Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Expert Systems With Applications 183 (2021) 115452

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Automatic method for classifying COVID-19 patients based on chest X-ray images, using deep features and PSO-optimized XGBoost

Domingos Alves Dias Júnior [a,*], Luana Batista da Cruz [a], João Otávio Bandeira Diniz [a,b,*], Giovanni Lucca França da Silva [a], Geraldo Braz Junior [a], Aristófanes Corrêa Silva [a], Anselmo Cardoso de Paiva [a], Rodolfo Acatauassú Nunes [c], Marcelo Gattass [d]

[a] Federal University of Maranhão Av. dos Portugueses, SN, Campus do Bacanga, Bacanga, 65085-580 São Luís, MA, Brazil
[b] Federal Institute of Maranhão BR-226, SN, Campus Grajaú, Vila Nova 65940-00, Grajaú, MA, Brazil
[c] Rio de Janeiro State University, Boulevard 28 de Setembro, 77, Vila Isabel 20551-030, Rio de Janeiro, RJ, Brazil
[d] Pontifical Catholic University of Rio de Janeiro, R. São Vicente, 225, Gávea, 22453-900, Rio de Janeiro, RJ, Brazil

ARTICLE INFO

Keywords:
Chest X-Rays
COVID-19
Deep features
Medical images
Particle swarm optimization
Extreme gradient boosting

ABSTRACT

The COVID-19 pandemic, which originated in December 2019 in the city of Wuhan, China, continues to have a devastating effect on the health and well-being of the global population. Currently, approximately 8.8 million people have already been infected and more than 465,740 people have died worldwide. An important step in combating COVID-19 is the screening of infected patients using chest X-ray (CXR) images. However, this task is extremely time-consuming and prone to variability among specialists owing to its heterogeneity. Therefore, the present study aims to assist specialists in identifying COVID-19 patients from their chest radiographs, using automated computational techniques. The proposed method has four main steps: (1) the acquisition of the dataset, from two public databases; (2) the standardization of images through preprocessing; (3) the extraction of features using a deep features-based approach implemented through the networks VGG19, Inception-v3, and ResNet50; (4) the classifying of images into COVID-19 groups, using eXtreme Gradient Boosting (XGBoost) optimized by particle swarm optimization (PSO). In the best-case scenario, the proposed method achieved an accuracy of 98.71%, a precision of 98.89%, a recall of 99.63%, and an F1-score of 99.25%. In our study, we demonstrated that the problem of classifying CXR images of patients under COVID-19 and non-COVID-19 conditions can be solved efficiently by combining a deep features-based approach with a robust classifier (XGBoost) optimized by an evolutionary algorithm (PSO). The proposed method offers considerable advantages for clinicians seeking to tackle the current COVID-19 pandemic.

1. Introduction

In December, the World Health Organization began reporting problems of an unidentified pneumonia arising in Wuhan, China (Zhang et al., 2020). It was quickly deemed a global health emergency (Zhu et al., 2020). In January, the disease was identified as a cause of death on several continents. Later in January, it was named severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2), later simplified to coronavirus disease (COVID-19). COVID-19 was reported to be the cause of various cases of pneumonia (Organization, 2020).

Compared to other respiratory problems such as severe acute respiratory syndrome (SARS) and middle east respiratory syndrome (MERS), COVID-19 produces the greatest number of contagions in the smallest amount of time (Petrosillo, Viceconte, Ergonul, Ippolito, & Petersen, 2020). An individual who exhibits COVID-19 symptoms should self-isolate at home for 14 days. This mechanism is very much in line with the fact that the transmission of this pathology is very sensitive to any contact with the infected person (Boccia, Ricciardi, & Ioannidis, 2020).

The main symptoms of COVID-19 include fever, cough, and fatigue; other symptoms can include sputum production, headache, hemoptysis, diarrhea, dyspnoea, and lymphopenia (Rothan & Byrareddy, 2020). The

* Corresponding author at: Federal Institute of Maranhão BR-226, SN, Campus Grajaú, Vila Nova 65940-00, Grajaú, MA, Brazil (J.O. Bandeira Diniz).
E-mail addresses: domingos.adj@nca.ufma.br (D.A. Dias Júnior), luana.b.cruz@nca.ufma.br (L.B. da Cruz), joao.bandeira@ifma.edu.br (J.O. Bandeira Diniz), gioh.lucca@gmail.com (G.L. França da Silva), geraldo@nca.ufma.br (G.B. Junior), ari@dee.ufma.br (A.C. Silva), paiva@deinf.ufma.br (A.C. de Paiva), rodolfoacatauassu@yahoo.com.br (R.A. Nunes), mgattass@tecgraf.puc-rio.br (M. Gattass).

https://doi.org/10.1016/j.eswa.2021.115452
Received 30 September 2020; Received in revised form 18 February 2021; Accepted 14 June 2021
Available online 22 June 2021

global community is currently seeking answers and solutions to eradicate this pandemic; thus, several case studies have been developed to investigate the particular characteristics of the disease. One of the main mechanisms considered to assist specialists is the evaluation of chest X-ray (CXR) images (El-Din Hemdan, Shouman, & Karar, 2020).

Because the imaging procedure limits the patient's exposure to high radiation doses, the CXR results exhibit only a slight contrast to distinguish soft tissues (Kroft et al., 2019). Thus, many computational methods have been proposed to assist specialists. Computer-aided detection and diagnostics (CAD and CADx, respectively) play important roles in this task (Diniz et al., 2018; Diniz et al., 2018; Diniz, Diniz, Valente, Silva, & Paiva, 2019; Souza et al., 2019; Carvalho et al., 2020; Cruz et al., 2020; Diniz, Ferreira, Diniz, Silva, & de Paiva, 2020). Thus, the literature contains several CXR-based methods suitable for identifying patients with COVID-19.

Artificial intelligence-based techniques have been successful in diagnosing diseases, and the number of COVID-19 cases worldwide has increased dramatically; thus, there is a need for automated systems to detect and diagnose diseases using established techniques such as deep learning and deep features. In the recent work of Narin, Kaya, and Pamuk (2020), models based on convolutional neural networks (ResNet50, Inception-v3, and Inception-ResNetV2) were applied to detect patients infected with COVID-19; a maximum accuracy of 98% was obtained. Apostolopoulos and Mpesiana (2020) also used different pre-trained deep learning models; they achieved a 98.75% accuracy and a 92.85% recall using two classes (COVID-19 vs. normal conditions).

In El-Din Hemdan et al. (2020) and Zhang, Xie, Li, Shen, and Xia (2020) also presented significant results for the CXR-based detection of patients with COVID-19. In both studies, several models of convolutional neural network (CNN) were used to verify the COVID-19 detection performance. In El-Din Hemdan et al. (2020), a 90% accuracy was obtained for the normal and COVID-19 classes. In the Zhang et al. (2020), achieved a recall of 96% for cases of COVID-19 and 70.65% for normal cases.

Pereira, Bertolini, Teixeira, Silla, and Costa (2020) used CXR images to distinguish pneumonia cases caused by COVID-19 from those caused by other types of diseases, as well as from healthy lungs. To this end, the authors used known texture descriptors to extract features and a pre-trained CNN model. They achieved an F1-score of 65% using a multi-class approach and an F1-score of 89% for the identification of COVID-19. In a study by Ozturk et al. (2020), a model (DarkCovidNet) for binary and multiclass classification was also developed. DarkCovidNet was inspired by the object detection system referred to as "You Only Look Once." The model achieved an accuracy of 98.08% for binary class and 87.02% for multiclass classification.

Sethy and Behera (2020) classified features obtained from the ResNet50 model, using a support vector machine (SVM) classifier; they obtained an accuracy of 95.38% and an F1-score of 91.41% in the detection of COVID-19. In Rahimzadeh and Attar (2020), a neural network was proposed; it used a combination of Xception and ResNet50V2 networks to detect COVID-19 cases, achieving an accuracy of 99.50%, a precision of 35.27%, and a recall of 80.53% in the detection of COVID-19. Whilst all of these approaches have achieved strong results, significant amount of work remains to be done; for instance, a broad range of rigorous tests and improvements are required before these techniques can be used in clinical practice.

Therefore, the present work proposes a method capable of classifying the CXR images of patients with COVID-19, implementing extraction via deep features and using an optimized XGBoost for classification. In this way, we hope to assist specialists in the task of diagnosing patients, by providing a robust mechanism for CXR analysis.We chose XGBoost as the classifier of the Deep Features extracted, because of its performance, compared to other existing solutions. According to Chen, He, Benesty, Khotilovich, and Tang (2015), XGBoost has performed better in terms of speed, scalability, memory consumption, and hardware resources. Besides, it has achieved state-of-the-art results in several tasks.

The key contributions of this work are summarized as follows: (a) this study proposes a fully automated method for classifying CXR images as showing COVID-19 or non-COVID-19 cases; (b) it develops a method capable of extracting deep features from CXR images; (c) it uses XGBoost as a classifier for deep features; and (d) it optimizes the XGBoost parameters using an evolutionary algorithm. With this, we provide a promising CADx tool, capable of assisting specialists in the task of identifying patients with COVID-19.

This paper is organized as follows. Section 2 describes the materials and methods used to classify COVID-19 patients from their CXR. Sections 3 and 4 present the experiments conducted to validate our research, our discussion of the corresponding results, and comparative analyses of other similar studies. Finally, our conclusions and future works are presented in Section 5.

