluation in the United States, *New Engl. J. Med.* 383 (2020) 400–401, <http://dx.doi.org/10.1056/NEJMc2014816>.

- [7] A.J. Rodriguez-Morales, V. Gallego, J.P. Escalera-Antezana, C.A. Méndez, L.I. Zambrano, C. Franco-Paredes, J.A. Suárez, H.D. Rodríguez-Enciso, G.J. Balbin-Ramón, E. Savio-Larriera, A. Risquez, S. Cimerman, COVID-19 in Latin America: The implications of the first confirmed case in Brazil, *Travel Med. Infectious Dis.* 35 (2020) 101613, <http://dx.doi.org/10.1016/j.tmaid.2020.101613>.
- [8] D. Roy, S. Tripathy, S.K. Kar, N. Sharma, S.K. Verma, V. Kaushal, Study of knowledge, attitude, anxiety & perceived mental healthcare need in Indian population during COVID-19 pandemic, *Asian J. Psychiatry* 51 (2020) 102083, <http://dx.doi.org/10.1016/j.ajp.2020.102083>.
- [9] A. Remuzzi, G. Remuzzi, COVID-19 and Italy: What Next? *Lancet* 395 (2020) 1225–1228, [http://dx.doi.org/10.1016/S0140-6736\(20\)30627-9](http://dx.doi.org/10.1016/S0140-6736(20)30627-9).
- [10] H. Harapan, N. Itoh, A. Yufika, W. Winardi, S. Keam, H. Te, D. Megawati, Z. Hayati, A.L. Wagner, M. Mudatsir, Coronavirus disease 2019 (COVID-19): A literature review, *J. Infection Public Health* 13 (2020) 667–673, <http://dx.doi.org/10.1016/j.jiph.2020.03.019>.
- [11] X. Cao, COVID-19: Immunopathology And its implications for therapy, *Nat. Rev. Immunol.* 20 (2020) 269–270, <http://dx.doi.org/10.1038/s41577-020-0308-3>.
- [12] P. Mehta, D.F. McAuley, M. Brown, E. Sanchez, R.S. Tattersall, J.J. Manson, COVID-19: consider cytokine storm syndromes and immunosuppression, *Lancet* 395 (2020) 1033–1034, [http://dx.doi.org/10.1016/S0140-6736\(20\)30628-0](http://dx.doi.org/10.1016/S0140-6736(20)30628-0).
- [13] S. Al Kasab, E. Almallouhi, C.A. Holmstedt, Optimizing the Use Of Teleurology During the COVID-19 Pandemic. <http://dx.doi.org/10.1089/tmj.2020.0109>.
- [14] A. Doshi, Y. Platt, J.R. Dressen, B.K. Mathews, J.C. Siy, Keep calm and log on: Telemedicine for COVID-19 pandemic response, 2020, <http://dx.doi.org/10.12788/jhm.3419>.
- [15] S. Omboni, Telemedicine During The COVID-19 in Italy: A Missed Opportunity? <http://dx.doi.org/10.1089/tmj.2020.0106>.
- [16] A. Tashiro, R. Shaw, COVID-19 pandemic response in Japan: What is behind the initial flattening of the curve?, *Sustainability* 12 (2020) 5250, <http://dx.doi.org/10.3390/su12135250>.
- [17] D. Lee, K. Heo, Y. Seo, COVID-19 in South Korea: Lessons for developing countries, *World Dev.* 135 (2020) 105057, <http://dx.doi.org/10.1016/j.worlddev.2020.105057>.
- [18] M. Ostaszewski, A. Mazein, M.E. Gillespie, I. Kuperstein, A. Niarakis, H. Hermjakob, A.R. Pico, E.L. Willighagen, C.T. Evelo, J. Hasenauer, F. Schreiber, A. Dräger, E. Demir, O. Wolkenhauer, L.I. Furlong, E. Barillot, J. Dopazo, A. Orta-Resendiz, F. Messina, A. Valencia, A. Funahashi, H. Kitano, C. Auffray, R. Balling, R. Schneider, COVID-19 disease map, building a computational repository of SARS-CoV-2 virus–host interaction mechanisms, *Sci Data.* 7 (2020) 136, <http://dx.doi.org/10.1038/s41597-020-0477-8>.
