Handling class imbalance in COVID-19 chest X-ray images classification: Using SMOTE and weighted loss

Ekram Chamseddine [a], Nesrine Mansouri [b,*], Makram Soui [c], Mourad Abed [d]

[a] *SIME Laboratory, National School of Engineers of Tunis, University of Tunis, Tunisia*
[b] *Artificial Intelligence Research Unit, National School of Computer Sciences, University of Manouba, Tunisia*
[c] *College of Computing and Informatics, Saudi Electronic University, Saudi Arabia*
[d] *LAMIH, Polytechnic University of Hauts-de-France, France*

ARTICLE INFO

ABSTRACT

Healthcare systems worldwide have been struggling since the beginning of the COVID-19 pandemic. The early diagnosis of this unprecedented infection has become their ultimate objective. Detecting positive patients from chest X-ray images is a quick and efficient solution for overloaded hospitals. Many studies based on deep learning (DL) techniques have shown high performance in classifying COVID-19 chest X-ray images. However, most of these studies suffer from a class imbalance problem mainly due to the limited number of COVID-19 samples. Such a problem may significantly reduce the efficiency of DL classifiers. In this work, we aim to build an accurate model that assists clinicians in the early diagnosis of COVID-19 using balanced data. To this end, we trained six state-of-the-art convolutional neural networks (CNNs) via transfer learning (TL) on three different COVID-19 datasets. The models were developed to perform a multi-classification task that distinguishes between COVID-19, normal, and viral pneumonia cases. To address the class imbalance issue, we first investigated the Weighted Categorical Loss (WCL) and then the Synthetic Minority Oversampling Technique (SMOTE) on each dataset separately. After a comparative study of the obtained results, we selected the model that achieved high classification results in terms of accuracy, sensitivity, specificity, precision, F1 score, and AUC compared to other recent works. DenseNet201 and VGG-19 claimed the best scores. With an accuracy of 98.87%, an F1_Score of 98.21%, a sensitivity of 98.86%, a specificity of 99.43%, a precision of 100%, and an AUC of 99.15%, the WCL combined with CheXNet outperformed the other examined models.

1. Introduction

Nowadays, coronavirus is the most dreadful disease worldwide. This pandemic started in Wuhan, China, in December 2019. Since then, it has been rapidly spreading all over the globe. Recently, the World Health Organization (WHO) reported that there had been 119,960,700 confirmed coronavirus cases worldwide, including 2,656,822 deaths in less than a year [1]. It is shown in Fig. 1 that these numbers are exponentially growing every day, making the world's first purpose is to reduce the number of contaminated individuals. Therefore, many strategies have been taken worldwide, including isolating infected patients, curfew, and lockdown. The International Committee for the Taxonomy of Viruses (ICTV) named the virus responsible for this disease SARS-CoV-2 [2]. COVID-19 is the name of the illness caused by

SARS-CoV-2 [3]. This virus possesses powerful pathogenicity and transmissibility. It mainly attacks the human respiratory system and brings on what clinicians call a respiratory tract infection [4]. For the most part, coronavirus spreads through person-to-person contact. The contamination starts with droplets from an infected person (cough, sneeze, or breath) ejected into the air or a surface that a healthy individual could breathe or touch, then touches his mouth, nose, or eyes [5].

Thus, the virus can reach the respiratory system. The spectrum of COVID-19 ranges from asymptomatic to severe, often fatal infection. Severe infections occur more frequently in males and patients with chronic diseases such as diabetes, hypertension, heart disease, and immunocompromised states [6].

This pandemic and the resulting lockdown have had a detrimental impact on several vital domains such as education, entertainment, the economy, and especially the healthcare sector, which is at the epicenter of the COVID-19 pandemic [4]. Indeed, healthcare systems across the globe have undergone unprecedented crises. It is mainly due to unpredictable and massive
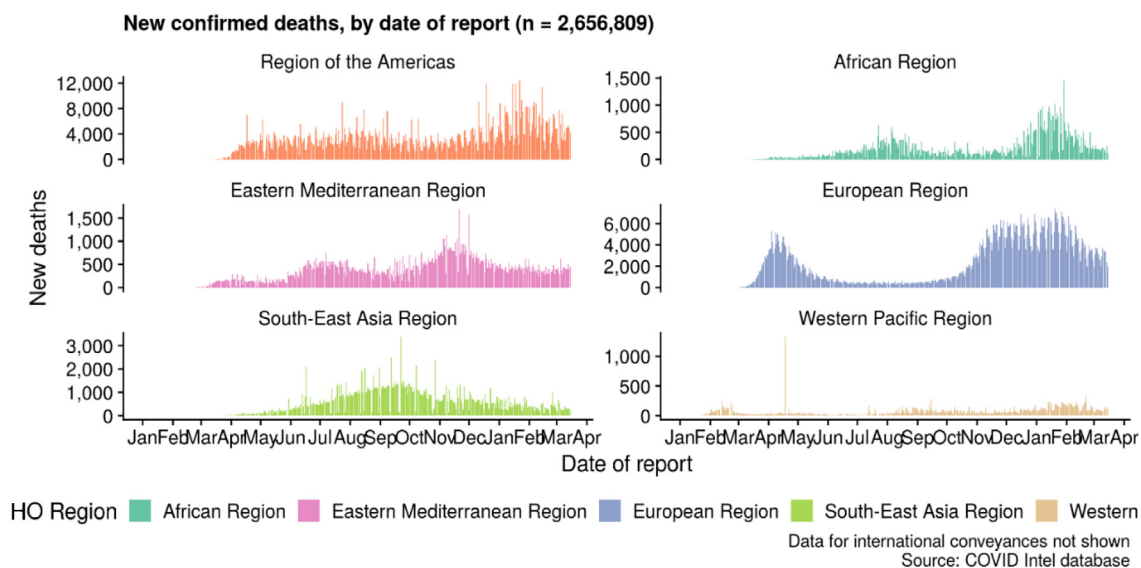
Fig. 1. New confirmed deaths worldwide were caused by COVID-19 in March 2021 [1].

health challenges, including hospitals' overloading with COVID-19 patients. Therefore, the urgent mobilization of resources was required. Unfortunately, these systems were not designed to manage such a critical situation. The high medical expenses, the deficiency of protective equipment, and the shortages of ICU beds and ventilators highlighted the defects in the delivery of patient care. Furthermore, medical staff contamination risk is one of the significant vulnerabilities of healthcare systems worldwide. Given that most healthcare workers have to work on-site, many procedures have been employed to protect them, including the early deployment of viral testing for asymptomatic and/or frontline healthcare staff [7]. The early detection and isolation of infected people have become the ultimate solutions to the issues above.

The diagnosis of COVID-19 disease is often performed using a Reverse Transcription Polymerase chain reaction (RT-PCR) approach that consists of taking respiratory specimens for testing [8]. Despite the widespread use of the RT-PCR test for the clinical screening of COVID-19 patients, this technique is reported as complicated, time-consuming, and requires human intervention with only 63% accuracy [8]. Furthermore, the PCR machine must be installed in a specialized biosafety lab that may cost 15 thousand to 90 thousand USD and the RT-PCR kit costs between 120 and 130 USD [9]. Even high-income countries consider this tool too expensive [10]. Moreover, the shortage of these testing kits is causing a significant delay in detecting contaminated patients, making the situation more critical.

Medical imaging techniques such as Computed Tomography (CT) and Chest X-ray (CXR) are also used for COVID-19 diagnosis [11]. Indeed, CT has proven to be more sensitive, significantly faster, and cheaper than RTPCR in diagnosing COVID19 [12,13]. However, this imaging modality is not recommended in regions of low disease spreading because of the significant rate of false positives [14]. Moreover, CT is often available only in large medical centers and is still in short supply in many countries. In addition, performing a CT scan for a patient with mild symptoms is unnecessary since patches can be detected even if the patient is asymptomatic [15]. In this case, CXR would be the optimal method for diagnosing COVID-19. Indeed, plain film chest x-rays are omnipresent globally, cheaper and with radiation doses 30 to 70 times lower than CT scans [16]. Thus, they are commonly used as the first screening modality for COVID-19. The American College of Radiology confirms that, for COVID-19, CXR is portable and easy to clean compared to CT [17]. Therefore,

although chest CXR has lower sensitivity than chest CT [18], it is the typical imaging modality employed for the early screening of suspected cases [19]. Nevertheless, due to the similarity between the CXR imaging features of COVID-19 and other viral pneumonia, CXR represents a wide inter-observer variability. In the current circumstances, radiologists are under increasing work pressure, making their tasks more and more difficult.

Artificial Intelligence (AI) has recently shown the potential to improve medical imaging abilities, including accurate analysis, higher automation, and enhanced productivity [20]. Using AI, the research community has made a great effort to develop computer-aided diagnosis (CAD) systems to assist clinicians in their mission. Indeed, the second opinion of the CAD system can help increase the diagnosis's accuracy, reduce inter and intra-observer variability, and avoid unnecessary procedures. Developing a CAD system can be challenging depending on the study stage [21]. Most research in this field has followed the traditional machine learning pipeline (ML). This pipeline requires a high level of application-specific expertise, especially for extracting and selecting the appropriate features from the images. Since 2012, Deep Learning (DL) based on Convolution Neural Networks (CNNs) has become an outstanding technique in several computer vision problems [22,23]. The medical imaging community has adopted this technique for diagnosing or segmenting organs and forms in medical images [23,24]. Many researchers investigating deep learning for medical diagnosis have achieved "near" human expert diagnosis performance [25].