2. Materials and method

In this section, the image datasets and the methods used to classify patients exhibiting COVID-19 and non-COVID-19 conditions are described in four steps, which are considered in detail in the next subsections. In the first section, we describe the test images acquired from the COVID chest X-ray dataset (Cohen, Morrison, & Dao, 2020) and Guangzhou Women and Children's Medical Center, Guangzhou (Kermany, Zhang, & Goldbaum, 2018). The second section describes the preprocessing applied to these images, through which we standardized them as input to our deep feature extractor. In the third section, we describe the techniques used to extract the features. Finally, in the fourth section, we present the classifier used to perform the experiments. Fig. 1 illustrates the steps of the proposed method.

2.1. Datasets

To develop a robust method, diversified samples are needed; in this study, these were acquired under normal clinical standards (i.e., without previous standardization) to improve the study. However, due to its nature as an ongoing and unprecedented problem, no such extensive database is available for patients affected by COVID-19.

The classification of CXR images for various types of pathology has been well-studied in the literature (Chapman, Fizman, Chapman, & Haug, 2001; Rajpurkar et al., 2017). Several datasets are available, containing many diverse classes of CXR images. For this work, we sought a database containing several samples of normal patients. This was for two reasons: (1) to facilitate the extraction of features from a diverse sample range, for the model to learn patterns from; (2) to develop a robust method trained upon a considerable number of normal patients (i.e., to simulate clinical practice). Thus, all experiments in this study were performed on datasets acquired from two public repositories containing CXR images.

The first dataset, referred to as the COVID chest X-ray dataset, consists of 660 chest X-ray/computed tomography images (Cohen et al., 2020). These include images of patients with acute respiratory distress syndrome (ARDS), COVID-19, MERS, pneumonia, and SARS. This collection of images was published by Dr. Joseph Cohene, Dr. Adrian Rosebrock, and Dr. Lan Dao.

Several images in this dataset are classified as multiple types of pathology. This is thought to occur as a result of suspicious diagnoses and/or because the image contains pre-existing characteristics of other pathologies, such as SARS, MERS, ARDS, or others. The present work only considered images that exclusively presented COVID-19 as the pathology. Furthermore, the dataset contained images with different views (only left lung, only right lung, etc.). Here, only frontal images of the two lungs were considered.

The second set of data was collected at Guangzhou Women and Children's Medical Center, Guangzhou (Kermany et al., 2018). The dataset contains 5,856 CXR images, of which 4273 images are from patients exhibiting pneumonia (bacterial or viral) and 1583 are from

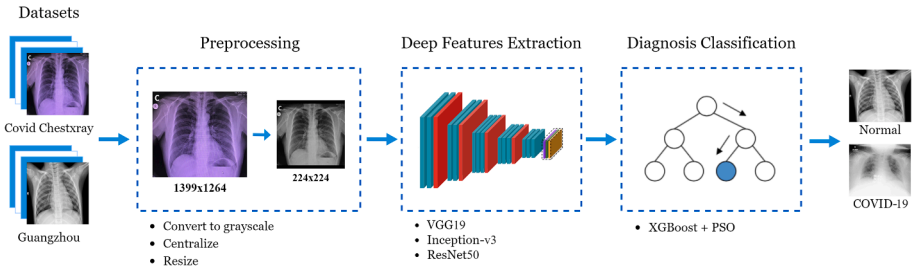

Fig. 1. Flowchart of the method.

patients with healthy lungs; these data were divided into training and testing datasets. For this work, we used only the training dataset, which contained more samples; hereafter, this is referred to as the "Guangzhou dataset.".

In our study, the experiments were performed with the goal of classifying patients into COVID-19 and non-COVID-19 classes. For this, 206 COVID-19-condition (Cohen et al., 2020) and 1341 normal-condition images (Kermany et al., 2018) were selected. Fig. 2 (a) shows examples of the CXR images of patients exhibiting normal conditions, and Fig. 2 (b) shows those of patients diagnosed with COVID-19.

2.2. Preprocessing

The images acquired from the two databases were not standardized, neither in the CXR digitalization process or the exam protocols. To standardize the CXR images used in this method, a preprocessing step is implemented. This step is divided into three sub-steps. First, it is observed that the images from the COVID chest X-ray dataset (Fig. 2(a)) present color-scale patterns other than grayscale. This prevents us from using them, because the classifier risks being biased by these patterns. Thus, to circumvent this problem and to keep the method concise, all images (from both the COVID chest X-ray and Guangzhou datasets) were converted to grayscale (Fig. 3(b)).

It can be observed that the images are of different widths and heights. Thus, we implemented a resizing that left the internal structures of the CXR images intact. First, we found the largest axis and created a rectangular image whose heights and widths matched the size of this axis. Then, the original image was centered on this new square image (Fig. 3(c)).

Finally, all images were resized to 224×224 pixels. This size was selected as being suitable for further processing within the deep learning pipeline (El-Din Hemdan et al., 2020); however, it did not diminish the resolution of the collected images. Fig. 3 illustrates the preprocessing steps.

2.3. Deep features extraction

In this step, we aim to obtain descriptive measurements of the lungs featured in the CXR images; these measurements will be used in the classification step. Extracting features from an image is an arduous task which requires a high level of expertise and therefore considerable time and effort. In addition to being a non-trivial task, the features obtained are specific to the problem; hence, the solutions may not be sufficiently representative of other cases. With the development of deep neural network models, it has become possible to extract features in an automated way (LeCun, Bengio, & Hinton, 2015).

The layers of a CNN act as feature extractors. Lower layers learn basic features (e.g., contours and borders), middle layers extract information such as color and shape, and deeper layers learn to identify the object in the image. Moreover, these networks feature a fully connected layer that acts as a classifier (H Mohamed, H El-Behaidy, Khoriba, & Li, 2020).

The technique used to extract features in this work is referred to as "Deep Features." It consists of removing the fully connected layer from a conventional CNN to obtain a feature vector as the network output. Thus, the CNN is converted into an automated feature extractor. Fig. 4 illustrates the approach.

Three network architectures were selected to conduct our experiments: VGG19 (Simonyan & Zisserman, 2014), Resnet50 (He, Zhang, Ren, & Sun, 2016), and Inception-v3 (Szegedy, Vanhoucke, Ioffe, Shlens, & Wojna, 2016). The pre-trained weights applied to the networks were trained on the ImageNet database (Deng et al., 2009).

2.4. Diagnosis classification using XGBoost

From the features extracted by Deep Features, we need to classify the image as depicting a patient exhibiting COVID-19 or non-COVID-19 conditions. For this, deep learning techniques are used to classify deep features. In this work, the XGBoost classifier was used, owing to its strong performance (in terms of speed, scalability, memory consumption, and hardware resources (Chen et al., 2015)) compared to other existing solutions.

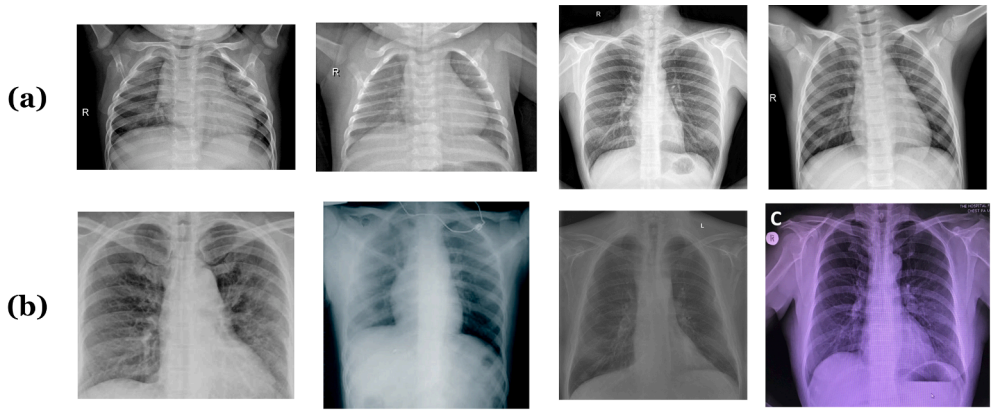

Fig. 2. Dataset information: (a) CXR images of patients exhibiting normal conditions; (b) CXR images of patients diagnosed with COVID-19.

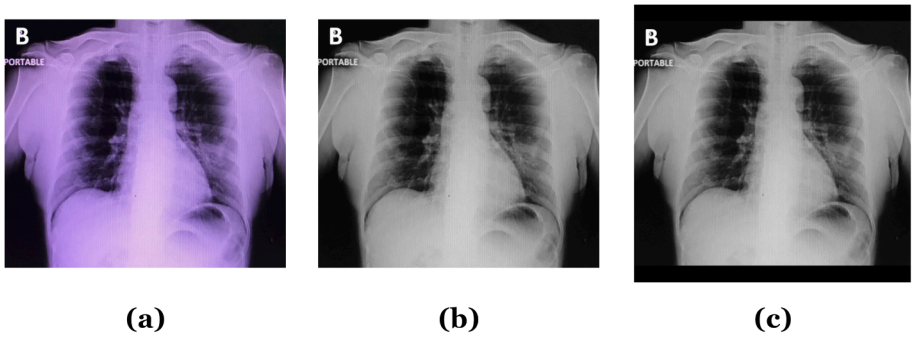

Fig. 3. Preprocessing: (a) input image; (b) color space in grayscale; (c) proportionally resized and centered.