- [19] N. Muralidharan, R. Sakthivel, D. Velmurugan, M.M. Gromiha, Computational studies of drug repurposing and synergism of lopinavir, oseltamivir and ritonavir binding with SARS-CoV-2 protease against COVID-19, *J. Biomol. Struct. Dyn.* (2020) 1–6, <http://dx.doi.org/10.1080/07391102.2020.1752802>.
- [20] Z. Li, J. Ge, M. Yang, J. Feng, M. Qiao, R. Jiang, J. Bi, G. Zhan, X. Xu, L. Wang, Q. Zhou, C. Zhou, Y. Pan, S. Liu, H. Zhang, J. Yang, B. Zhu, Y. Hu, K. Hashimoto, Y. Jia, H. Wang, R. Wang, C. Liu, C. Yang, Vicarious traumatization in the general public, members, and non-members of medical teams aiding in COVID-19 control, *Brain, Behav. Immunity* 88 (2020) 916–919, <http://dx.doi.org/10.1016/j.bbi.2020.03.007>.
- [21] D.A. Drew, L.H. Nguyen, C.J. Steves, C. Menni, M. Freydin, T. Varsavsky, C.H. Sudre, M.J. Cardoso, S. Ourselin, J. Wolf, T.D. Spector, A.T. Chan, COPE Consortium: Rapid implementation of mobile technology for real-time epidemiology of COVID-19, *Science.* 368 (2020) 1362–1367, <http://dx.doi.org/10.1126/science.abc0473>.
- [22] F. Shi, J. Wang, J. Shi, Z. Wu, Q. Wang, Z. Tang, K. He, Y. Shi, D. Shen, Review of artificial intelligence techniques in imaging data acquisition, segmentation and diagnosis for COVID-19, *IEEE Rev. Biomed. Eng.* 1 (2020) <http://dx.doi.org/10.1109/RBME.2020.2987975>.
- [23] A. Alimadadi, S. Aryal, I. Manandhar, P.B. Munroe, B. Joe, X. Cheng, Artificial intelligence and machine learning to fight COVID-19, *Physiol. Genomics* 52 (2020) 200–202, <http://dx.doi.org/10.1152/physiolgenomics.00029.2020>.
- [24] A.S.R. Srinivasa Rao, J.A. Vazquez, Identification of COVID-19 can be quicker through artificial intelligence framework using a mobile phone-based survey when cities and towns are under quarantine, *Infect. Control Hosp. Epidemiol.* 41 (2020) 826–830, <http://dx.doi.org/10.1017/ice.2020.61>.
- [25] R. Vaishya, M. Javaid, I.H. Khan, A. Haleem, Artificial intelligence (AI) applications for COVID-19 pandemic, *Diabetes Metab. Syndr. Clin. Res. Rev.* 14 (2020) 337–339, <http://dx.doi.org/10.1016/j.dsx.2020.04.012>.
- [26] E.J. Topol, High-performance medicine: the convergence of human and artificial intelligence, *Nat. Med.* 25 (2019) 44–56, <http://dx.doi.org/10.1038/s41591-018-0300-7>.
- [27] I.V. Buzzaev, V.V. Plechev, I.E. Nikolaeva, R.M. Galimova, Artificial intelligence: Neural network model as the multidisciplinary team member in clinical decision support to avoid medical mistakes, *Chronic Dis. Transl. Med.* 2 (2016) 166–172, <http://dx.doi.org/10.1016/j.cdtm.2016.09.007>.