In contrast with machine learning, CNNs do not need feature engineering. They automatically learn an appropriate image representation. A CNN has a unique architecture biologically inspired by the Artificial Neural Network (ANN). It has layers arranged in a hierarchical structure and is specially designed for learning visual features [26]. Many CNN-based systems are developed to automatically diagnose COVID-19 using both CT and X-ray imaging modalities [27]. Some are designed based on a pre-trained model with transfer learning [28,29], and a few others are built using customized networks [30–32]. Most existing works suffer from a class imbalance, a common problem in deep learning-based classifiers, particularly in medical diagnostics [33,34]. Class imbalance occurs when some classes have significantly more samples in the training dataset than others. It has been proven that class imbalance may have a negative impact on training a CNN model. Indeed, it can influence the convergence during the training step

as well as the model generalization on the test set. It results in a model with poor predictive performance, specifically for the minority class [35]. There are two main categories of class imbalance solutions [36]. The first category is data-level solutions in which a change is made in the class distribution of the dataset itself, either by oversampling or undersampling the training dataset. The oversampling techniques are widely employed in deep learning and have been shown to be reliable. Random minority oversampling is the primary form of oversampling. It merely replicates randomly chosen samples from minority classes. It has been demonstrated to be efficient, although it can result in overfitting problems [35,37]. SMOTE [37] is a more sophisticated oversampling technique that seeks to resolve this problem. It generates new synthetic samples by interpolating nearby data. The second category is classifier-level solutions. These solutions operate on the classification algorithm (model) while keeping the dataset unchanged. For example, some methods can alter the previous class probabilities [38]. Others add weights to the misclassification of samples from different classes [38]. Adding sample weights to the loss function is a straightforward technique to handle class imbalance. The idea is to weigh the loss calculated for different samples differently based on whether they are associated with the minority or the majority classes. A higher weight is assigned to the loss computed by the samples of minor classes [39]. Solutions from both categories can be combined to address the class imbalance problem [35]. Recently, Generative Adversarial Networks(GANs) [40–42] have been demonstrated to be a powerful technique to rebalance datasets [43] by generating artificial samples. However, GANs are fairly challenging to train and have high computational resources. The second solution involves altering the classification algorithm (model) while maintaining the dataset. Only a few studies on the detection of COVID-19 from X-ray images addressed the issue of class imbalance [40–42].

In this work, our main contributions are as follows:

- Our objective is to develop an accurate deep learning model trained on a balanced dataset that assists radiologists in the early diagnosis of COVID-19 cases.
- To this end, we trained six popular CNNs, including Dense-Net201, CheXNet, MobileNetV2, ResNet152, VGG19, and Xception, via transfer learning (TL) on three different COVID-19 datasets. The models were developed to perform a multi-classification task that distinguishes between COVID-19, normal (no infection), and viral pneumonia cases.
- To address the class imbalance issue, we first investigated the Weighted Categorical Loss (WCL) [40] and then the Synthetic Minority Oversampling Technique (SMOTE) [37] on each dataset separately.
- After testing the developed models, we studied the outcomes of different evaluation criteria. This step aims to find the best model for the early detection of COVID-19 from CXR images.
- After conducting an experimental comparison with other works, we noticed that the use of WCL and SMOTE to balance the datasets provided models with prominent results compared to the models that were trained with imbalanced data.

The rest of this paper is organized as follows. Section 2 reviews the state-of-the-art techniques employed for COVID-19 diagnosis using CXR images. Section 3 presents the methodology of this study. Section 4 introduces the experimental setup. An ablation study is provided in Section 5. Section 6 reports our results and discussion. Section 7 presents the limitations of our work. Finally, Section 8 summarizes our contribution and presents our future work. The source code is publicly released for research purposes.[1]

---

[1] https://github.com/EkramCh/COVID19_Detection_-SMOTE_WCL.

2. Related works

Several deep learning-based COVID-19 diagnosis systems have been developed based on data collected from CXR imaging samples. Some are built based on customized deep learning models. Others used trained deep learning models via transfer learning.

2.1. Customized deep learning models

Wang et al. [44] developed an architecture called COVID-Net. This work was one of the most popular approaches used to diagnose COVID-19. The primary characteristic of this study is the use of the COVIDx dataset, which contains around 13,975 CXR images. However, it suffers from an imbalance in terms of the number of images in each class. Oh et al. [30] succeeded in developing a CNN-based system with a limited number of trainable parameters to categorize chest X-ray images into three different classes, including COVID-19, non-COVID-19, and normal. The proposed model presented better sensitivity than COVID-Net [44] but not better accuracy. Li et al. [45] used the discriminative cost-sensitive learning (DCSL) technique to screen COVID-19 automatically. This proposed approach was designed by combining fine-grained classification and cost-sensitive learning. Although it provided a good result, the dataset used in this work contained only 239 COVID-19 images and 1000 samples for each of the two other classes. Khobahi et al. [46] aimed to detect COVID-19 cases using a semi-supervised DL system built on auto-encoders called CoroNet. In order to train this model, the authors merged three open-access datasets to provide 18,529 images of different classes. However, only 99 images were used for the COVID-19 class, while 9579 were of viral pneumonia and 8851 samples were presented as normal cases.

2.2. Transfer learning models for detecting COVID-from chest X-ray images

In order to automatically detect COVID-19 from chest X-ray images, Chowdhury et al. [47] have explored different CNN models via transfer learning. The authors used a customized dataset that contains 423 COVID-19, 1485 viral pneumonia, and 1579 normal chest X-ray images. A binary classification scheme (normal and COVID-19 pneumonia) and a multi-class classification scheme (normal, viral, and COVID-19 pneumonia) are investigated. The final evaluation showed that DenseNet201 outperforms other deep CNN networks by providing better accuracy for the second scheme. However, addressing class imbalance, effective fine-tuning, and validation of the models have not been explored. Apostolopoulos and Tzani [48] published a transfer learning method. They investigated five deep learning models to classify a dataset of 1427 CXR images divided as follows: 224 COVID-19, 700 Bacterial Pneumonia, and 504 normal images. The proposed model achieved promising results. However, these results are based on a small dataset. Bassi and Attux [49] aimed to develop a CNN-based classifier for COVID-19 detection from CXR images. The classifier was built using the DenseNet architecture. First, it was pre-trained on ImageNet, then on the ChestX-ray14 dataset, and finally, on a customized COVID-19 dataset. The proposed model classifies X-ray images as COVID-19, viral pneumonia, and normal. The provided results demonstrate that CNN-based CXR image analysis is an accurate and costless approach for diagnosing the coronavirus. Yet, the data imbalance between the classes has not been handled. Luz et al. [50] used a dataset of 183 COVID-19, 16,132 normal images, and 14,348 pneumonia images to detect COVID-19 from chest X-ray images. The idea is to create a deep learning method based

on EfficientNet. Their experiment provided good results. However, larger and more heterogeneous datasets are still needed to validate their method. Punn and Agarwal [51] employed a random oversampling technique, a weighted class loss function, and transfer learning in different deep learning architectures. The authors performed binary classification (as normal and COVID-19 cases) and multi-class classification (as COVID-19, pneumonia, and normal cases) of posteroanterior chest X-ray images. The experimental results for each scheme were promising; however, NASNetLarge displayed better scores than other architectures. Hemdan et al. [52] developed a deep learning framework called COVIDX-Net using the Covid Chest x-ray dataset [53]. This framework aims to detect COVID-19 from CXR images automatically. Seven different deep CNN architectures were explored, including for performance evaluation. The VGG19 and DenseNet201 models outperform other deep neural classifiers in terms of accuracy. El Asnaoui and Chawki [29] conducted a comparative study between different deep learning models to solve the problem of CXR image classification for COVID-19 detection. The dataset contained 2780 images of bacterial pneumonia, 1493 of coronavirus, 231 of Covid19, and 1583 normal. After evaluation, the authors reported that inceptionResnetV2 and DenseNet201 had better performance than the other architectures. Loey et al. [40] explored three pre-trained models and a Generative Adversarial Network (GAN) to detect COVID-19 from the CXR image. The GAN was used to increase the limited number of COVID-19 samples. Three schemes have been investigated; the first included four classes from the dataset, the second included three classes, and the final included two classes. Although the authors tried to generate balanced classes, the adopted dataset is still small for a DL approach. Bhattacharyya et al. [54] developed a new method for detecting COVID-19 in X-ray images. The proposed method consists of three steps: the segmentation of x-ray images using C-GAN in order to extract lung images; then, the segmented images feed into a novel pipeline that combines key point extraction methods and trained deep neural networks (DNN) to extract the relevant features. The final evaluation showed that the VGG19 models linked with the binary robust invariant scalable key points (BRISK) obtained a promising result that can be used efficiently to diagnose infected patients. Demir [55] proposed a new approach based on a deep LSTM called DeepCoroNet to distinguish infected COVID-19 patients from x-ray images. Indeed, this method is different from other techniques, such as transfer learning and deep feature extraction, because it is learned from scratch. To increase the performance of the final model, the author applied the Sobel gradient as well as the marker-controlled watershed segmentation operations during the preprocessing stage. The proposed approach was a helpful tool for radiologists and experts in detecting, determining quantity, and controlling the rise of COVID-19 cases. Jain et al. [41] conducted a comparative study using three DL architectures to select the appropriate model for COVID-19 detection using X-ray images. A total of 5467 chest x-ray scan samples were used to train the models, while 965 were used for validation. The outcomes showed that the Xception model achieved the highest accuracy in detecting COVID-19 patients compared to other studied models. Bargshady et al. [42] implemented a new approach called Inception-CycleGAN that can detect COVID-19 infected X-ray and CT Chest Images. This work aims to augment the number of training samples by applying the semi-supervised CycleGAN (SSA-CycleGAN) technique. Then, the Inception V3 transfer learning model is developed and fine-tuned to train the model for detecting COVID-19.

Despite their success, most of these works suffer from data imbalance issues, which significantly impact classification results. Thus, the main goal of this work is to propose an accurate yet efficient model of COVID-19 screening using chest X-ray images trained on a balanced dataset.

3. Methodology

3.1. Deep learning

Deep learning is increasingly becoming a vital tool in artificial intelligence applications [26]. Indeed, convolution neural networks produced excellent results in different areas, such as speech recognition, natural language processing, and computer vision [56]. Image classification is one of the tasks in which CNN's excel [57]. It aims to label distinct image according to a set of potential classes. From a deep learning aspect, the image classification challenge can be handled through transfer learning [58], especially for medical image classification, which is an essential instrument for disease diagnosis in the healthcare sector. Several up-to-date results in medical image classification have been established on transfer learning solutions [59–61]. In fact, with the advent of machine learning, the development of CAD systems has become one of the most explored research directions. However, the acquisition of medical images requires the use of specific medical tools. Besides, only experienced clinicians are often employed for their labeling. Thus, obtaining enough data for CNN training is often tough and expensive. In this case, transfer learning could be the ultimate solution for medical imaging analysis.

3.1.1. Convolution neural network

CNNs were revealed to excel in various computer vision challenges [62]. Two of the main aspects driving the fame of CNN over the last few years are its automatic and outstanding performance. Fig. 2 illustrates this particular neural network. A CNN has convolution and classifier parts. The convolution part is a pile of convolutional and pooling layers that aim to extract features from the image [63]. The classification part is composed of fully-connected and softmax layers. The features extracted from the first layers are general and can be reused in different problem domains, while the features extracted from the final layer are specific to the used dataset and task. The primary asset of CNNs is that they can automatically learn and extract hierarchical feature representations [21].