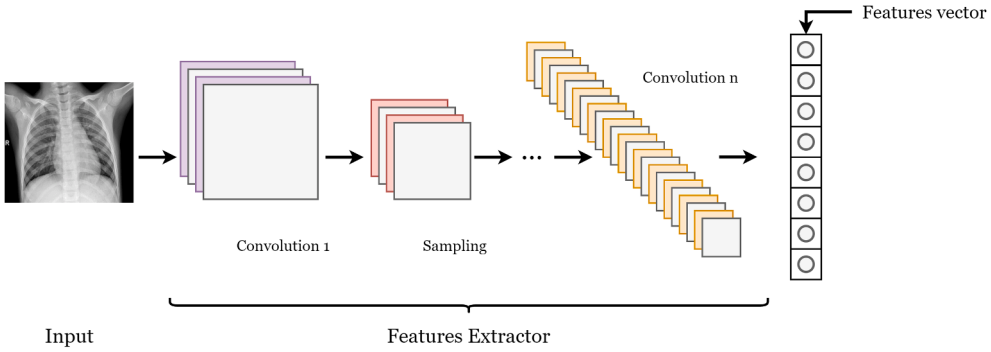

Fig. 4. CNN Architecture.

XGBoost consists of the library proposed in (Chen et al., 2015) and is based on the gradient increase framework developed by Friedman (2001). It can be used for various objective functions, including regression and sorting (Song, Chen, Deng, & Li, 2016). XGBoost offers an efficient and scalable implementation of the gradient-based decision tree algorithm. It has been widely applied and its efficiency has been recognized in several machine learning and data mining challenges (e.g., the Kaggle website). It has achieved state-of-the-art results for a wide range of task problems (Chen et al., 2015).

XGBoost is extensible, and its parameters can be easily changed (Carvalho et al., 2020) and optimized. Its parameters include max depth, learning rate, range, colsample by tree, min child weight, and fitness. Max depth denotes the maximum depth of the tree. Increasing this property increases the complexity of the model; however, the propensity for overfitting is also increased. The learning rate determines the size of the model's evolution step in each iteration. Gamma denotes the minimum loss reduction required to form an additional partition in a leaf node of the tree; increasing this makes the algorithm more conservative. Colsample by tree indicates the column subsample rate when building each tree. Min child weight denotes the minimum sum of the instance weights (Hessian) required in a child node; increasing its value makes the algorithm more conservative. The fitness parameter specifies the task and the learning objective (Chen, He, & Benesty, 2018).

As described, XGBoost features several parameters. In computational problems, one method of finding the optimal parameters is to observe how these parameters change during the validation phase. However, this is an arduous and ineffective process. To make our method even more automatic and robust, we propose using a particle swarm optimization (PSO) to optimize the parameters.

2.4.1. *XGBoost parameter optimization using PSO*

The PSO algorithm is an evolutionary technique inspired by the swarming and collaborative behavior of biological populations (Eberhart & Kennedy, 1995). It seeks an ideal solution by iteratively changing particle speeds and positions according to the particle's and group's flight experiences, guiding them towards the location of Gbest and Pbest in the subsequent iterations. Gbest corresponds to the optimum population fitness value achieved by any particle, whereas Pbest corresponds to the optimum particle fitness value achieved so far (Kennedy, 2010).

Also, PSO can generate a high-quality solution within a shorter calculation time and exhibiting more effective stable convergence characteristics than other optimization techniques. Moreover, there are fewer control parameters to adjust, and it is more efficient in maintaining the diversity of the swarm as all the particles use the information related to the most successful particle (Gbest particle) to enhance themselves. For this reason, we chose the PSO to optimize the XGBoost parameters (Kennedy, 2010; da Silva, Valente, Silva, de Paiva, & Gattass, 2018).

The XGBoost parameters used for optimization are those described in Section 2.4: max depth, colsample by tree, min child weight, gamma, and learning rate. To represent a PSO particle, a five-position vector is created, in which each vector component represents one of the aforementioned parameters. Each parameter requires a search space, which is simply denoted by the limits of the maximum and minimum values that can be assumed. For the max depth, a discrete threshold was defined between three and ten. The min child weights were set as integer values between one and seven. Finally, for the other parameters, continuous thresholds were defined between zero and one. Furthermore, to allow the PSO algorithm to evolve, a fitness function was required.

The fitness of each particle was evaluated using the results obtained by the XGBoost model, which operated on the validation subset according to the selected parameters. Here, we explain the weighting method used in fitness. The main purpose of weighting is to maintain a balance between recall and precision; for this, we used the F-score (Sasaki & Fellow, 2007) to obtain superior models in the classification of COVID-19 cases. This fitness is defined as Eq. 1.

$$Fitness = F - Score = 2 * \frac{Pre * Rec}{Pre + Rec} \tag{1}$$

where $Pre = \frac{TP}{TP+FP}$ and $Rec = \frac{TP}{TP+FN}$. True positive (TP) indicates the

correctly detected cases. False positive (FP) denotes the negative cases mistakenly detected as positive. True negative (TN) refers to the truly detected negative cases. False negative (FN) denotes the positive cases mistakenly detected as negative.

After defining a particle, its thresholds, and its fitness function, the following steps were performed (Le, Nguyen, Zhou, Dou, & Moayedi, 2019):

- Step 1: A population of ten random particles was created. Then, the fitness function of each particle was computed.
- Step 2: Each particle traveled circularly through the search space at an initial speed, as established in the previous step. For each iteration, the optimal local particle was sought. The best overall result represents the current best particle. The speeds were also updated during this step.
- Step 3: After the speed was calculated and updated, the particles flew in the search space at this new speed.
- Step 4: The best location and best overall result were updated to the best position, according to the fitness function.
- Step 5: The search stop condition was checked. If the fitness function of the particle was optimal, the search was interrupted. Otherwise, we returned to Step 2.

At the end of this optimization, we obtained the best XGBoost parameters of the model validation step. Finally, a test database was applied, and validation metrics were extracted to calculate the method's robustness.

3. Results

This section describes the experimental setup, the experimental procedures performed to validate the proposed method, the validation metrics adopted, and the database preparation procedure.

3.1. Experimental setup

The proposed method was implemented using the Python library. We primarily used the Keras deep learning library (Chollet et al., 2015) with TensorFlow-GPU (Abadi et al., 2015) as the back-end. The computer used in these experiments consisted of an Intel Core i7-7700 K 4.20 GHz CPU, 16 GB of RAM, and Nvidia GeForce GTX 1080-Ti graphics card, running on a Windows 10 operating system.

3.2. Evaluation metrics

To validate the method, we applied popular metrics from the field of medical imaging; these consisted of accuracy (Acc), precision (Pre), recall (Rec), and F1-Score (F1) (Duda, 1973). The metrics were calculated based on the confusion matrix.

As explained in Section 2, two public datasets were used to implement the proposed method and measure its efficiency through experiments. Of the images contained in these datasets, 206 were from the COVID-19 chest X-ray dataset and contained cases classified as COVID-19; 1341 images were of normal patients, taken from the Guangzhou Dataset.

To validate our method, we applied the k-fold cross-validation technique, setting $k = 5$. In this approach, the method is trained and tested five times, by dividing the database in an 80:20 ratio to ensure the proportion for both classes. At the end of each training procedure, the validation metrics were extracted. As a result, we obtained the average of each metric along with its standard deviation; this enabled us to make more appropriate analyses of the method. It also made it possible to avoid the overfitting problem, because the entire database was trained at least once in the k-fold validation procedure.

Table 1 shows the relationships between the individuals used for training and testing in each fold, for both the normal and COVID-19 cases.

Table 1
Relationship between individuals used for training and testing in each fold.

Dataset	Proportion	Normal	COVID-19	Total Sample
Train	80%	1071	165	1236
Test	20%	270	41	311
Total	100%	1341	206	1547

3.3. Experimental results

For our experimental results, we studied the execution of the entire method and the collection of validation metrics. However, when proposing a new and robust method, it is necessary to compare it against other methods already established in the literature. Thus, we conducted several experiments to demonstrate the robustness of our method compared to other techniques. All metrics are shown in terms of the average of the five folds as well as the standard deviation.

3.3.1. Confusion matrix

To assess the robustness of the method, we first present the summation confusion matrix of the fivefold test results. From this matrix, the accuracy, precision, recall, and F1-score metrics were calculated. The results of the confusion matrix are presented in Table 2.

3.3.2. Deep features with XGBoost (with and without PSO)

First, we demonstrate the execution of our entire method and the collection of metrics. However, it should be noted that we propose a step of parameter optimization. To prove the necessity of this step, we show the results both with and without the use of PSO. For the case without PSO, only the default parameters were used. Table 3 describes the training time for a fold as well as the default parameters and those found after optimization for the three deep feature extractors.

Notably, after the training and model creation procedures, the classification for an individual patient can be performed immediately. After optimization, the training and metric testing for the method with and without PSO were performed. The results are displayed in Table 4.

We can see that Deep Features was effective in capturing the content of the images. Promising results were seen for both models (i.e., with and without PSO). When considering accuracy and F1-score, we observe a similar average correctness between classes; furthermore, we can see a correspondence between precision and recall, with both metrics in the two types of networks (with and without PSO) exceeding 97% and 98%, respectively. Despite only a slight gain in metrics, the use of PSO is necessary. This is because the parameter selection step is costly and depends on the expertise of the method developer. By using an optimized search method, we have further automated the method and made it independent of the parameter selection.