- [28] F.H.D. Araújo, A.M. Santana, P. de A. Santos Neto, Using machine learning to support healthcare professionals in making preauthorisation decisions, *Int. J. Med. Inform.* 94 (2016) 1–7, <http://dx.doi.org/10.1016/j.ijmedinf.2016.06.007>.
- [29] L. Luo, J. Li, C. Liu, W. Shen, Using machine-learning methods to support health-care professionals in making admission decisions, *Int. J. Health Plann. Mgmt.* 34 (2019) <http://dx.doi.org/10.1002/hpm.2769>.
- [30] The Northwell COVID-19 Research Consortium, S. Debnath, D.P. Barnaby, K. Coppa, A. Makhnevich, E.J. Kim, S. Chatterjee, V. Tóth, T.J. Levy, M. d. Paradis, S.L. Cohen, J.S. Hirsch, T.P. Zanos, Machine learning to assist clinical decision-making during the COVID-19 pandemic, *Bioelectron Med.* 6 (2020) 14, <http://dx.doi.org/10.1186/s42234-020-00050-8>.
- [31] C. Dong, C.C. Loy, X. Tang, Accelerating the super-resolution convolutional neural network, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), *Computer Vision – ECCV, Springer International Publishing, Cham*, 2016, pp. 391–407, http://dx.doi.org/10.1007/978-3-319-46475-6_25.
- [32] S. Lawrence, C.L. Giles, A.D. Ah Chung Tsoi, Face recognition: a convolutional neural-network approach, *IEEE Trans. Neural Netw.* 8 (1997) 98–113, <http://dx.doi.org/10.1109/72.554195>.
- [33] D.K. Das, S. Koley, S. Bose, A.K. Maiti, B. Mitra, G. Mukherjee, P.K. Dutta, Computer aided tool for automatic detection and delineation of nucleus from oral histopathology images for OSCC screening, *Appl. Soft Comput.* 83 (2019) 105642, <http://dx.doi.org/10.1016/j.asoc.2019.105642>.
- [34] X. Ji, Q. Yu, Y. Liu, S. Kong, A recognition method for Italian alphabet gestures based on convolutional neural network, in: D.-S. Huang, V. Bevilacqua, P. Premaratne (Eds.), *Intelligent Computing Theories and Application, Springer International Publishing, Cham*, 2019, pp. 653–664, http://dx.doi.org/10.1007/978-3-030-26763-6_63.
- [35] A. Tavanaei, M. Ghodrati, S.R. Kheradpisheh, T. Masquelier, A. Maida, Deep learning in spiking neural networks, *Neural Netw.* 111 (2019) 47–63, <http://dx.doi.org/10.1016/j.neunet.2018.12.002>.
- [36] R. Karthik, M. Hariharan, Sundar Anand, Priyanka Mathikshara, Annie Johnson, R. Menaka, Attention embedded residual CNN for disease detection in tomato leaves, *Appl. Soft Comput.* 86 (2020) 105933, <http://dx.doi.org/10.1016/j.asoc.2019.105933>.
- [37] Y. Wang, X. Wei, H. Shen, L. Ding, J. Wan, Robust fusion for RGB-D tracking using CNN features, *Appl. Soft Comput.* 92 (2020) 106302, <http://dx.doi.org/10.1016/j.asoc.2020.106302>.
- [38] J.C. Rangel, J. Martínez-Gómez, C. Romero-González, I. García-Varea, M. Cazorla, Semi-supervised 3D object recognition through CNN labeling, *Appl. Soft Comput.* 65 (2018) 603–613, <http://dx.doi.org/10.1016/j.asoc.2018.02.005>.
- [39] C. Wang, Z. Zhao, Q. Ren, Y. Xu, Y. Yu, A novel multi-focus image fusion by combining simplified very deep convolutional networks and patch-based sequential reconstruction strategy, *Appl. Soft Comput.* 91 (2020) 106253, <http://dx.doi.org/10.1016/j.asoc.2020.106253>.
