



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.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.



Metaheuristics based COVID-19 detection using medical images: A review

Mamoona Riaz, Maryam Bashir^{*}, Irfan Younas

FAST School of Computing, National University of Computer and Emerging Sciences, Lahore, Pakistan

ARTICLE INFO

Keywords:

COVID-19
Metaheuristics
Image classification
Nature-inspired algorithm

ABSTRACT

Many countries in the world have been facing the rapid spread of COVID-19 since February 2020. There is a dire need for efficient and cheap automated diagnosis systems that can reduce the pressure on healthcare systems. Extensive research is being done on the use of image classification for the detection of COVID-19 through X-ray and CT-scan images of patients. Deep learning has been the most popular technique for image classification during the last decade. However, the performance of deep learning-based methods heavily depends on the architecture of the deep neural network. Over the last few years, metaheuristics have gained popularity for optimizing the architecture of deep neural networks. Metaheuristics have been widely used to solve different complex non-linear optimization problems due to their flexibility, simplicity, and problem independence. This paper aims to study the different image classification techniques for chest images, including the applications of metaheuristics for optimization and feature selection of deep learning and machine learning models. The motivation of this study is to focus on applications of different types of metaheuristics for COVID-19 detection and to shed some light on future challenges in COVID-19 detection from medical images. The aim is to inspire researchers to focus their research on overlooked aspects of COVID-19 detection.

1. Introduction

Coronavirus is a type of virus that targets the human respiratory system. It is labeled as “Corona” due to the “crown-like spikes” on its surface. A few examples of this virus include Severe Acute Respiratory Syndrome (SARS), Middle East Respiratory Syndrome (MERS), and the seasonal flu [1]. Coronavirus was first detected in 1937 in birds. This infectious disease caused bronchitis in birds. Later in 1960, scientists found a type of coronavirus in the human nose during observation of the common cold. This type of corona causes a mild illness that mostly occurs in winter (cold season) [1].

COVID-19 was first discovered in Wuhan, China on December 1, 2019 [1]. It started to spread rapidly to the rest of the world due to a large number of international flights connecting the entire world. Almost all continents were affected by this disease within months of its first case in China. Scientists claim that the virus originated from an animal with an acquired common infectious disease. Many researchers believe that COVID-19 first infected bats before spreading to other animals and humans. Although this information is yet to be proven, scientists are still searching its source and its spreading pattern [1]. The origin of this virus is yet to be discovered. The virus has evolved over the last few decades and was known as Severe Acute Respiratory Syndrome

Coronavirus (SARS-CoV) and MERS in 2002 and 2012, respectively. However, the current form of this virus, which was discovered in Wuhan, is called the 2019 coronavirus (2019-nCoV). This particular virus can cause serious pneumonia [2]. In 2020, the International Committee on Taxonomy of Virus (ICTV) announced the 2019 coronavirus as SARS-Cov-2, while the disease was named coronavirus disease 2019.

According to the World Health Organization statistics, there is a total of 349,641,119 confirmed cases of COVID-19, out of which 5,592,266 have lost their lives, as reported on January 24, 2021. Among all countries, the United States is on top of the list, with total reported cases of 69,727,991. Fig. 1 shows the statistics of confirmed cases of COVID-19.

COVID-19 virus exponentially spread around the globe. In March 2020, the World Health Organization declared it a pandemic. The reason for its exponential growth is that it spreads through the touch, breath, or cough of a COVID-19 infected person. The cough of a COVID-19 patient emits droplets, which can enter the mouth or nose of a healthy person and cause the same infection. Therefore, it is important to maintain a distance of six feet from COVID-19 carriers [1]. If protective measures are being followed and a safe distance is being maintained, then this disease will stay within the carrier bodies only.

^{*} Corresponding author.

E-mail address: maryam.bashir@nu.edu.pk (M. Bashir).