3.3.3. XGBoost + PSO vs Other Optimization Methods

To verify the effectiveness of the PSO in optimizing the parameters of the XGBoost, we did experiments using two other optimizers:

1. Genetic Algorithm (GA) (Mirjalili, 2019): A genetic algorithm is a search heuristic that is inspired by Charles Darwin's theory of natural evolution. This algorithm reflects the process of natural selection where the fittest individuals are selected for reproduction to produce offspring of the next generation. Usually, its parameters are the

Table 2
Summation fivefold confusion matrix.

	True Positive	True Negative
Predicted Positive	1335	15
Predicted Negative	5	200

Table 3
XGBoost parameters: default and with PSO.

Parameters	Deep Features	Time (sec)	Max depth	Colsample by tree	Min child weight	Gamma	Learning rate
Default	—-	434	6	1	1	0	0.3
XGBoost + PSO	VGG19	732	8	0.381	7	0.352	0.408
	Inception-v3	811	5	0.769	7	0.917	0.519
	ResNet50	698	9	0.178	5	0.179	0.488

Table 4
Classification of COVID-19 Chest X-ray images: XGBoost with PSO and without.

Optmization	Deep Features	Time(sec)	Acc(%)	Pre(%)	Rec(%)	F1(%)
Only XGBoost	VGG19	340	97.74 ± 1.21	97.81 ± 1.14	99.7 ± 0.31	98.75 ± 0.60
	Inception-v3	530	97.16 ± 0.95	98.43 ± 0.40	98.29 ± 0.85	98.36 ± 0.55
	ResNet50	432	97.67 ± 0.42	97.88 ± 0.68	99.48 ± 0.33	98.67 ± 0.24
XGBoost + PSO	VGG19	732	98.71 ± 0.32	98.89 ± 0.53	99.63 ± 0.65	99.25 ± 0.20
	Inception-v3	811	97.48 ± 1.08	98.29 ± 1.35	98.81 ± 0.79	98.54 ± 0.64
	ResNet50	698	98.51 ± 0.84	98.80 ± 0.95	99.48 ± 0.34	99.14 ± 0.51

number of iterations, mutation rate, crossover rate, size of the individual, and number of population. For the experiments, default parameters were used.

2. Bee Colony Optimization (BCO) (Teodorovic, Lucic, Markovic, & Dell'Orco, 2006): An optimization algorithm based on the intelligent foraging behavior of honey bee swarm. three groups of bees: employed bees, onlookers, and scouts. It is assumed that there is only one artificial employed bee for each food source. In other words, the number of employed bees in the colony is equal to the number of food sources around the hive. Employed bees go to their food source and come back to hive and dance in this area. The employed bee whose food source has been abandoned becomes a scout and starts to search for finding a new food source. Onlookers watch the dances of employed bees and choose food sources depending on dances. Usually, its parameters are the number of iterations, colony rate, scouts, and size of the individual. For the experiments, default value parameters were used.

To optimize the XGBoost, the GA and BCO algorithms were also executed in the parameters of Max depth, Colsample by tree, Max child weight, Gamma, and Learning Rate. The result of these parameters for the two algorithms executed in the three networks is expressed in Table 5 together with the parameters found by the PSO.

With all XGBoost parameters optimized for the three networks using GA, BCO, and PSO, the next step is to apply the models on the test datasets and calculate the metrics to verify their effectiveness and highlight why the PSO is better than the other algorithms optimization. Table 6 presents a comparison of the validation metrics and the processing time of these combinations.

We can see that all three optimizers were able to achieve promising results, superior to using XGBoost without optimization. However, as presented in Section 2 and in the *Time* column of the Table 6, PSO was the fastest optimize. Also, in the metric of F1-Score, which expresses the harmony between the metrics of Precision and Recall, PSO was the only one capable of reaching 99% in two types of architecture (VGG19 and ResNet50). The same occurs in the F1-Score variance in these two architectures, mainly in VGG19, where there was only 0.20% standard deviation. Thus, we highlight how robust PSO was to optimize effectively, quickly, and achieve promising validation metrics in the XGBoost classifier.

3.3.4. Traditional texture features vs. deep features with XGBoost

A second test that highlights the innovations proposed by deep feature extraction, is the use of texture features; this method is well-established in the literature. Thus, we performed the whole method again but exchanged the Deep Features extraction step for traditional texture feature extraction (Hu moments, Haralick, and local binary pattern (LBP) (Gonzalez & Woods, 2008)). The comparison of these approaches against the optimized XGBoost is presented in Table 7.

It can be seen that the use of Deep Features offers superior results to the existing feature-extraction techniques. Furthermore, it is known that extracting features manually is a non-trivial task. The type of features (e.g., texture, shape, geometry, or border) chosen depends on the elements of the image and its specific characteristics. By proposing the use of Deep Features, our method can extract features automatically, without needing to explicitly perform feature extraction and selection steps. Furthermore, metrics based on Deep Features and optimized XGBoost are more consistent in each fold, showing smaller standard deviation.

3.3.5. Deep features with other classifiers

Another experiment to validate the use of optimized XGBoost is to compare it against traditional classifiers presented in the literature. To assess the robustness of optimized XGBoost, we use a regression-based classifier (logistic regression (Kleinbaum, Dietz, Gail, Klein, & Klein, 2002)) and another decision tree-based classifier (random forest (Breiman, 2001)). The results of this experiment are presented in Table 8.

Table 5
XGBoost parameters optimized with GA, BCO, and PSO.

Parameters	Deep Features	Max depth	Colsample by tree	Min child weight	Gamma	Learning rate
XGBoost + GA	VGG19	3	1	8	0.402	0.950
	Inception-v3	7	1	2	0.679	0.318
	ResNet50	2	0.126	1	0.198	0.089
XGBoost + BCO	VGG19	8	0.807	3	0.207	1
	Inception-v3	3	0.644	3	0.138	0.215
	ResNet50	8	0.169	2	0.285	0.485
XGBoost + PSO	VGG19	8	0.381	7	0.352	0.408
	Inception-v3	5	0.769	7	0.917	0.519
	ResNet50	9	0.178	5	0.179	0.488

Table 6
Comparison of XGBoost optimized with GA, BCO, and PSO.

Optimization	Extraction	Time(sec)	Acc(%)	Pre(%)	Rec(%)	F1(%)
XGBoost + GA	VGG19	930	98.06 ± 0.60	99.25 ± 0.70	98.51 ± 0.63	98.88 ± 0.45
	Inception-v3	1461	97.29 ± 0.51	98.22 ± 0.64	98.66 ± 0.51	98.44 ± 0.19
	ResNet50	1076	97.80 ± 0.52	98.09 ± 0.59	99.41 ± 0.29	98.74 ± 0.29
XGBoost + ACO	VGG19	840	98.13 ± 0.32	98.45 ± 0.44	99.41 ± 0.44	98.92 ± 0.18
	Inception-v3	937	97.35 ± 0.56	98.30 ± 0.36	98.66 ± 0.50	98.48 ± 0.31
	ResNet50	726	97.80 ± 0.71	98.23 ± 0.49	99.26 ± 0.53	98.74 ± 0.32
XGBoost + PSO	VGG19	732	98.71 ± 0.32	98.89 ± 0.53	99.63 ± 0.65	99.25 ± 0.20
	Inception-v3	811	97.48 ± 1.08	98.29 ± 1.35	98.81 ± 0.79	98.54 ± 0.64
	ResNet50	698	98.51 ± 0.84	98.80 ± 0.95	99.48 ± 0.34	99.14 ± 0.51

Table 7
Classification of COVID-19 chest X-ray images: traditional texture features vs. Deep Features.

Extraction	Features	Time(sec)	Acc(%)	Pre(%)	Rec(%)	F1(%)
Traditional Texture	Hu moments (Gonzalez & Woods, 2008)	332	89.01 ± 1.84	89.00 ± 1.80	99.63 ± 0.60	94.00 ± 1.07
	Haralick (Gonzalez & Woods, 2008)	357	95.22 ± 1.94	96.41 ± 1.50	98.11 ± 1.49	97.24 ± 1.16
	LBP (Gonzalez & Woods, 2008)	491	94.45 ± 1.95	95.32 ± 2.04	98.46 ± 1.11	96.85 ± 1.10
XGBoost + PSO	VGG19	732	98.71 ± 0.32	98.89 ± 0.53	99.63 ± 0.65	99.25 ± 0.20
	Inception-v3	811	97.48 ± 1.08	98.29 ± 1.35	98.81 ± 0.79	98.54 ± 0.64
	ResNet50	698	98.51 ± 0.84	98.80 ± 0.95	99.48 ± 0.34	99.14 ± 0.51

Table 8
Classification of COVID-19 chest X-ray images: other classifiers vs. XGBoost.

Classifiers	Deep Features	Time(sec)	Acc(%)	Pre(%)	Rec(%)	F1(%)
Random Forest	VGG19	371	97.74 ± 1.21	97.81 ± 1.14	99.70 ± 0.31	98.75 ± 0.60
	Inception-v3	556	97.16 ± 0.95	98.43 ± 0.40	98.29 ± 0.85	98.36 ± 0.55
	ResNet50	461	97.67 ± 0.42	97.88 ± 0.68	99.48 ± 0.33	98.67 ± 0.24
Logistic Regression	VGG19	332	96.83 ± 0.96	98.13 ± 0.71	98.20 ± 0.69	98.17 ± 0.57
	Inception-v3	499	97.48 ± 1.00	98.65 ± 0.65	98.43 ± 0.85	98.54 ± 0.60
	ResNet50	416	98.26 ± 0.74	98.66 ± 0.90	99.33 ± 0.31	98.99 ± 0.44
XGBoost + PSO	VGG19	732	98.71 ± 0.32	98.89 ± 0.53	99.63 ± 0.65	99.25 ± 0.20
	Inception-v3	811	97.48 ± 1.08	98.29 ± 1.35	98.81 ± 0.79	98.54 ± 0.64
	ResNet50	698	98.51 ± 0.84	98.80 ± 0.95	99.48 ± 0.34	99.14 ± 0.51

Analyzing Table 8, it can be seen that deep features play a fundamental role in extracting features. Even when using other classifiers, the results are still effective and present a low standard deviation; we highlight the use of logistic regression with ResNet50, which attains an average accuracy of 98.26%. However, when using the proposed method with XGBoost + PSO, the results are still superior to those obtained using other classifiers; this is seen in all metrics, though predominantly in the accuracy of all three deep feature architectures.