3.1.2. Pre-trained a CNN model

Considering the large size and high computational charge of training a CNN model, importing and employing pre-trained models using the ImageNet dataset [65] is widely used in the research field. The above dataset was created for visual object recognition research [66]. It contains over 14,000,000 human-labeled images, millions of images with bounding boxes, and more than 20,000 classes [67]. Pretrained models are often trained with an ImageNet subset with 1000 classes. The pre-trained models adopted in this study are the following:

– The Visual Geometry Group Network (VGG) [68] is a classical CNN architecture that performed well on the ImageNet dataset. It used 3 x 3 filters to ameliorate the feature extraction process. This architecture is defined as being efficient and straightforward. VGG16 and VGG19 are the two versions of this deep CNN. VGG19 has more layers than VGG16.

– The Dense Convolutional Network (DenseNet) [69] has considerably reduced the number of parameters, minimized the vanishing-gradient issue, and boosted feature propagation [52]. Many versions of DenseNet exist. DenseNet201 has shown remarkable performance while dealing with the COVID-19 diagnosis from CXR images [49]. A 121-layer Dense Convolutional Network (DenseNet121), named CheXNet, has been trained on the Chest-ray 14 dataset [70]. This dataset is the largest publicly available chest X-ray dataset,
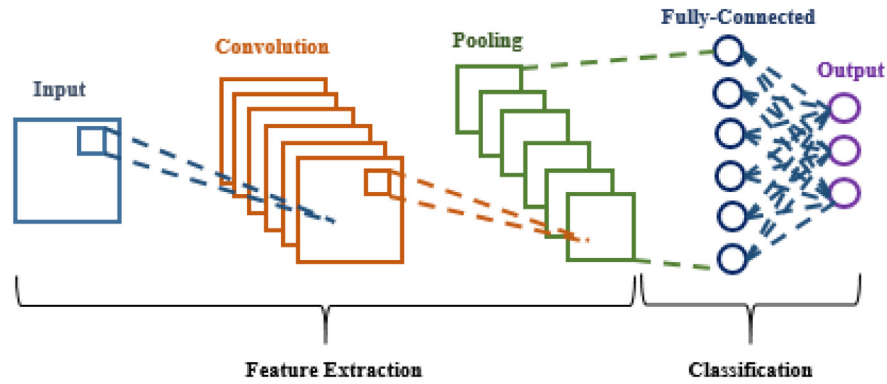
Fig. 2. Typical CNN architecture [64].

with more than 100,000 X-ray images and 14 categories. After evaluation, CheXNet outperformed average radiologist performance on the F1-score metric.

– Residual Neural Network (ResNet) [71] aimed to accomplish powerful convergence behaviors by bouncing some network layers. Trained on ImageNet, the residual nets are evaluated with up to 152 layers of depth (ResNet152), which is eight times deeper than VGG nets. This model was first ranked in the ILSVRC 2015 classification task.

– Xception [72] is a new deep CNN model inspired by Inception, in which a depth-wise separable convolution takes the place of Inception modules. This architecture has relatively outperformed InceptionV3 on the ImageNet dataset. With the wide use of smartphones, Sandler et al. [73] proposed an architecture named MobileNetV2 for machines with limited computing power. This architecture reduces not only the number of learning parameters but also the memory consumption. Furthermore, the pre-trained implementation of this model is available in several well-known deep learning frameworks.

3.1.3. Transfer learning mechanism

Transfer learning includes a variety of pre-trained models that are built on large CNNs [74]. At the end of the learning step, the extracted knowledge is "transferred" to the simplified tasks with limited specific data. This approach is mainly employed because it provides efficient models in a short amount of time. Indeed, one of the most popular transfer learning approaches is to train a CNN on the domain source and then fine-tune it according to the samples from the target domain.

Roughly speaking, it consists of using the features learned when solving a different problem rather than performing the learning procedure from scratch. In computer vision, the idea of transfer learning is built on the use of pre-trained models. Thus, this approach is efficient when training a deep neural network with a limited computational resource. Recently, Zhuang et al. [58] provided a comprehensive survey on transfer learning. Fig. 3 illustrates the idea behind transfer learning.

3.2. Imbalanced learning approach

In predictive modeling, imbalanced classifications are a challenging problem because most ML classification algorithms were developed with the assumption that all the classes have the same number of samples [76]. Class imbalance occurs when the classes in the dataset are not equally distributed. Some classes have a significantly smaller number of samples in the training dataset (minority classes) than the other classes (majority classes). Training a classification algorithm with imbalanced data provides inefficient predictive models, which may poorly classify the minority class. Thus, the minority class, often the most essential for the classification task, is more sensitive to classification errors than the majority class. Buda et al. [35] reported that class imbalance has a negative impact on training CNN models [35]. Indeed, it can influence both the convergence during the training phase and the model generalization during the test phase. To address this problem, the research community developed different solutions [36], including data-level methods and classifier-level methods. In this work, the adopted datasets are significantly imbalanced. This can generate biased learning of the investigated models. Therefore, class-balancing techniques are used to equalize the learning procedure. This study used two-class imbalance strategies: weighted loss and SMOTE.

3.2.1. Weighted loss

While training a neural network, the cost function is crucial because it regulates the layers' weights to produce an adequate ML model. The neural network is fed with a training set during forwarding propagation and generates outputs. The produced outcome is then compared to the target label, and the loss function computes the cost for any variation between the output and the target label. The loss function's partial derivative is determined for each trainable weight of the backward propagation. Thus, the weights are adjusted automatically to provide a model with as minimal loss as possible [63].

The loss function can be changed to overcome the class imbalance problem. Additional weights are applied to the calculated loss for different samples based on the class to which the sample belongs. Introducing these sample weights aims to measure the importance of the computed loss for different samples, depending on whether the samples belong to the majority class or the minority class. A higher weight is assigned to the loss encountered by the samples associated with minor classes, which is the case of the COVID-19 class in this study. This approach of class balancing is known as the weighted class approach [39]. Eq. (1) represents the formula with which the weights of each class are determined.

$$w\left(c\right) = C_c . \frac{\sum_{c=0}^{K} k_c}{K . k_c} \tag{1}$$

where Cc represents the class constant for class c, K is the number of classes, and kc is the number of samples in class c. The generated weight of each class is later included in the loss function, which is, in our case, the standard weighted categorical cross-entropy loss represented by Eq. (2)

$$J_{wcce} = -\frac{1}{M} \sum_{k=1}^{K} \sum_{m=1}^{M} w_k \times y_m^k \times \log(h_\theta(x_{m}, k)) \tag{2}$$

where M is the number of training samples, K represents the number of classes. wk is the weight for class k. xm is the target
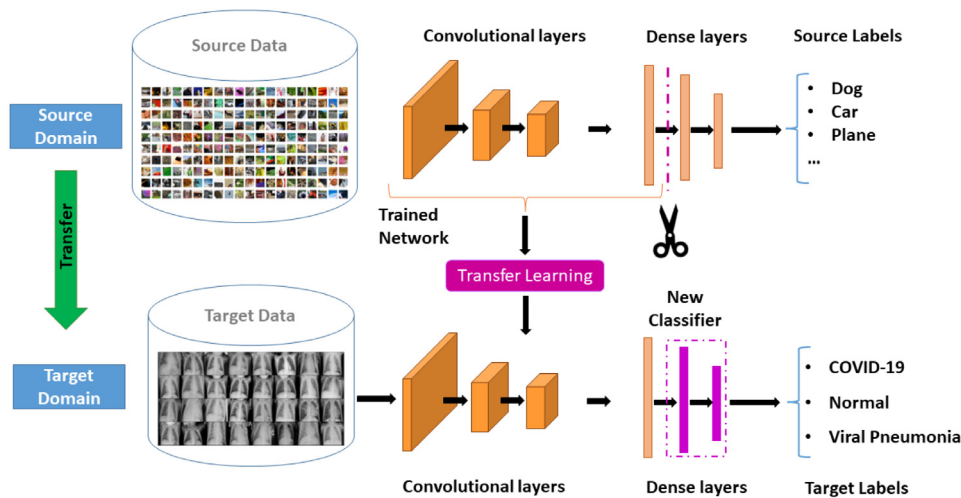
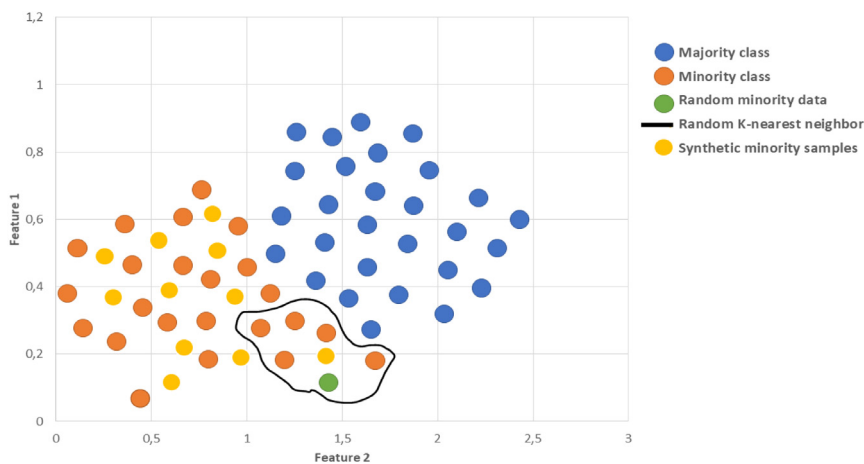**Fig. 3.** The architecture of transfer learning inspired by [75].

Fig. 4. An illustration of SMOTE oversampling technique.

label for training example, m for class k. *x*m is the input for training example m. $h\theta$ is the model with neural network weights θ.

3.2.2. SMOTE

The imbalanced data problem has an impact on the quality of the built model. A model trained with imbalanced data cannot efficiently learn the decision boundary because of the lack of minority class samples. To address this problem, one of the widely used methods is the Synthetic Minority Oversampling Technique (SMOTE), proposed by Chawla et al. [37]. This technique aims to synthesize new samples of the minority class rather than duplicate them. Fig. 4 illustrates how the SMOTE increases and evens out the class distribution of the minority class.

The SMOTE algorithm chooses samples close to the feature space, draws a separation line between the feature space's samples, and creates a novel example at a point near that line. First, a random sample is selected from the minority class. Then, k of that sample's nearest neighbors are found. In most cases, k is equal to 5. Finally, a neighbor is chosen randomly, and a synthetic sample is generated at a randomly selected point between the two samples in the feature space. The procedure is repeated enough times until the minority class has the same proportion as the majority class. The Algorithm SMOTE below is the pseudo-code for SMOTE [37].