- [40] L.T. Duong, P.T. Nguyen, C. Di Sipio, D. Di Ruscio, Automated fruit recognition using efficientnet and mixnet, *Comput. Electron. Agric.* 171 (2020) 105326, <http://dx.doi.org/10.1016/j.compag.2020.105326>.
- [41] M. Tan, Q.V. Le, Efficientnet: Rethinking model scaling for convolutional neural networks, 2019, [arXiv:1905.11946](https://arxiv.org/abs/1905.11946) [cs, stat].
- [42] Efficientnet: Improving accuracy and efficiency through automl and model scaling, 2020, <http://ai.googleblog.com/2019/05/efficientnet-improving-accuracy-and.html>, (Accessed 15 August 2020).
- [43] T. Ozturk, M. Talo, E.A. Yildirim, U.B. Baloglu, O. Yildirim, U. Rajendra Acharya, Automated detection of COVID-19 cases using deep neural networks with X-ray images, *Comput. Biol. Med.* 121 (2020) 103792, <http://dx.doi.org/10.1016/j.combiomed.2020.103792>.
- [44] S. Vaid, R. Kalantar, M. Bhandari, Deep learning COVID-19 detection bias: accuracy through artificial intelligence, *Int. Orthopaedics* 44 (2020) 1539–1542, <http://dx.doi.org/10.1007/s00264-020-04609-7>.
- [45] 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 (2020) 635–640, <http://dx.doi.org/10.1007/s13246-020-00865-4>.
- [46] 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, Solitons Fractals* 138 (2020) 109944, <http://dx.doi.org/10.1016/j.chaos.2020.109944>.
- [47] S. Toraman, T.B. Alakus, I. Turkoglu, Convolutional capsnet: A novel artificial neural network approach to detect COVID-19 disease from X-ray images using capsule networks, *Chaos, Solitons Fractals* 140 (2020) 110122, <http://dx.doi.org/10.1016/j.chaos.2020.110122>.
- [48] M. Nour, Z. Cömert, K. Polat, A novel medical diagnosis model for COVID-19 infection detection based on deep features and Bayesian optimization, *Appl. Soft Comput.* 106580 (2020) <http://dx.doi.org/10.1016/j.asoc.2020.106580>.
- [49] S. Ahuja, B.K. Panigrahi, N. Dey, T. Gandhi, V. Rajinikanth, Deep transfer learning - based automated detection of COVID-19 from lung CT scan slices, 2020, <http://dx.doi.org/10.36227/techrxiv.12334265.v1>.

- [50] D. Konar, B.K. Panigrahi, S. Bhattacharyya, N. Dey, Auto-diagnosis of COVID-19 using lung CT images with semi-supervised shallow learning network, in: Review, 2020, <http://dx.doi.org/10.21203/rs.3.rs-34596/v1>.
- [51] Chest X-ray images (pneumonia), 2020, <https://kaggle.com/paultimothymooney/chest-xray-pneumonia>. (Accessed 15 August 2020).
- [52] J.P. Cohen, P. Morrison, L. Dao, K. Roth, T.Q. Duong, M. Ghassemi, COVID-19 image data collection: Prospective predictions are the future, 2020, [arXiv:2006.11988](https://arxiv.org/abs/2006.11988) [cs, eess, q-bio].
- [53] A. Buslaev, V.I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, A.A. Kalinin, Albumentations: Fast and flexible image augmentations, *Information* 11 (2020) 125, <http://dx.doi.org/10.3390/info11020125>.
- [54] M. Tukiainen, ImageDataAugmentor. GitHub, GitHub repository.
- [55] Md.N.Y. Ali, Md.G. Sarowar, Md.L. Rahman, J. Chaki, N. Dey, J.M.R.S. Tavares, Adam deep learning with SOM for human sentiment classification, *Int. J. Ambient Comput. Intell.* 10 (2019) 92–116, <http://dx.doi.org/10.4018/IJACI.2019070106>.