3.3.6. *Deep learning methods vs. XGBoost + PSO*

Finally, it is known that CNNs play an important role in several imaging domains, and medical imaging is no exception. Thus, to validate our method (when using only Deep Features), we compare it against a method that uses end-to-end deep learning. However, given its computational cost, deep learning can be performed only once. We selected 70% of the dataset for training, 10% for validation, and 20% for testing. Furthermore, we used the standard architectures of VGG19, Inception-v3, and ResNet50. All networks were executed with 50 epochs. The results are described in Table 9.

The effectiveness of deep learning methods is known, and their effectiveness holds for the classification of patients into COVID-19 and non-COVID-19 groups. The method achieves good results for the task. However, deep learning network training is computationally expensive. Moreover, because it takes a long time, the search for parameters (to create a robust model) becomes even more complex. Conversely, XGBoost is a tree-based classifier that offers high performance in terms of speed, scalability, memory consumption, and hardware resources Chen et al., 2015. Nevertheless, the results of our method proved to be

Table 9
Classification of COVID-19 chest X-ray images: deep learning vs. XGBoost + PSO.

Methods	Deep Features	Time(sec)	Acc(%)	Pre(%)	Rec(%)	F1(%)
Deep Learning	VGG19	3300	89.94	89.87	99.59	94.5
	Inception-v3	7250	91.23	91.10	99.59	95.17
	ResNet50	4150	94.47	94.00	99.96	96.91
XGBoost + PSO	VGG19	732	98.71 ± 0.32	98.89 ± 0.53	99.63 ± 0.65	99.25 ± 0.20
	Inception-v3	811	97.48 ± 1.08	98.29 ± 1.35	98.81 ± 0.79	98.54 ± 0.64
	ResNet50	698	98.51 ± 0.84	98.80 ± 0.95	99.48 ± 0.34	99.14 ± 0.51

superior in all metrics compared with deep learning methods.

Thus, based on the experiments presented, we conclude that the proposed method shows great promise. Combined with clinical practice and specialist expertise, this robust, fast method consumes minimal computational resources and can constitute a fundamental tool for combating the COVID-19 pandemic. It should be noted that in developing countries, where resources are scarce, this method could offer considerable advantages, given its efficiency characteristics and accuracy.

4. Discussion

In this section, the results obtained with the method are discussed. Furthermore, a qualitative assessment is made for two case studies: a correct classification and an incorrect classification. For this, we chose the deep feature extraction model that exhibited the optimum accuracy (i.e., VGG19). Still, we present results applying the model achieved by the method in patients with different types of pneumonia (bacterial and viral) to show the efficiency of the model built. Then, the studies found in the literature are considered and compared. Finally, the advantages and limitations of the proposed method are discussed.

4.1. Case study

Initially, we assessed the significance of the COVID-19 and normal image regions, by projecting the weights of the output layer onto the maps of convolutional features (Zhou, Khosla, Lapedriza, Oliva, & Torralba, 2016). For this, we used the activation maps for the images of patients under normal conditions and those affected by COVID-19, as shown in Fig. 5 (a) and (b), respectively. In the activation map, the closer a feature is to the red region, the more important its extraction is for the method decision. Features closer to the blue regions are less important. From this, we can see that, despite the different databases, the main features are consistently extracted from the pulmonary region; this indicates that our method is not biased by the choice of database.

Therefore, we presented a set of case studies in the next subsections to analyze and discuss the most common situations after applying the proposed method.

4.1.1. Case study - Patients affected by COVID-19

To qualitatively evaluate the method, we considered three case studies of patients affected by COVID-19. In the first case, the model misclassified the patient, and in the other two, the model's diagnosis was correct.

The first case study is a patient diagnosed by the specialist as having COVID-19. However, the model using deep features (from VGG19) and XGBoost + PSO classified it as normal. This example is illustrated in Fig. 6.

Despite this classification, by analyzing the image we can see that the regions of the lungs are very clean, with no evidence of impairment. It is worth remembering that our diagnosis is based only upon the image, whereas the diagnosis of the specialist takes into account other factors (e.g., the clinical and laboratory conditions) of the patient. However, as we evaluated only the image, this was considered an error case.

COVID-19 in its early stages does not cause severe pulmonary involvement, which makes it almost imperceptible to the human eye in diagnosis. This case is shown in Fig. 7, which presents the CXR image of

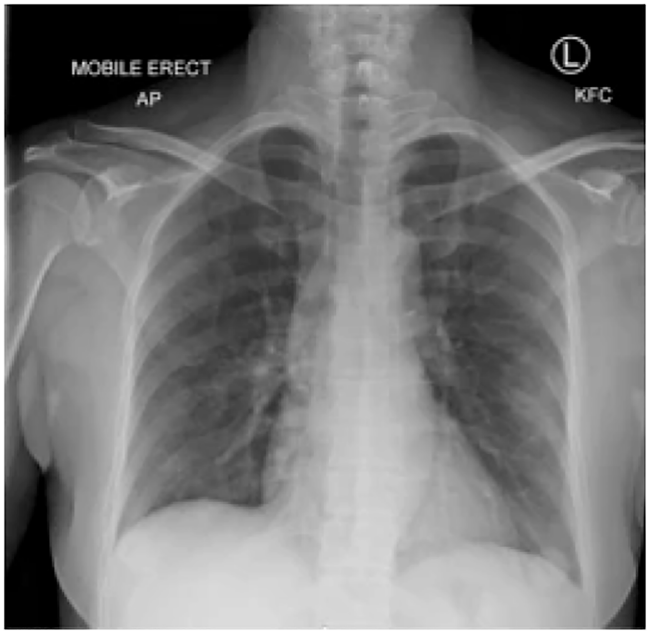

Fig. 6. Patient affected by COVID-19, wrongly classified by the method.

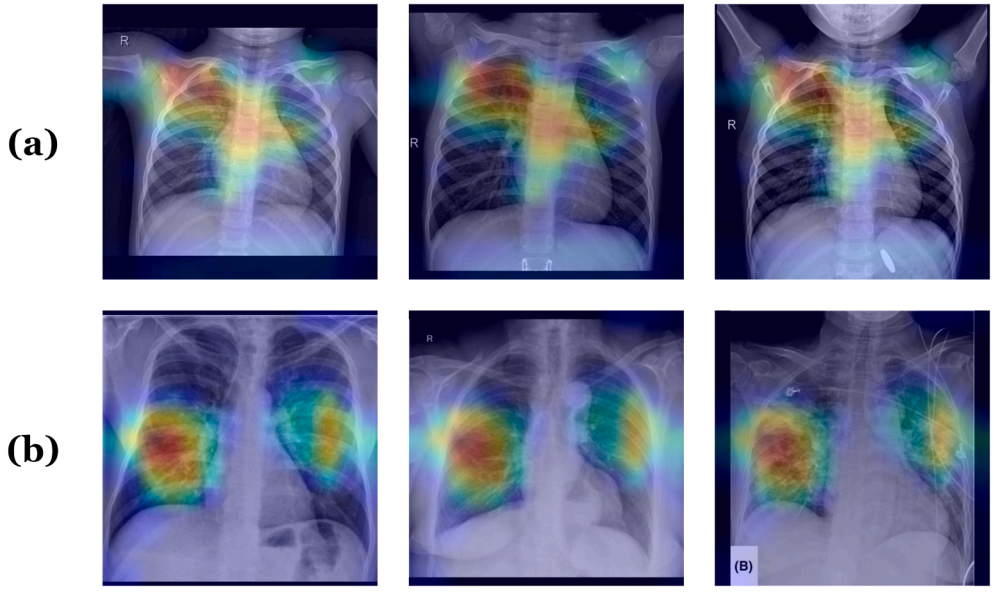

Fig. 5. Activation map for images of patients (a) under normal conditions and (b) affected by COVID-19..

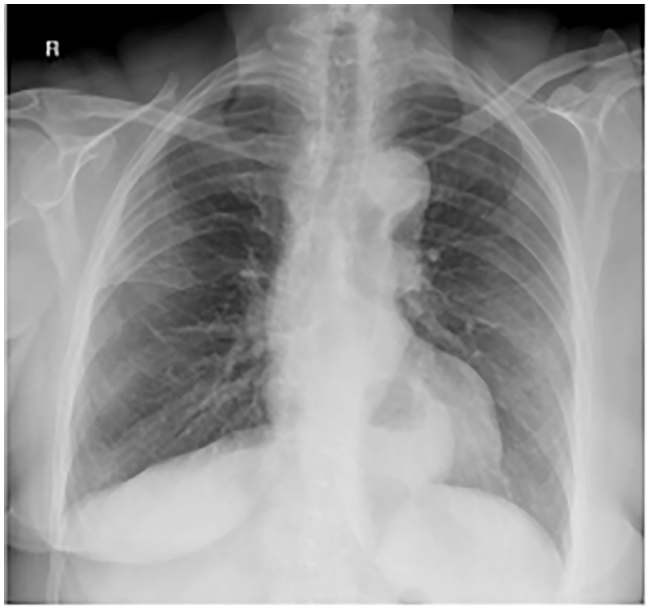

Fig. 7. Patient affected by COVID-19 but seemingly normal, classified correctly by the method.

a patient affected by COVID-19, though it presents strong characteristics of a patient under normal conditions. Even so, our method was able to correctly classify the patient as suffering from COVID-19. This demonstrates that our method can accurately detect even the most complex cases.