3.3. Proposed method

In this study, we trained six state-of-the-art pre-trained CNN models, including DenseNet201, CheXNet, MobileNetV2, ResNet-152, VGG19, and Xception, to classify X-ray images from three well-known COVID-19 datasets. To address the class imbalance of these datasets, we investigated the effectiveness of two different class balancing techniques: WCL and SMOTE. To this end, our proposed method contains three major phases: data preprocessing, classification, and a testing phase. First, the datasets are preprocessed, split, and balanced. Then, the balanced datasets are used for the training of the CNN architectures separately. In this phase, we built and tested six models that classify each dataset into COVID-19, normal (no infection), and viral pneumonia cases. Finally, we conducted a comparative study to determine the model that predicts the best COVID-19 cases from X-ray images. Fig. 5 represents an overview of our proposed method.

3.3.1. Data preprocessing

Due to computational limitations, the input images in this phase are normalized and downsized. In the first experiment in which we applied the WCL, the images of the three datasets were resized to (224×224). Indeed, the used hardware was not capable of high-resolution training models. In the second experiment, in which we applied the SMOTE, the images of the three datasets were resized to (128×128). We did not keep the same size as the

---

Algorithm: *SMOTE (N, P, k)*

Input: *N* number of the samples of the minority class; The SMOTE amount *P%*; *k* the number of nearest neighbors

Output: (*P/100*)* N samples of the synthetic minority class

1. If *P* is less than 100
2.   then Randomize the *N* samples of the minority class
3.   *N=(P / 100) * N*
4.   *P*=100
5. endif
6. *P*= (*int*) (*P/100*) (* the SMOTE's amount is expected to be integer multiples of 100. *)
7. *k* = The number of nearest neighbors
8. *attrsnum* = The number of attributes
9. *Sample* [][]: an array that contains the samples of the original minority class
10. *newindex*: Initialized to 0, it counts the number of generated synthetic samples
11. *Synthetic* [][]: an array to save the synthetic samples.
    (* Compute k nearest neighbors for each minority class sample only. *)
12. for *i* in 1 to *N*
13.   *Calculate the k nearest neighbors for i, then save the indices in the nnarray*
14.   *Generate* (*P, i, nnarray*)
15. end for

    Generate(P, i, nnarray) (* *Function to generate the synthetic samples**)

16. while P $\neq$ 0
17.   select a random number between 1 and *k*, name it *nn*. This step chooses one of the *k* nearest neighbors of *i*.
18.   For *attr*in1 to *attrsnum*
19.   *Calculate :difference = Sample [nnarry[nn]] [attr] - Sample [i][attr]*
20.   *Calculate*: gap = random number between 0 and 1
21.   *Synthetic [newindex][attr]= Sample[i][att]+gap * difference*
22.   endfor
23.   *newindex ++*
24.   *P = P − 1*
25. end while
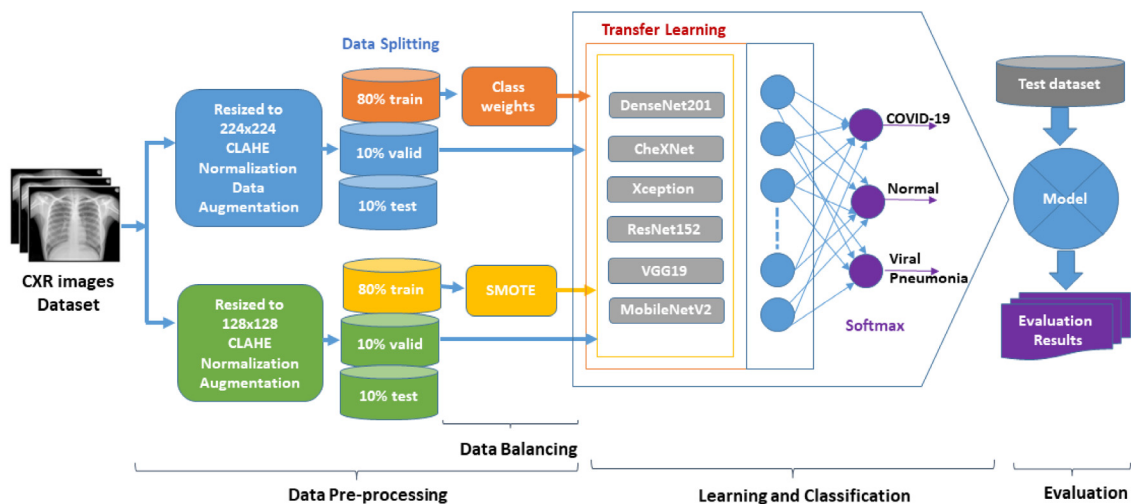26. return (*end of *Generate*. *)
    End of Algorithm

Fig. 5. An overview of the proposed approach.

first experiment because more significant amounts of random-access memory were needed to perform the SMOTE balancing techniques with images larger than (128×128). Due to the small number of images, we applied a data augmentation procedure to datasets 1 and 2 to increase the variety of training sets and reduce overfitting when training the models. We performed a rotation of $10°$ and translation with a shift of (15,15) for the training sets of dataset1 and dataset 2. According to [77], these geometric transformations result in realistic chest x-ray images and help train an ML model for COVID-19 detection. We emphasize that the data augmentation procedure is not used here to balance the dataset. It is only used to avoid overfitting problems. The size of the training sets before and after augmentation is detailed in Table 1.

Finally, the Contrast Limited Adaptive Histogram Equalization (CLAHE) has been applied to get more explicit images. CLAHE is a variant of the adaptive histogram equalization (AHE) algorithm. It was developed to reduce the noise amplification problem [78].

Table 1

The size of the training sets for dataset 1 and dataset 2 before and after data augmentation.

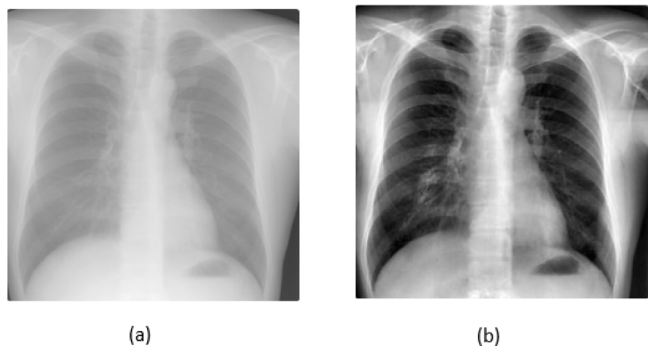| Datasets	Class	Before DA	After DA
Dataset 1	COVID-19	155	465
	Viral pneumonia	1085	3255
	Normal	1085	3255
Dataset 2	COVID-19	200	600
	Viral pneumonia	950	2850
	Normal	950	2850
Dataset 3	COVID-19	800	–
	Viral pneumonia	5000	–
	Normal	5000	–

Fig. 6. (a) is the original image and (b) is the image with Contrast Limited Adaptive Histogram Equalization (CLAHE).

CLAHE aims to increase the contrast of the image's small tiles and to associate the adjacent tiles through bilinear interpolation. This process removes the artificially induced edges. Haghanifar et al. [79] described CLAHE as the most popular enhancement method for various image types. This image enhancement algorithm generated improved nodular-shaped opacity of COVID-19 infection in X-ray images. Fig. 6 shows the results before and after using CLAHE on an image from dataset 2.

At the end of this step, the preprocessed images in each dataset are split using an 80-10-10 scheme. That means 80% of image data is used for the training phase. At the same time, 10% of the dataset is given for the validation and test sets, respectively. Then, the weighted loss balancing technique is applied. In fact, the class weights are computed and assigned to each class in the training set. Then, the weighted categorical cross-entropy loss function is used to train the deep learning models.

In the second part of this study, the SMOTE is applied to generate new images from the minority class [80]. The training samples are first flattened to create a vector embedding of each image (typically known as latent space). Then, this latent space is fed to the SMOTE algorithm, which adds a random factor to the latent space of the image to create a slightly different latent space than the original image of the minority class.

3.3.2. Training and classification

Transfer learning starts by loading the chosen deep learning classification models. The fully connected output layers of the model are fine-tuned, allowing novel head layers to be added and trained. A global spatial average pooling layer is added to create a vector that can be used as a feature descriptor for the input. Finally, three neurons are added, one neuron for each class, followed by the SoftMax activation function for the classification. During the training for each architecture, extensive trials are conducted with different hyper-parameters (optimizer, learning rate, batch size, and the number of epochs). We adopted the Adam

optimizer, or adaptive moment optimization algorithm [81], to adjust the learning rate. Indeed, this optimizer requires relatively low memory and often gives good results with a little hyperparameter tuning. The most relevant results are obtained using a learning rate of 0.0001. The number of epochs and batch size are randomly altered in order to find the values that provide the best results. For each epoch, the evaluation metrics are calculated in the training and validation steps to control the model's performance during these steps and to improve the generated results. All these steps are implemented using Python and executed via Google Collab (25 GB Ram, GPU).

3.3.3. Testing and evaluation

Finally, the testing data is fed to the tuned deep learning classifier to categorize all the image patches into one of three cases: confirmed positive COVID-19, normal, and viral pneumonia (negative COVID-19). At the end of the workflow, the overall performance analysis for each deep learning classifier will be assessed based on the evaluation metrics described in Section 5.

4. Experimental setup

4.1. Datasets description

Since COVID-19 is a recent illness, there are a limited number of publicly available CXR images of infected patients. In this study, three well-known datasets are used. More details about these datasets are presented below. Note that the reported number of images was on the date of access. It is expected to rise over time with more available x-ray images.

– **COVID-19 Radiography Database**[2] is one of the first CXR datasets collected for the COVID-19 diagnosis. A group of researchers from the Middle East and Asia, in collaboration with clinicians, has collected this CXR images database for COVID-19 positive cases along with normal and viral pneumonia images. When used in our experiment, this dataset contained 219 COVID-19 positive images, 1341 normal images, and 1345 viral pneumonia images. The size of the CXR images is 1024 × 1024 [47].

– **Covid Chest Xray Dataset**[3] was collected by Cohen et al. [53] as part of the COVID-19 Image Data Collection of CXR and CT images. It contained 481 chest X-ray images. Normal and viral pneumonia images have been collected from the RSNA Pneumonia Detection Challenge. We chose to use 1500 images for each class. The CXR images in this dataset have different sizes.