The third study is that of a patient with COVID-19; in this case, the model using deep features (from VGG19) and XGBoost + PSO gave the correct diagnosis. We can see in Fig. 8 that this patient exhibits large changes in their lungs, which indicates a certain degree of impairment; this caused the model to classify them as a patient with COVID-19.

4.1.2. Case study – Patients under normal conditions

To qualitatively evaluate the method, another two case studies of patients with normal CXR images are presented. In one case, the model misclassified the patient; in the other, the model's diagnosis was correct.

The first patient was healthy but classified as having COVID-19. The image is presented in Fig. 9; as we can see, although the image was classified as normal by the specialist, it presents pulmonary regions with high density; this confused the classifier, which considered the patient as suffering from COVID-19.

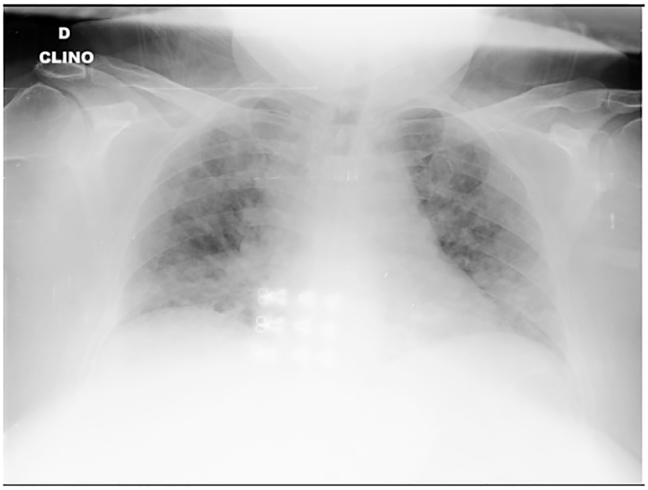

Fig. 8. Patient affected by COVID-19, classified correctly by the method.

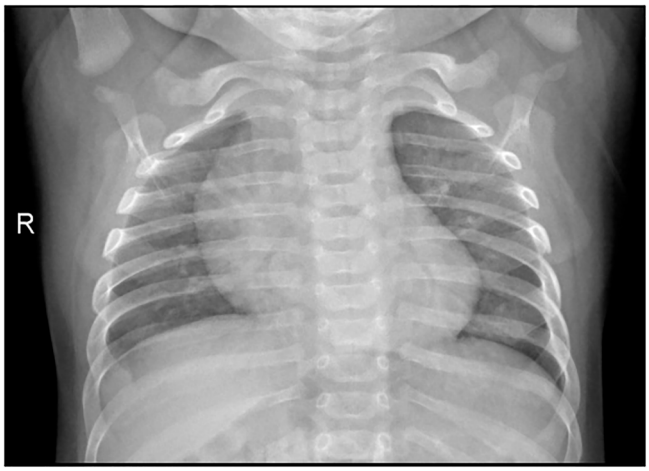

Fig. 9. Normal patient, classified incorrectly by the method.

The last case study presents a normal patient who was correctly classified by the method (Fig. 10). Once again, the method proved to be promising in assessing CXR images. This demonstrates the generalizability of the method, which achieved over 98% in all metrics.

4.2. Proposed method vs. other pneumonia

To validate our method can differentiate the CXR of patients with COVID-19 from other existing cases of pneumonia, we used CXR images of other pneumonia from the Guangzhou dataset. The purpose of the experiment is to show that the proposed model does not confuse other pneumonia with COVID-19. For this, we chose the deep feature extraction model that exhibited the optimum accuracy (i.e., VGG19).

The Guangzhou dataset has examples of two types of pneumonia, bacterial and viral. Thus, of the total of 3,875 images, 1,345 corresponded to CXR of patients with viral pneumonia and 2,530 of bacterial. In the next subsections, we present the results achieved using the proposed method.

4.2.1. Proposed method vs. bacterial pneumonia

First, we will validate our method on CXR images of patients with

Fig. 10. Normal patient, classified correctly by the method.

bacterial pneumonia. Given the total image set, these were submitted to the model and the classification result was collected. Table 10 shows the results achieved.

As we can see in the table, the model classified 87% as non-COVID-19 patients. It shows that the model can satisfactorily differentiate patients with bacterial pneumonia from COVID-19. Despite the model making some patients wrong, it is worth mentioning that the diagnosis decision is accompanied by a series of laboratory and clinical exams, and that this model only has the role of supporting the specialist. Furthermore, patterns of pulmonary involvement of pneumonia are similar to those of COVID-19, which causes a conflict in the correct classification. Fig. 11 presents activation maps for the images of patients under bacterial pneumonia conditions.

Figure shows both cases, those classified correctly for non-COVID-19 and those wrong classified for COVID-19, the activation map took into account the classification of the pulmonary regions. However, there is a high degree of patient impairment, and even so, the model is able to correct 87% of cases for non-COVID-19.

4.2.2. Proposed method vs. viral pneumonia

Analogous to the experiment done with bacterial pneumonia, in this, we will validate our method in CXR images of patients with viral pneumonia. Table 11 shows the results achieved.

We can see that the model proved to be effective in the task of differentiating patterns of viral pneumonia from that caused by COVID-19. Similar to bacterial, there is an error in the classification of some exams. It is worth mentioning that this tool is not intended to diagnose patients, but rather to serve as a screening and assistance tool for specialists. Also analogous to the previous experiment, we remember that there is great pulmonary involvement caused by other pneumonia, which generates conflict to the classifier. Fig. 12 presents activation maps for the images of patients under viral pneumonia conditions.

Once again, it is observed that the regions activated for classification of the model are regions with great pulmonary impairment. Even so, the model was efficient in 88% of the cases, in correctly correcting the class of non-COVID-19.

4.3. Comparison with related works

Though it is a recent problem, the pandemic caused by COVID-19 has prompted several researchers to develop a methods to help doctors and frontline professionals; many of these methods have been developed to analyze CXR images. A comparison of these works against the proposed method is presented in Table 12.

We can see that almost all use deep learning architectures in their approach. Deep learning techniques hold great promise, primarily because there is no explicit need for the extraction, selection, and classification of characteristics. However, training with this type of approach is costly, and many training sections are required to find the best parameters.

In Narin et al. (2020), the experiments used CXR image datasets containing 50 normal patients and 50 patients with COVID-19, both taken from public repositories. The experimental results show an accuracy of 98%. In El-Din Hemdan et al. (2020), a 90% accuracy was achieved by considering 50 CXR images, 25 of which were confirmed cases of COVID-19. In Zhang et al. (2020), 100 CXR images from patients with COVID-19 and 1341 from normal patients were used to validate the experiments; their method achieved a 96% accuracy. In our study, we used a larger dataset to validate our method; it contained 206 images of COVID-19 patients, and 1341 images of normal patients; in the best case, we achieved results exceeding 98% in accuracy, surpassing the studies described above.

Apostolopoulos and Mpesiana (2020) evaluated several deep architectures and transfer learning techniques. Using a VGG19 to detect COVID-19, they obtained results of 98.75% and 92.85% in accuracy and recall, respectively; their research used a dataset consisting of 224 images of patients with COVID-19 and 1204 images of patients with pneumonia or healthy lungs. Pereira et al. (2020) composed a database referred to as RYDLS-20; it contains 1144 CXR images of pneumonia caused by different pathologies, as well as images of healthy lungs. In their experiments, they achieved an 89% F1-score. Ozturk et al. (2020) proposed a CNN model based on DarkNet to detect COVID-19 cases. The model achieved an accuracy of 98.08% for a dataset composed of 125 COVID-19 CXR images, 500 pneumonia images, and 500 normal images. Our method can still outperform these.

CNNs automatically extract features from input data and perform classifications based on these extracted features, using a classifier such as softmax. The softmax classifier is a common option for CNNs; however, it is not mandatory and can be replaced by any classifier. One of these experiments was carried out in Sethy and Behera (2020), where the ResNet50 model was used in conjunction with an SVM classifier to detect patients with COVID-19. The model achieved an accuracy of 95.38% and an F1-score of 91.41%. In our experiments, we obtained superior results to all of these.

Finally, the method proposed in Rahimzadeh and Attar (2020) achieved an impressive 99.50% accuracy. However, only a 35.27% precision and 80.53% recall were achieved for detection of the COVID-19 class. This was because their method was more focused on selecting classes of normal patients/pneumonia, which is not our focus here. In our experiments, we obtained balanced results for the two classes (COVID-19 and normal), presenting results exceeding 98% for the metrics of accuracy, precision, recall, and F1-score.