– **COVIDx dataset**[4] is one of the most used datasets in the literature, which is employed to train and evaluate the proposed COVID-Net. It includes a total of 13,975 CXR images of 13,870 patients. The COVIDx dataset is the largest publicly available dataset in terms of the number of COVID-19 samples. To generate the COVIDx dataset, five different opensource and accessible data repositories have been combined and modified.

4.2. Evaluation criteria

In order to evaluate the efficiency of our proposed approaches, seven metrics are used for assessing the performance of the CNNs,including accuracy, sensitivity, specificity, precision, F1

---

2 https://www.kaggle.com/tawsifurrahman/covid19-radiography-database.

3 https://github.com/ieee8023/covid-chestxray-dataset.

4 https://github.com/lindawangg/COVID-Net/blob/master/docs/COVIDx.md.

Table 2
Classification results: without balancing technique.

Dataset	CNN	ACC (%)	SEN (%)	SPE (%)	PPV (%)	NPV (%)	F1 (%)	AUC (%)
Dataset 1	**DenseNet201**	**97.51**	**97.51**	**98.01**	**97.50**	**95.41**	**97.50**	**97.59**
	CheXNet	96.15	96.14	98.07	96.15	94.18	96.14	97.01
	Xception	97.05	97.05	98.53	97.05	96.65	97.05	97.10
	ResNet152	95.46	95.46	95.50	95.46	94.43	95.48	96.60
	VGG19	**97.15**	**97.15**	**98.59**	**97.16**	**96.42**	**97.15**	**97.43**
	MobileNetV2	94.10	94.08	97.16	94.13	93.21	94.10	95.58
Dataset 2	**DenseNet201**	**92.60**	**92.30**	**96.10**	**92.34**	**93.54**	**92.25**	**94.41**
	CheXNet	90.49	90.43	95.47	90.50	82.90	90.40	93.40
	Xception	88.94	88.91	93.96	88.90	90.42	91.44	92.34
	ResNet152	90.19	90.20	95.59	90.18	88.92	90.20	92.52
	VGG19	**92.58**	**92.50**	**96.28**	**92.57**	**91.86**	**92.50**	**93.52**
	MobileNetV2	89.94	89.90	94.49	90.28	87.58	90.10	92.42
Dataset 3	**DenseNet201**	**91.63**	**93.85**	**96.92**	**91.69**	**91.15**	**91.66**	**88.70**
	CheXNet	89.38	89.34	94.88	86.38	84.66	88.72	88.00
	Xception	91.44	91.42	95.72	91.40	89.56	91.44	88.61
	ResNet152	91.89	91.96	96.94	92.00	90.05	91.91	88.06
	VGG19	**91.60**	**91.60**	**95.80**	**91.60**	**90.98**	**91.57**	**93.85**
	MobileNetV2	91.44	91.51	95.72	91.44	90.07	91.57	88.45

score, and area under a receiver operating characteristic curve (AUC).

In this study, we are solving a multiclassification problem. Therefore, our models are evaluated using a multi-class confusion matrix [82]. These metrics are derived from the following values: TP stands for true positive, which is the number of COVID-19 patients that correctly predicted infection cases. FN false negative, representing the number of COVID-19 patients, is wrongly predicted as carrying no infection of COVID-19, and P is the total number of COVID-19 patients. TN is true negative and represents the number of non-COVID-19 patients who are correctly predicted as carrying no infection of COVID-19. FP false positive represents the number of non-COVID-19 patients who are wrongly classified as having the virus, and N is the total number of non-COVID-19 patients.

– Sensitivity (SEN) represents the percentage of COVID-19 patients that are correctly predicted as carrying the virus. It is represented by Eq. (3):

$$Sensitivity = \frac{TP}{TP + FN} \qquad (3)$$

– Specificity (SPE) represents the percentage of non-COVID-19 patients that are correctly classified as having no infection of COVID-19 as shown in Eq. (4):

$$Specificity = \frac{TN}{TN + FP} \qquad (4)$$

– Accuracy (ACC) of the classification is represented by Eq. (5):

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \qquad (5)$$

– Positive Predictive Value (PPV) is the exactness that represents the number of positive class predictions that belong to the positive class. It is a measure of the proportion of patients detected by COVID-19 that had the virus. It is defined by Eq. (6):

$$PPV = \frac{TP}{TP + FP} \qquad (6)$$

– Negative Predictive Value (NPV) is the probability that following a negative test result, that individual will truly not have that specific disease. It is represented by Eq. (7):

$$NPV = \frac{TN}{TN + FN} \qquad (7)$$

– F1-Score: is a harmonic mean of Sensitivity and Precision value. It strikes the perfect balance between Precision and Sensitivity, thereby providing a correct evaluation of the model's performance in classifying COVID-19 patients. It is represented by Eq. (8):

$$F1 - Score = 2 \times \frac{PRE \times SEN}{PRE + SEN} \qquad (8)$$

– AUC: it allows us to compute how much the model can distinguish between patients infected by COVID-19 or not. It is represented by Eq. (9):

$$AUC = \frac{SEN + SPE}{2} \qquad (9)$$

5. Ablation study

Before applying a balancing technique, the chosen models are trained on the three above-mentioned imbalanced datasets to compare the results and investigate which model has been improved or deteriorated. The models are trained using the categorical weighted loss function. Table 2 presents the classification outcomes obtained from the transfer learning of the six chosen deep learning models without applying a data balancing technique. By making an overview of this table, notice that all the pre-trained CNNs achieved high accuracy, sensitivity, specificity, precision, F1 score, and AUC. Especially VGG19 and DenseNet201, which generated the highest results on the three datasets. When training using dataset 1, DenseNet201 achieved the highest classification results with an accuracy of 97.51% and AUC of 97.59%. Additionally, VGG19 provides slightly inferior results with an accuracy of 97.15% and an AUC of 97.73%. While MobileNetV2 generated the lowest results with only 94.10% of accuracy and 95.58% of AUC, The results are lower when the models are trained using datasets 2 and 3. Probably because some chest X-ray images collected in the above datasets contain artifacts including arrows, symbols, texts, and pixel-level noise. Nevertheless, DenseNet201 is still on top with accuracy and an AUC of 92.6% and 94.41%, respectively, using dataset 2. followed by VGG19, which generated an accuracy of 92.58%. When trained using dataset 3, DenseNet201 and VGG19 yielded good results. They provide an accuracy of 91.63% and 91.6%, respectively. Still, Xception and MobileNetV2 have the lowest outcomes.

Table 3
Classification results of experiment 1: Using weighted class balancing technique.

Dataset	CNN	ACC (%)	SEN (%)	SPE (%)	PPV (%)(%)	NPV (%)	F1 (%)	AUC (%)
Dataset 1	DenseNet201 121	96.60	96.59	98.29	96.58	94.58	96.36	97.80
	CheXNet	**98.87**	**98.86**	**99.43**	100.00	98.61	**98.21**	**99.15**
	Xception	95.92	95.91	97.95	95.92	92.90	95.91	96.94
	ResNet152	97.51	97.50	98.75	97.51	95.22	97.50	98.13
	VGG19	**98.41**	**98.41**	**99.20**	98.41	96.43	**98.41**	**98.81**
	MobileNetV2	95.69	95.69	97.84	95.69	93.90	95.69	96.76
Dataset 2	**DenseNet201**	**93.58**	**93.58**	**96.79**	93.58	90.18	**93.58**	**95.38**
	CheXNet	91.82	91.82	95.91	97.27	95.70	73.00	94.09
	Xception	89.94	89.91	94.96	90.05	88.15	89.55	92.34
	ResNet152	87.55	87.56	93.77	89.29	90.29	90.86	93.81
	VGG19	**92.97**	**92.87**	**96.48**	92.92	91.86	**92.97**	**91.21**
	MobileNetV2	91.57	91.57	95.78	91.57	90.17	91.68	93.86
Dataset 3	**DenseNet201**	**94.00**	**93.85**	**96.92**	93.94	92.04	**94.87**	**94.08**
	CheXNet	92.97	92.96	96.48	92.96	90.41	92.93	93.06
	Xception	93.00	92.91	96.45	93.00	90.12	92.87	93.62
	ResNet152	90.43	90.43	95.21	90.42	89.37	90.37	89.00
	VGG19	**94.87**	**94.87**	**97.43**	94.92	92.98	**94.87**	**94.86**
	MobileNetV2	95.69	91.51	95.51	91.62	88.53	91.57	90.69

6. Results and discussion

First, the weighted categorical cross-entropy loss function is employed for the training of the CNN models with the three chosen imbalanced datasets. The outcomes of this experiment are shown in Table 3. After several trials, it is observed that CheXNet provided the best classification results while trained on Dataset 1. CheXNet showed a PPV of 100% and almost a perfect AUC of 99.15 as well as high sensitivity, specificity, and F1_score of 98.86%, 99.43%, and 98.21%, respectively. The good results are probably due to the classification framework's hierarchical structure, which considers correlations between different labels. This structure is detailed in [69]. CheXNet, which is originally a DenseNet121, pre-trained on 14 classes of Chest-ray 14 dataset. This architecture consists of one convolution layer with a (7×7) kernel, fifty-eight convolution layers with a (3×3) kernel, and sixty-five convolution layers of (1×1) kernel, four average pooling layers, and one fully connected layer. DenseNets simplify the connectivity pattern between layers [69]. In this work, we replace the final fully-connected layer with one that has three outputs, after which we apply a SoftMax. The weights of the network are initialized with weights from a model pre-trained on ImageNet [65]. The training parameters are detailed in Table 5. On the other hand, VGG19 succeeded in classifying the CXR images with an accuracy of 98.41% and an AUC of 98.81%. Using dataset 2, DenseNet201 provided an overall accuracy of 93.58% and an AUC of 95.38%, while ResNet152 provided the lowest accuracy of 87.55%. For dataset 3, DenseNet201 and VGG19 generated the highest outcomes with accuracy and an AUC over 94%. However, VGG19 has slightly outperformed DenseNet201 in this case. It is observed in Fig. 7 that the use of categorical weighted loss has improved the classification results of the DenseNet201 model. The improvement is highly remarkable with dataset 3. However, with dataset 1 and dataset 2, there is a weak improvement.

The same observation can be concluded from Fig. 8. The VGG19 successfully classified the x-ray images better using the weighted loss techniques on the three datasets. Later, SMOTE is used to balance the training set before it feeds to the CNN models. The outcomes of this experiment are shown in Table 4. The results of experiment 1 and experiment 2 are quite close in terms of top performing models and datasets. However, it is noticeable that DenseNet201 trained on dataset 1 with the application of SMOTE outperformed all the other scenarios in all the used evaluation metrics with an accuracy of 98.64%, a sensitivity of 98.36%, a specificity of 99.35%, a precision of 98.64%, an F1-Score of 98.63% and an AUC of approximately 99%. Followed

by the VGG19, which provided very close outcomes. ResNet152's results have improved slightly more than experiment 1's results, while Xcpetion's classification results have decreased somewhat. MobileNetV2 succeeded in classifying the image with high accuracy and AUC when trained with dataset 1 but generated poor results with datasets 2 and 3.

As depicted in Figs. 7 and 8, SMOTE enhanced the AUC values of DenseNet201 and VGG19 compared to the AUC of the training with an imbalanced dataset. The proposed balancing techniques have remarkably increased the classification results of the chosen pre-trained deep learning models even though, in some cases, one of the techniques surpassed the other in terms of results. A summary of the proposed models that provided the best results is presented in Table 5, in addition to the hyper-parameters used for their training.

The best-obtained results are compared with the other recently proposed approaches to detect COVID-19 from X-ray images. Table 6 shows that our proposed model outperformed other works with the three datasets in terms of performance. For dataset 1, our model achieved an accuracy of almost 99%, while Chowdhury et al. [47] achieved only 97.9%. In addition, our model provided a perfect precision of 100%, compared with only 98% for Chowdhury et al. [47]. For dataset 2, our proposed WCL with the DenseNet201 model achieved a higher classification rate compared to the models proposed by Ozturk et al. [83] and Haghanifar et al. [79] in terms of accuracy and F1-score, as shown in Table 5. Our model that applied WCL with VGG19 on dataset 3 outperformed the COVID-NET model developed by Wang et al. [44] that, achieved an accuracy of 92.4%, a sensitivity of 88%, and a precision of 91%, while our approach achieved on the same dataset an accuracy of 94.87%, a sensitivity of 94.87% and a precision of almost 95%. Our approach also succeeded in classifying X-ray images for COVID-19 diagnosis better than Oh et al. [30] in terms of accuracy and F1-score.

7. Limitations

Due to a lack of computational resources and low memory, the training process of most DL networks takes many hours, an average of 3.5 h for each algorithm. Indeed, seven different deep convolution neural networks were trained with a multi-classification head in this work. Each CNN was trained with a batch size chosen randomly between 8, 16, or 32, with an input image of the size (244×244) while applying the WCL and the size of (128×128) while using the SMOTE. Overall, the training on the three datasets took 300 h. The computational complexity
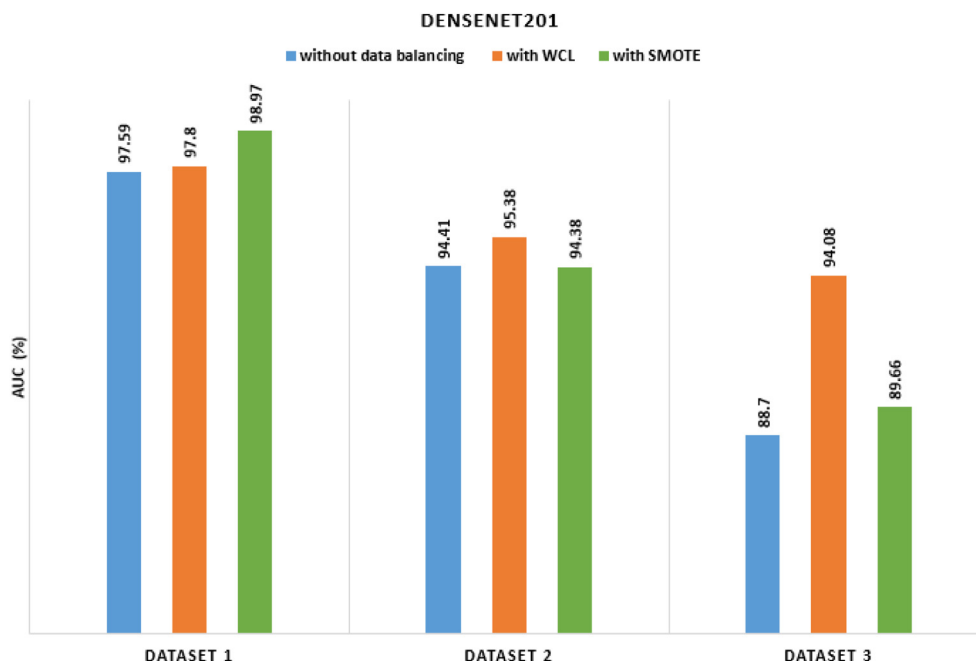
Fig. 7. Comparison of AUC metric provided by DenseNet201 for each dataset: (Left bar) without data balancing, (middle bar) with WCL and (right bar) with SMOTE.

Table 4
Classification results of experiment 2: Using SMOTE balancing technique.

Dataset	CNN	ACC (%)	SEN (%)	SPE (%)	PPV (%)	NPV (%)	F1 (%)	AUC (%)
Dataset 1	**DenseNet201**	**98.64**	**98.36**	**99.35**	98.64	98.64	**98.63**	**98.97**
	CheXNet	97.51	97.50	98.75	97.51	97.51	97.50	98.20
	Xception	96.15	96.15	98.07	96.15	96.15	96.14	96.94
	ResNet152	95.46	95.64	97.73	95.46	95.46	95.46	96.60
	VGG19	**98.19**	**98.19**	**99.09**	98.19	98.19	**98.19**	**98.63**
	MobileNetV2	97.51	97.51	98.75	97.54	97.54	97.50	98.13
Dataset 2	**DenseNet201**	**92.28**	**92.20**	**96.10**	92.20	92.20	**92.20**	**94.38**
	CheXNet	91.19	91.19	95.59	91.18	91.18	91.12	93.42
	Xception	89.44	89.43	94.71	89.43	89.43	89.43	91.81
	ResNet152	91.82	91.82	95.91	91.82	91.82	91.82	93.84
	VGG19	**92.08**	**92.07**	**96.03**	92.08	92.08	**92.07**	**93.95**
	MobileNetV2	83.31	89.30	94.65	89.41	89.41	89.29	92.16
Dataset 3	**DenseNet201**	**92.71**	**92.71**	**96.36**	92.77	92.77	**92.71**	**90.66**
	CheXNet	92.02	92.02	92.02	96.01	96.01	92.10	89.54
	Xception	90.00	90.01	95.00	90.00	90.00	90.02	86.53
	ResNet152	91.32	91.32	95.66	91.32	91.32	91.32	88.93
	VGG19	**93.66**	**93.66**	**96.83**	93.66	93.66	**93.66**	**92.00**
	MobileNetV2	91.38	91.38	95.69	91.65	91.65	91.35	89.00

Table 5
A summary of the proposed models and training hyperparameters with best results.

Dataset	Models	Loss function	Activation function	Classifier	Optimizer (lr:learning rate)	Batch	Epochs
Dataset 1	WCL + CheXNet	WCL	RELU	Softmax	Adam (lr = 10^4)	32	50
	SMOTE + DenseNet201	Categorical cross-entropy	RELU	Softmax	Adam (lr = 10^4)	8	72
Dataset 2	WCL + DenseNet201	WCL	RELU	Softmax	Adam (lr = 10^4)	8	20
	SMOTE + DenseNet201	Categorical cross-entropy	RELU	Softmax	Adam (lr = 10^4)	8	50
Dataset 3	WCL + VGG19	WCL	RELU	Softmax	Adam (lr = 10^4)	8	20
	SMOTE + VGG19	Categorical cross-entropy	RELU	Softmax	Adam (lr = 10^4)	8	50

is proportional to the dataset size, the used balancing technique, and the depth of the trained CNN. The training of DensNet201 has the longest training time, 8h16m50s (for a 128 × 128 input image, batch size of 8 and 72 epochs). In contrast, the VGG19 had the lowest training time since it has fewer layers. The application of the WCL had a neglectable execution time compared to SMOTE. Indeed, the use of SMOTE as a balancing technique was very

memory and time-consuming. The overall time of SMOTE running was between 15 and 90 min, depending on the size of the training set. One of the major limitations of this work was the high computational consumption. Moreover, the proposed model needs to be validated externally before undergoing any clinical use. This step is crucial as it properly assesses the model's performance against real cases in different settings.
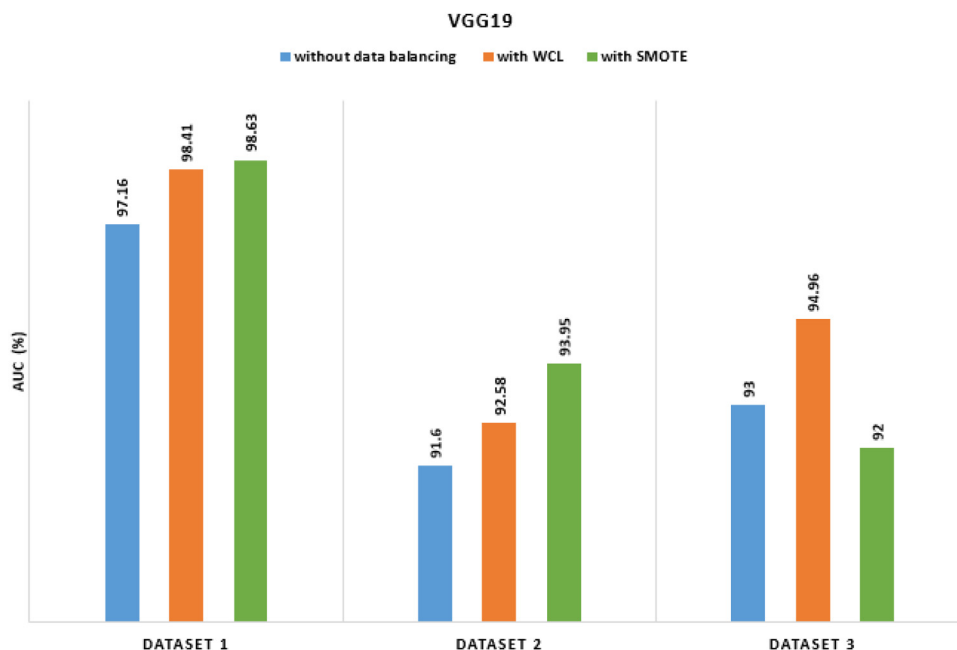
Fig. 8. Comparison of AUC metric provided by VGG19 for each dataset: (Left bar) without data balancing, (middle bar) with WCL, and (right bar) with SMOTE.