Thus, in contrast to the methods described in the literature, our work proposes a method that exclusively uses deep features, which are extracted from the convolution and subsampling layers of deep learning networks. For the classification step, we propose the use of XGBoost, which does not require a large number of resources and has shown promising results in several studies regarding data classification. A technique based on the evolutionary algorithm (PSO) was proposed to identify the best parameters for XGBoost. Thus, we demonstrate the promise of our method, by presenting results comparable to those found in the literature; our method can be used as a tool to combat the pandemic caused by COVID-19.

4.4. Advances and limitations of the proposed method

Because it proposes a new and fully automated method for classifying patients into COVID-19 and normal groups, our method offers a series of merits and advances, of which we highlight the main ones as follows:

1. It offers an automated method, developed using two different, public databases (Section 2). The diversity of the databases simulates the clinical routine, and their publicity makes the method amenable to comparison.
2. Because the CXR image databases differ and are not standardized in either the data-acquisition or examination processes, the images present numerous patterns. Thus, our proposed method implements a crucial stage of image preprocessing; despite its simplicity, this is necessary for the successful execution of the entire method.
3. Using Deep Features, the method automatically extracts and selects characteristics. It is known that the process of choosing representative characteristics is not trivial when using the deep features

Table 10
Proposed Method vs. Bacterial Pneumonia.

	Quantity	Percentage(%)
Wrongly classified as COVID-19	328	13
Correctly classified as non-COVID-19	2,202	87
Total	2,530	100

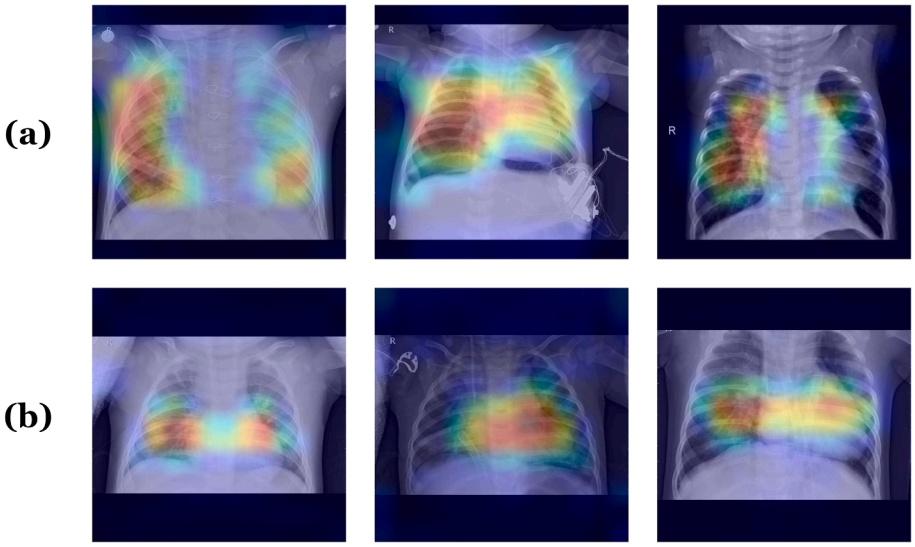

Fig. 11. Activation map for images of patients (a) correctly classified as non-COVID-19 and (b) wrongly classified as COVID-19..

Table 11
Proposed Method vs. Viral Pneumonia.

	Quantity	Percentage(%)
Wrongly classified as COVID-19	168	12
Correctly classified as non-COVID-19	1,177	88
Total	1,345	100

approach; hence, these characteristics are extracted and selected implicitly.

4. To the best of our knowledge, this is the first method to use Deep Features combined with an XGBoost classifier to diagnose patients into COVID-19 and healthy groups.
5. The addition of the robust tree-based classifier XGBoost makes the method more effective, because this classifier offers considerable generalizability; moreover, it does not require large quantities of hardware resources, which is one of the biggest problems facing deep learning-based methods.
6. Automated methods have been studied by many researchers; however, these methods always encounter a parameterization barrier. This is no different for the XGBoost classifier, which features a range of parameters. Thus, we proposed a PSO to automatically optimize these parameters and bypass the parameter selection step.
7. All of these steps increase the method's utility. The present study obtained results superior to those found in the literature and proposes an innovative new method for the classification of patients with COVID-19.
8. The proposed method, using deep features (from VGG19) and XGBoost + PSO, achieved a maximum average accuracy of 98.71%, an average precision of 98.89%, an average recall of 99.63%, and an average F1-score of 99.25%. These results indicate the method's potential.

However, as with any computational method, it has some limitations. We highlight these as follows:

1. Our method did not propose a new deep architecture. We used the existing architectures and demonstrated their effectiveness for capturing deep features. It is believed that developing a new architecture will further improve results.

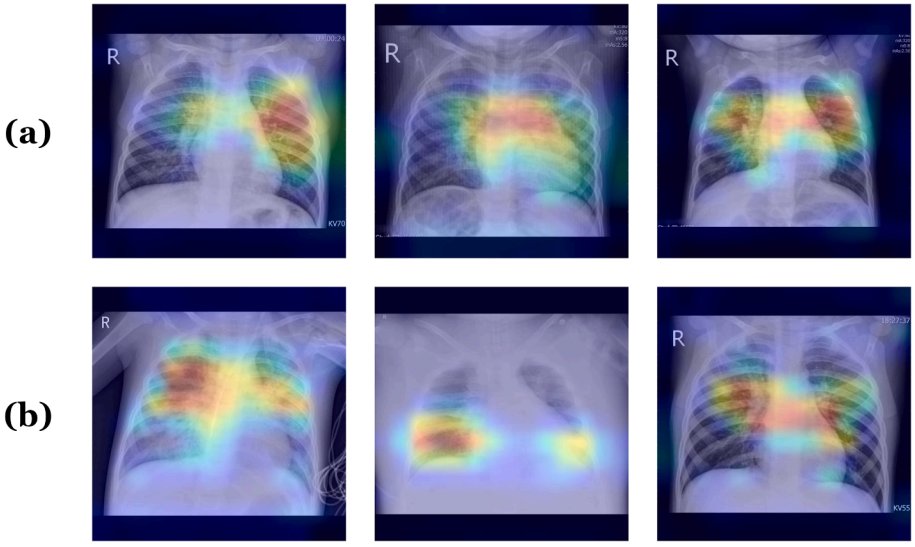

Fig. 12. Activation map for images of patients (a) correctly classified as non-COVID-19 and (b) wrongly classified as COVID-19..

Table 12
Comparison with related works.

Work	Deep Architecture	Acc (%)	Prec (%)	Rec (%)	F1 (%)
Narin et al. (2020)	ResNet50	98	-	-	-
Apostolopoulos and Mpesiana (2020)	VGG19	98.75	-	92.85	-
El-Din Hemdan et al. (2020)	VGG19 and DenseNet201	90	-	-	-
Zhang et al. (2020)	ResNet	96	-	-	-
Pereira et al. (2020)	Inception-v3 and texture descriptors	-	-	-	89
Ozturk et al. (2020)	DarkCovidNet	98.08c	-	-	-
Sethy and Behera (2020)	ResNet50 and SVM	95.38	-	-	91.41
Rahimzadeh and Attar (2020)	Xception and ResNet50V2	99.50	35.27	80.53	-
Proposed Method	VGG19	98.71	98.89	99.63	99.25
	Inception-v3	97.48	98.29	98.81	98.54
	ResNet50	98.51	98.80	99.48	99.14

2. The present method indicates whether a CXR image is considered to exhibit COVID-19 symptoms or not. Adding a segmentation stage to identify the regions affected by COVID-19 could further help front-line professionals.

Consequently, we believe that the method can play a major role in classifying COVID-19 patients from CXR images. It is known that the most-used mechanism for medical imaging in developing countries is the X-ray. Proposing an automated system capable of generating good outcomes is of crucial significance in helping health practitioners tackle this pandemic.

5. Conclusion

The pandemic caused by COVID-19 has global impacts. Various methods designed to assist frontline professionals have been developed. The present work proposed a fully automated method, using Deep Features to extract feature information from CXR images; furthermore, it proposed PSO-optimized XGBoost, to classify these images as depicting COVID-19 or healthy lungs.

Despite COVID-19 being a new problem, several works have already been developed along these lines, and the use of public databases is necessary for comparing the proposed methods. Thus, this study used two public databases, one containing COVID-19 patients and the other containing normal patients. Because we used different and non-standardized databases, a preprocessing step was used to standardize the examination results.

Then, a deep features extraction step was used to extract the features of the images automatically. Finally, we proposed to use the XGBoost classifier, which offers a high performance in terms of speed, scalability, memory consumption, and hardware resources. To further increase the promise of the method, an optimization step (using PSO) was performed for the parameters of this classifier.

The present work, using deep features of VGG19 and XGBoost + PSO, achieved an average accuracy of 98.71%, an average precision of 98.89%, an average recall of 99.63%, and an average F1-score of 99.25%. Our results, compared with those reported the literature, are prominent. Thus, we believe that we have proposed a promising, innovative, and capable method to be used by professionals for classifying the CXR images of patients affected by COVID-19.

As a future research direction, we highlight the use of other architectures or the development of new, function-specific ones. Our method did not propose a new architecture for extracting deep features. It used existing ones and showed that they are effective for capturing features when used in conjunction with XGBoost. It is believed that developing new architectures will further improve the results.

Moreover, the present method indicates whether a CXR image is considered to show COVID-19 symptoms. Adding a segmentation step to identify the regions affected by COVID-19 can further assist front-line professionals. Another idea is to extend the method, using it to classify other pathologies with pulmonary involvement. We believe that these changes can add value to the proposed method.