Table 6
Comparative analysis of the proposed model with recently proposed models.

Dataset	Author	Model	ACC (%)	SEN (%)	SPE (%)	PPV (%)	F1 (%)	AUC (%)
Dataset 1	Chowdhury et al. [47]	DenseNet201	97.9	97.9	98.8	97.95	–	–
	Bassi et al. [49]	DenseNet121	98.3	–	–	98.3	98.30	–
	Proposed models	**WCL + CheXNet**	**98.87**	**98.86**	**99.43**	**100**	98.21	**99.15**
		SMOTE + DenseNet201	98.64	98.36	99.35	98.64	**98.63**	98.97
Dataset 2	Haghanifar et al. [79]	CheXNet	–	–	–	–	85	–
	Ozturk et al. [83]	DarkNet	87.02	–	–	–	–	–
	Proposed models	**WCL + DenseNet201**	**93.58**	**93.58**	**96.79**	**93.58**	**93.58**	**95.38**
		SMOTE + DenseNet201	92.28	92.20	96.10	92.20	92.20	94.38
Dataset 3	Wang et al. [44]	COVID-Net	92.4	88	–	91	–	–
	Oh et al. [30]	ResNet-18	91.9	–	–	76.9	–	–
	Proposed models	**WCL + VGG19**	**94.87**	**94.87**	**97.43**	**94.92**	**94.87**	**94.86**
		SMOTE + VGG19	93.66	93.66	96.83	93.66	93.66	92.00

8. Conclusion

One of the major issues identified in most related works concerning the COVID-19 disease diagnosis using X-ray images is class imbalance. This paper presents an investigation of two classic data balancing techniques applied to three well-known datasets. Six deep learning models are trained on the selected datasets separately to identify which combination of the balancing technique and CNN architecture provides the best classification results. Each trained model was evaluated using benchmark performance metrics,e.g., accuracy, precision, area under curve, specificity, and F1 score under two different experiences concerned with imbalanced learning. With extensive trials, it was observed that models achieve different scores in different scenarios, among which DenseNet201 and VGG19 displayed better performance for the multi-classification of COVID-19 samples. This study was developed using limited computational resources, causing a long training time, especially when applying the SMOTE as a data balancing technique. As an extension to this work, we can investigate, using more powerful computational resources, other deep learning models and data balancing techniques such as data generation using generative adversarial networks (GAN). Furthermore, we can explore other datasets that contain more samples of COVID-19.

Declaration of competing interest

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

References

[1] World health organization (WHO), Coronavirus disease (COVID-19) dashboard, 2021, https://covid19.who.int/. Accessed 16 March 2021.

[2] A.E. Gorbalenya, S.C. Baker, et al., Coronaviridae Study Group of the International Committee on Taxonomy of Viruses: The species Severe acute respiratory syndrome-related coronavirus: classifying 2019-nCoV and naming it SARS-CoV-2, Nat. Microbiol. 5 (2020) 536–544, http://dx.doi.org/10.1038/s41564-020-0695-z.

[3] World health organization (WHO), Diagnostic testing for SARS-CoV-2. Interim guidance, 2020, https://apps.who.int/iris/handle/10665/334254. Accessed 16 March 2021.

[4] M. Nicola, Z. Alsafi, et al., The socio-economic implications of the coronavirus pandemic (COVID-19): A review, IJS 78 (2020) 185–193, http://dx.doi.org/10.1016/j.ijsu.2020.04.018.

[5] M. Soui, N. Mansouri, R. Alhamad, M. Kessentini, K. Ghedira, NSGA-II as feature selection technique and AdaBoost classifier for COVID-19 prediction using patient's symptoms, Nonlinear Dynam. 106 (2) (2021) 1453–1475.

[6] K. Subbarao, S. Mahanty, Respiratory virus infections: Understanding COVID-19, Immunity 52 (6) (2020) 905–909, http://dx.doi.org/10.1016/j.immuni.2020.05.004.

[7] J.H. Tanne, E. Hayasaki, M. Zastrow, P. Pulla, P. Smith, A.G. Rada, Covid-19: how doctors and healthcare systems are tackling coronavirus worldwide, BMJ 368 (2020) m1090, http://dx.doi.org/10.1136/bmj.m1090.

[8] W. Wang, Y. Xu, R. Gao, R. Lu, K. Han, G. Wu, W. Tan, Detection of SARS-CoV-2 in different types of clinical specimens, JAMA 323 (18) (2020) 1843–1844, http://dx.doi.org/10.1001/jama.2020.3786.

[9] Aljazeera News, Bangladesh scientists create $3 kit. Can it help detect COVID-19? 2020, https://www.aljazeera.com/news/2020/3/24/bangladesh-scientists-create-3-kit-can-it-help-detect-covid-19. Accessed 1 November 2020.

[10] N. Kurani, K. Pollitz, D. Cotliar, N. Shanosky, C. Cox, COVID-19 Test Prices and Payment Policy, Peterson-KFF Health System Tracker, 2020, https://www.healthsystemtracker.org/brief/covid-19-test-prices-and-payment-policy/. Accessed 1 November 2020.

[11] G.D. Rubin, C.J. Ryerson, L.B. Haramati, et al., The role of chest imaging in patient management during the COVID-19 pandemic: A multinational consensus statement from the fleischner society, Radiology 158 (1) (2020) 106–116, http://dx.doi.org/10.1148/radiol.2020201365.

[12] T. Ai, Z. Yang, H. Hou, et al., Correlation of chest CT and RT-PCR testing for coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases, Radiology 296 (2019) E32–40, 2020.

[13] Y. Fang, H. Zhang, J. Xie, M. Lin, L. Ying, P. Pang, W. Ji, Sensitivity of Chest CT for COVID-19: Comparison to RT-PCR, Radiology 296 (2) (2020) E115–E117, http://dx.doi.org/10.1148/radiol.2020200432.

[14] T. Ai, Z. Yang, H. Hou, C. Zhan, C. Chen, W. Lv, Q. Tao, Z. Sun, L. Xia, Correlation of chest CT and RT-PCR testing for coronavirus disease 2019 (COVID-19) in China: A report of 1014 cases, Radiology 296 (2) (2020) E32–E40, http://dx.doi.org/10.1148/radiol.2020200642.

[15] 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.compbiomed.2020.103792.

[16] E.C. Lin, Radiation risk from medical imaging, Mayo Clin. Proc. 85 (12) (2010) 1142–1146, http://dx.doi.org/10.4065/mcp.2010.0260.

[17] American College of Radiology, ACR recommendations for using chest radiography and computed tomography (CT) for suspected COVID-19 infection, 2020, https://www.acr.org/Advocacy-and-Economics/ACR-PositionStatements/Recommendations-for-Chest-Radiography-and-CT-for-Suspected-COVID19-Infection. Accessed 1 November 2020.

[18] A. Borakati, A. Perera, J. Johnson, et al., Diagnostic accuracy of X-ray versus CT in COVID-19: a propensity-matched database study, BMJ Open 10 (2020) e042946, http://dx.doi.org/10.1136/bmjopen-2020-042946.

[19] H.Y.F. Wong, H.Y.S. Lam, A.H. Fong, et al., Frequency and distribution of chest radiographic findings in COVID-19 positive patients, Radiology (2020) http://dx.doi.org/10.1148/radiol.2020201160.

[20] A. Hosny, C. Parmar, J. Quackenbush, L.H. Schwartz, H.J.W.L. Aerts, Artificial intelligence in radiology, Nat. Rev. Cancer 18 (8) (2018) 500–510, http://dx.doi.org/10.1038/s41568-018-0016-5.

[21] D. Albashish, Ensemble of adapted convolutional neural networks (CNN) methods for classifying colon histopathological images, PeerJ Comput. Sci. 8 (2022) e1031.

[22] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, 2012, pp. 1097–1105.

[23] X. Chen, X. Wang, K. Zhang, R. Zhang, K. Fung, T.C. Thai, K. Moore, R.S. Mannel, H. Liu, B. Zheng, Y. Qiu, Recent advances and clinical applications of deep learning in medical image analysis, Med. Image Anal. 79 (2022) 102444.

[24] D. Albashish, R. Al-Sayyed, A. Abdullah, M.H. Ryalat, N.A. Almansour, Deep CNN model based on VGG16 for breast cancer classification, in: 2021 International Conference on Information Technology, ICIT, IEEE, 2021, pp. 805–810.

[25] X. Liu, L. Faes, A. Kale, S. Wagner, D. Fu, A. Bruynseels, T. Mahendiran, G. Moraes, M. Shamdas, C. Kern, J. Ledsam, M.D. Schmid, K. Balaskas, E. Topol, L. Bachmann, P. Keane, A. Denniston, A comparison of deep learning performance against healthcare professionals in detecting diseases from medical imaging: a systematic review and meta-analysis, Lancet Digit. Health 1 (2019) http://dx.doi.org/10.1016/S2589-7500(19)30123-2.

[26] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444, http://dx.doi.org/10.1038/nature14539.

[27] J. Rasheed, A. Jamil, A.A. Hameed, U. Aftab, J. Aftab, S.A. Shah, D. Draheim, A survey on artificial intelligence approaches in supporting frontline workers and decision makers for COVID-19 pandemic, Chaos Solitons Fractals (2020) 110337, http://dx.doi.org/10.1016/j.chaos.2020.110337.

[28] 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) 109–944, http://dx.doi.org/10.1016/j.chaos.2020.109944.

[29] K. El Asnaoui, Y. Chawki, Using X-ray images and deep learning for automated detection of coronavirus disease, J. Biomol. Struct. Dyn. (2020) 1–12, http://dx.doi.org/10.1080/07391102.2020.1767212.

[30] Y. Oh, S. Park, J.C. Ye, Deep learning COVID-19 features on CXR using limited training data sets, IEEE Trans. Med. Imaging 39 (8) (2020) 2688–2700, http://dx.doi.org/10.1109/TMI.2020.2993291.

[31] D.P. Fan, T. Zhou, G. Ji, et al., Inf-net: Automatic COVID-19 lung infection segmentation from CT images, IEEE Trans. Med. Imaging 39 (8) (2020) 2626–2637, http://dx.doi.org/10.1109/TMI.2020.2996645.