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.

Acknowledgements

The authors acknowledge the help from Applied Computing Group (NCA) from the Federal University of Maranhão, the Federal Institute of Maranhão (IFMA), Tecgraf from the Pontifical Catholic University of Rio de Janeiro (PUC-Rio), and The National Council of Federal Institutions of Professional, Scientific and Technological Education (CONIF). We thank Dr. Joseph Paul Cohen from the University of Montreal for providing a dataset of COVID-19 images for the research community.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., et al., 2015. Tensorflow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Apostolopoulos, I. D., & Mpesiana, T. A. (2020). Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. *Physical and Engineering Sciences in Medicine, 1*.

Boccia, S., Ricciardi, W., Ioannidis, J.P., 2020. What other countries can learn from italy during the covid-19 pandemic. JAMA internal medicine.

Breiman, L. (2001). *Random forests. Machine learning, 45*, 5–32.

Carvalho, E. D., Antônio Filho, O., Silva, R. R., Araújo, F. H., Diniz, J. O., Silva, A. C., Paiva, A. C., & Gattass, M. (2020). Breast cancer diagnosis from histopathological images using textural features and cbir. *Artificial Intelligence in Medicine, 105*, Article 101845.

Chapman, W. W., Fizman, M., Chapman, B. E., & Haug, P. J. (2001). A comparison of classification algorithms to automatically identify chest x-ray reports that support pneumonia. *Journal of biomedical informatics, 34*, 4–14.

Chen, T., He, T., Benesty, M., 2018. Xgboost documentation.

Chen, T., He, T., Benesty, M., Khotilovich, V., & Tang, Y. (2015). Xgboost: extreme gradient boosting. *R package version*, (4-2), 1–4.

Chollet, F., et al., 2015. Keras. URL:https://keras.io.

Cohen, J.P., Morrison, P., Dao, L., 2020. Covid-19 image data collection. arXiv 2003.11597 URL:https://github.com/ieee8023/covid-chestxray-dataset.

da Cruz, L. B., Araújo, J. D. L., Ferreira, J. L., Diniz, J. O. B., Silva, A. C., de Almeida, J. D. S., de Paiva, A. C., & Gattass, M. (2020). Kidney segmentation from computed tomography images using deep neural network. *Computers in Biology and Medicine, 103906*. https://doi.org/10.1016/j.compbiomed.2020.103906

Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition Ieee* (pp. 248–255).

Diniz, J. O. B., Diniz, P. H. B., Valente, T. L. A., Silva, A. C., & Paiva, A. C. (2019). Spinal cord detection in planning ct for radiotherapy through adaptive template matching, imslic and convolutional neural networks. *Computer Methods and Programs in Biomedicine, 170*, 53–67.

Diniz, J. O. B., Diniz, P. H. B., Valente, T. L. A., Silva, A. C., de Paiva, A. C., & Gattass, M. (2018). Detection of mass regions in mammograms by bilateral analysis adapted to breast density using similarity indexes and convolutional neural networks. *Computer Methods and Programs in Biomedicine, 156*, 191–207.

Diniz, J. O. B., Ferreira, J. L., Diniz, P. H. B., Silva, A. C., & de Paiva, A. C. (2020). Esophagus segmentation from planning ct images using an atlas-based deep learning approach. *Computer Methods and Programs in Biomedicine, 197*, Article 105685.

Diniz, P. H. B., Valente, T. L. A., Diniz, J. O. B., Silva, A. C., Gattass, M., Ventura, N., Muniz, B. C., & Gasparetto, E. L. (2018). Detection of white matter lesion regions in mri using slic0 and convolutional neural network. *Computer Methods and Programs in Biomedicine, 167*, 49–63.

Duda, R. (1973). Pattern classification and scene analysis. Wiley-Interscience. *Publication, 512*.

Eberhart, R., & Kennedy, J. (1995). Particle swarm optimization, in. In *Proceedings of the IEEE international conference on neural networks Citeseer* (pp. 1942–1948).

El-Din Hemdan, E., Shouman, M.A., Karar, M.E., 2020. Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray images. arXiv, arXiv–2003.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics, 29*, 1189–1232.

Gonzalez, R., & Woods, R. (2008). *Digital image processing*. Pearson: Prentice Hall.

H Mohamed, E., H El-Behaidy, W., Khoriba, G., Li, J., 2020. Improved white blood cells classification based on pre-trained deep learning models. Journal of Communications Software and Systems 16, 37–45.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition, in. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Kennedy, J. (2010). Particle swarm optimization. *Encyclopedia of machine learning*, 760–766.

Kermany, D., Zhang, K., Goldbaum, M., 2018. Labeled optical coherence tomography (oct) and chest x-ray images for classification. Mendeley data 2.

Kleinbaum, D. G., Dietz, K., Gail, M., Klein, M., & Klein, M. (2002). *Logistic regression*. Springer.

Kroft, L. J., van der Velden, L., Girón, I. H., Roelofs, J. J., de Roos, A., & Geleijns, J. (2019). Added value of ultra–low-dose computed tomography, dose equivalent to chest x-ray radiography, for diagnosing chest pathology. *Journal of Thoracic Imaging, 34*, 179.

Le, L. T., Nguyen, H., Zhou, J., Dou, J., Moayedi, H., et al. (2019). Estimating the heating load of buildings for smart city planning using a novel artificial intelligence technique pso-xgboost. *Applied Sciences, 9*, 2714.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*, 436–444. https://doi.org/10.1038/nature14539

Mirjalili, S. (2019). Genetic algorithm. *Evolutionary algorithms and neural networks. Springer*, 43–55.

Narin, A., Kaya, C., Pamuk, Z., 2020. Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. arXiv preprint arXiv:2003.10849.

Organization, W. H., et al. (2020). *Coronavirus disease (COVID-19) technical guidance: The Unity Studies: Early Investigations Protocols*. Technical Report. World Health Organization.

Ozturk, T., Talo, M., Yildirim, E. A., Baloglu, U. B., Yildirim, O., & Acharya, U. R. (2020). Automated detection of covid-19 cases using deep neural networks with x-ray images. *Computers in Biology and Medicine, 121*, Article 103792. https://doi.org/10.1016/j.compbiomed.2020.103792

Pereira, R. M., Bertolini, D., Teixeira, L. O., Silla, C. N., & Costa, Y. M. (2020). Covid-19 identification in chest x-ray images on flat and hierarchical classification scenarios. *Computer Methods and Programs in Biomedicine, 194*, Article 105532. https://doi.org/10.1016/j.cmpb.2020.105532

Petrosillo, N., Viceconte, G., Ergonul, O., Ippolito, G., & Petersen, E. (2020). Covid-19, sars and mers: are they closely related? *Clinical Microbiology and Infection*.

Rahimzadeh, M., & Attar, A. (2020). 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. *Informatics in Medicine Unlocked, 19*, Article 100360. https://doi.org/10.1016/j.imu.2020.100360

Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C., Shpanskaya, K., et al., 2017. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. arXiv preprint arXiv:1711.05225.

Rothan, H. A., & Byrareddy, S. N. (2020). The epidemiology and pathogenesis of coronavirus disease (covid-19) outbreak. *Journal of autoimmunity, 102433*.

Sasaki, Y., & Fellow, R. (2007). *The truth of the f-measure*. manchester: Mib-school of computer science. University of Manchester.

Sethy, P. K., & Behera, S. K. (2020). Detection of coronavirus disease (covid-19) based on deep features. *Preprints, 2020030300*, 2020.

da Silva, G. L. F., Valente, T. L. A., Silva, A. C., de Paiva, A. C., & Gattass, M. (2018). Convolutional neural network-based pso for lung nodule false positive reduction on ct images. *Computer Methods and Programs in Biomedicine, 162*, 109–118.

Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Song, R., Chen, S., Deng, B., & Li, L. (2016). extreme gradient boosting for identifying individual users across different digital devices. In B. Cui, N. Zhang, J. Xu, X. Lian, & D. Liu (Eds.), *Web-Age Information Management* (pp. 43–54). Cham: Springer International Publishing.

Souza, J. C., Diniz, J. O. B., Ferreira, J. L., da Silva, G. L. F., Silva, A. C., & de Paiva, A. C. (2019). An automatic method for lung segmentation and reconstruction in chest x-ray using deep neural networks. *Computer Methods and Programs in Biomedicine, 177*, 285–296.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2818–2826).

Teodorovic, D., Lucic, P., Markovic, G., & Dell'Orco, M. (2006). Bee colony optimization: principles and applications. In *2006 8th Seminar on Neural Network Applications in Electrical Engineering* (pp. 151–156).

Zhang, H. W., Yu, J., Xu, H. J., Lei, Y., Pu, Z. H., Dai, W. C., Lin, F., Wang, Y. L., Wu, X. L., Liu, L. H., et al. (2020). Corona virus international public health emergencies: implications for radiology management. *Academic radiology*.

Zhang, J., Xie, Y., Li, Y., Shen, C., Xia, Y., 2020. Covid-19 screening on chest x-ray images using deep learning based anomaly detection. arXiv preprint arXiv:2003.12338.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., & Torralba, A. (2016). Learning deep features for discriminative localization. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2921–2929).

Zhu, N., Zhang, D., Wang, W., Li, X., Yang, B., Song, J., Zhao, X., Huang, B., Shi, W., Lu, R., et al. (2020). China novel coronavirus investigating and research team. a novel coronavirus from patients with pneumonia in china, 2019. *The New England Journal of Medicine, 382*, 727–733.