[32] R.M. Pereira, D. Bertolini, L.O. Teixeira, C.N. Silla, Y.M.G. Costa, COVID-19 identification in chest X-ray images on flat and hierarchical classification scenarios, Comput. Methods Programs Biomed. 194 (2020) http://dx.doi.org/10.1016/j.cmpb.2020.105532.

[33] E. Tartaglione, C.A. Barbano, C. Berzovini, M. Calandri, M. Grangetto, Unveiling COVID-19 from CHEST X-ray with deep learning: A hurdles race with small data, IJERPH 17 (18) (2020) http://dx.doi.org/10.3390/ijerph17186933.

[34] M.M. Islam, F. Karray, R. Alhajj, J. Zeng, A review on deep learning techniques for the diagnosis of novel coronavirus (COVID-19), IEEE Access 9 (2021) 30551–30572, http://dx.doi.org/10.1109/ACCESS.2021.3058537.

[35] M. Buda, A. Maki, M.A. Mazurowski, A systematic study of the class imbalance problem in convolutional neural networks, Neural Netw. 106 (2018) 249–259.

[36] H. He, Y. Ma, Imbalanced Learning: Foundations, Algorithms, and Applications, first ed., Wiley-IEEE Press, New York, 2013.

[37] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, SMOTE: Synthetic minority over-sampling technique, J. Artificial Intelligence Res. 16 (2002) 321–357, http://dx.doi.org/10.1613/jair.953.

[38] F. Thabtah, S. Hammoud, F. Kamalov, A. Gonsalves, Data imbalance in classification: Experimental evaluation, Inform. Sci. 513 (2020) 429–441.

[39] Y. Ho, S. Wookey, The real-world-weight cross-entropy loss function: Modeling the costs of mislabeling, IEEE Access 8 (2019) 4806–4813, http://dx.doi.org/10.1109/ACCESS.2019.2962617.

[40] M. Loey, F. Smarandache, M. Khalifa, E. Nour, Within the lack of chest COVID-19 X-ray dataset: A novel detection model based on GAN and deep transfer learning, Symmetry 12 (4) (2020) 651, http://dx.doi.org/10.3390/sym12040651.

[41] R. Jain, M. Gupta, S. Taneja, D.J. Hemanth, Deep learning-based detection and analysis of COVID-19 on chest X-ray images, Appl. Intell. 51 (3) (2021) 1690–1700.

[42] G. Bargshady, X. Zhou, P.D. Barua, R. Gururajan, Y. Li, U.R. Acharya, Application of CycleGAN and transfer learning techniques for automated detection of COVID-19 using X-ray images, Pattern Recognit. Lett. 153 (2022) 67–74.

[43] A.Y.A. Saeed, A.E. Ba Alawi, Covid-19 diagnosis model using deep learning with focal loss technique, in: 2021 International Congress of Advanced Technology and Engineering, ICOTEN, 2021, pp. 1–4, http://dx.doi.org/10.1109/ICOTEN52080.2021.9493477.

[44] L. Wang, Z.Q. Lin, A. Wong, COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images, Sci. Rep. 10 (1) (2020) 19549, http://dx.doi.org/10.1038/s41598-020-76550-z.

[45] T. Li, Z. Han, B. Wei, Y. Zheng, Y. Hong, J. Cong, Robust screening of COVID-19 from chest X-ray via discriminative cost-sensitive learning, 2020, ArXiv, abs/2004.12592.

[46] M.S.S. Khobahi, C. Agarwal, CoroNet: A deep network architecture for semi-supervised task-based identification of COVID-19 from chest X-ray images, MedRxiv (2020) http://dx.doi.org/10.1101/2020.04.14.20065722.

[47] M.E.H. Chowdhury, T. Rahman, A. Khandakar, et al., Can AI help in screening viral and COVID-19 pneumonia? IEEE Access 8 (2020) 132665-132676, http://dx.doi.org/10.1109/ACCESS.2020.3010287.

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

[49] P.R. Bassi, R. Attux, A deep convolutional neural network for COVID-19 detection using chest X-rays, 2020, ArXiv, abs/2005.01578.

[50] E. Luz, P. Silva, R. Silva, et al., Towards an effective and efficient deep learning model for covid-19 patterns detection in x-ray images, 2020, arXiv:2004.05717.

[51] N.S. Punn, S. Agarwal, Automated diagnosis of COVID-19 with limited posteroanterior chest X-ray images using fine-tuned deep neural networks, Appl. Intell. (2020) 1–14, http://dx.doi.org/10.1007/s10489-020-01900-3, Advance online publication.

[52] E.E.D. Hemdan, M.A. Shouman, M.E. Karar, COVIDX-Net: A framework of deep learning classifiers to diagnose COVID-19 in X-ray images, 2020, arXiv preprint arXiv:2003.11055.

[53] J.P. Cohen, P. Morrison, L. Dao, Covid-19 image data collection, 2020, https://github.com/ieee8023/covid-chestxray-dataset. Accessed 1 November 2020.

[54] A. Bhattacharyya, D. Bhaik, S. Kumar, P. Thakur, R. Sharma, R.B. Pachori, A deep learning based approach for automatic detection of COVID-19 cases using chest X-ray images, Biomed. Signal Process. Control 71 (2022) 103182.

[55] F. Demir, DeepCoroNet: A deep LSTM approach for automated detection of COVID-19 cases from chest X-ray images, Appl. Soft Comput. 103 (2021) 107160.

[56] A. Canziani, A. Paszke, E. Culurciello, An analysis of deep neural network models for practical applications, 2016, arXiv preprint arXiv:1605.07678.

[57] B.K. Obaid, R. Subhi, M. Zeebaree, O.M. Ahmed, Deep learning models based on image classification: A review, Int. J. Sci. Bus. 4 (11) (2020) 75–81, http://dx.doi.org/10.5281/zenodo.4108433.

[58] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, Q. He, A comprehensive survey on transfer learning, Proc. IEEE 109 (2021) 43–76, arXiv:1911.02685.

[59] M. Tsiakmaki, G. Kostopoulos, S. Kotsiantis, O. Ragos, Transfer learning from deep neural networks for predicting student performance, Appl. Sci. 10 (6) (2020) 2145, http://dx.doi.org/10.3390/app10062145.

[60] Y. Liu, Z. Li, H. Liu, Z. Kan, Skill transfer learning for autonomous robots and human–robot cooperation: A survey, Robot. Auton. Syst. 128 (2020) 103515, http://dx.doi.org/10.1016/j.robot.2020.103515.

[61] J. Wang, H. Zhu, S.H. Wang, Y.D. Zhang, A review of deep learning on medical image analysis, Mob. Netw. Appl. (2020) http://dx.doi.org/10.1007/s11036-020-01672-7.

[62] Y. Bengio, Learning deep architectures for AI, Found. Trends Mach. Learn. 2 (1) (2009) 1–127.

[63] F. Chollet, Deep Learning with Python, Manning Publications Co., USA, 2017.

[64] S. Balaji, Binary image classifier CNN using TensorFlow, 2020, https://medium.com/techiepedia/binary-image-classifier-cnn-using-tensorflow-a3f5d6746697. Accessed 10 December 2020.

[65] J. Deng, W. Dong, R. Socher, L.J. Li, Kai Li, Li Fei-Fei, ImageNet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, http://dx.doi.org/10.1109/cvpr.2009.5206848.

[66] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105, http://dx.doi.org/10.1145/3065386.

[67] D. Mishkin, N. Sergievskiy, J. Matas, Systematic evaluation of convolution neural network advances on the Imagenet, Comput. Vis. Image Underst. 161 (2017) 11–19, http://dx.doi.org/10.1016/j.cviu.2017.05.007.

[68] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, 2015, CoRR, abs/1409.1556.

[69] G. Huang, Z. Liu, K.Q. Weinberger, Densely connected convolutional networks, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2017, pp. 2261–2269, http://dx.doi.org/10.1109/CVPR.2017.243.

[70] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, et al., Chexnet: Radiologist-level pneumonia detection on chest X-rays with deep learning, 2017, ArXiv, abs/1711.05225.

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

[72] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2017, pp. 1800–1807, http://dx.doi.org/10.1109/CVPR.2017.195.

[73] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L. Chen, MobileNetV2: Inverted residuals and linear bottlenecks, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 4510–4520, http://dx.doi.org/10.1109/CVPR.2018.00474.

[74] A. Voulodimos, N. Doulamis, A. Doulamis, E. Protopapadakis, Deep learning for computer vision: A brief review, Comput. Intell. Neurosci. (2018) http://dx.doi.org/10.1155/2018/7068349.

[75] W. Tao, M. Al-Amin, H. Chen, M.C. Leu, Z. Yin, R. Qin, Real-time assembly operation recognition with fog computing and transfer learning for human-centered intelligent manufacturing, Procedia Manuf. 48 (2020) 926–931.

[76] M. Kuhn, K. Johnson, Applied Predictive Modeling, Springer, New York, 2013.

[77] M. Elgendi, M.U. Nasir, Q. Tang, D. Smith, J.-P. Grenier, C. Batte, B. Spieler, W.D. Leslie, C. Menon, R.R. Fletcher, N. Howard, R. Ward, W. Parker, S. Nicolaou, The effectiveness of image augmentation in deep learning networks for detecting COVID-19: A geometric transformation perspective, Front. Med. 8 (2021) 629134, http://dx.doi.org/10.3389/fmed.2021.629134.

[78] S.M. Pizer, E.P. Amburn, J.D. Austin, et al., Adaptive histogram equalization and its variations, Comput. Vis. Graph. Image Process. 39 (3) (1987) 355–368, http://dx.doi.org/10.1016/s0734-189x(87)80186-x.

[79] A. Haghanifar, M.M. Majdabadi, S. Ko, COVID-CXNet: Detecting COVID-19 in frontal chest X-ray images using deep learning, 2020, ArXiv, abs/2006.13807.

[80] B. Aditya, How to use SMOTE for dealing with imbalanced image dataset for solving classification problems, 2020, https://medium.com/swlh/how-to-use-smote-for-dealing-with-imbalanced-image-dataset-for-solving-classification-problems-3aba7d2b9cad. Accessed 1 November 2020.

[81] S. Ruder, An overview of gradient descent optimization algorithms, 2016, arXiv:1609.04747.

[82] M. Sokolova, G. Lapalme, A systematic analysis of performance measures for classification tasks, Inf. Process. Manag. 45 (4) (2009) 427–437, http://dx.doi.org/10.1016/j.ipm.2009.03.002.

[83] 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, 103792. http://dx.doi.org/10.1016/j.compbiomed.2020.103792.