With the current situation where COVID-19 has been declared a pandemic and has no permanent cure, the only solution is to maintain a safe distance from infected people. In the absence of any reliable cure to this disease, early and accurate detection of COVID-19 positive cases is very important. This will help to separate infected populations from healthy individuals. There are multiple ways to identify COVID-19 patients, such as the standard confirmatory clinical test, including Reverse Transcription-Polymerase Chain Reaction (RT-PCR) and chest image screening [4]. Meanwhile, RT-PCR is a manual, complex, and time-consuming test. However, the limited availability of test kits, domain experts in the hospitals, and rapid increase in the number of infected people make it very difficult and expensive to perform these tests for the whole population repeatedly. On the other hand, chest scanning is relatively less time-consuming and less costly. In such circumstances, the detection of COVID-19 using chest X-rays can provide a major aid. However, there is a need to develop a system with high precision. Chest scan image classification lies in the domain of computer vision. Meanwhile, many studies have been carried out to classify images of the COVID-19 patients from other patients and healthy people. The majority of the research used machine learning and deep learning algorithms for image classification of chest scans. Fig. 2 presents basic steps of image processing.

Deep Learning (DL) and Machine Learning (ML) have shown state-of-the-art performance on a wide range of image classification tasks. Hyper-parameter settings play a vital role in the performance of deep learning [5] and machine learning models. In DL, hyper-parameters include the number of layers, neurons per layer, activation function, learning rates, dropout rates, and batch size. There is no optimal generic configuration for hyper-parameter optimization for all image classification tasks. These parameters can be optimized manually, but it is very time-consuming and require expert knowledge. Automatic optimization of hyper-parameters can be done using grid search or random search, but these methods are computationally expensive and take a long time to produce quality results. In the recent past, metaheuristics have been widely and successfully used for the optimization of hyper-parameters of ML and DL models to solve different COVID-19 detection and other computer vision tasks. Metaheuristics are search algorithms that can find near-optimal solutions in less time compared to random search. They have shown promising results for feature selection in both deep learning and machine learning models. Fig. 3 presents a series of steps when metaheuristic optimizers are applied in image processing.

Since various metaheuristics have been applied for hyper-parameter optimization and feature selection in different COVID-19 detection-related tasks, there is a need to review the state-of-the-art work on metaheuristics-based COVID-19 detection using medical images.

Previous literature is impoverished regarding the comprehensive and technical review of metaheuristics-based methods for diagnosing COVID-19 infected patients. Owing to this disadvantage, this paper presents a detailed review of state-of-the-art metaheuristics-based optimization solutions to reduce a large number of parameters and obtain high-quality results for chest image classification. The main objectives of this study are as follows:

- Explore the applications of metaheuristics at different steps of COVID-19 medical image classification.
- Present the type of available datasets for COVID-19 chest images and the pre-processing techniques used for these datasets.
- Analyze different feature extraction, feature selection, and hyper-parameter optimization methods for COVID-19 medical image classification.
- Discuss and compare the accuracy of existing approaches for the classification of COVID-19 medical images.
- Present research gaps and limitations in existing applications of metaheuristics for COVID-19 images.
- Present future research directions in the prediction of COVID-19 from medical images for current researchers working in the domain of image classification.

Fig. 4 shows the number of references for various use cases of nature-inspired algorithms in COVID-19 detection. Feature selection is the most common use case for nature-inspired algorithms in COVID-19 detection. Meanwhile, other sections of this paper are organized as follows. Section 2 presents details of datasets used for image classification of chest images. Section 3 discusses the pre-processing techniques used in image classification. Section 4 discusses different feature extraction techniques used for COVID-19 image classification. Section 5 presents feature selection techniques and description of various nature-inspired algorithms used for feature selection in COVID-19 detection. Section 6 presents applications of metaheuristics for parameter optimization of machine learning and deep learning algorithms. Section 7 presents different types of classification algorithms for COVID-19 chest image classification. Section 8 discusses the results of different approaches for COVID-19 detection. Section 9 discusses future challenges, and section 10 discusses limitations in the task of image classification for COVID-19 detection. Lastly, section 11 concludes the paper.

2. Dataset analysis

In this paper, there are two types of images used in COVID-19 detection research: X-ray images and Computed Tomography (CT)

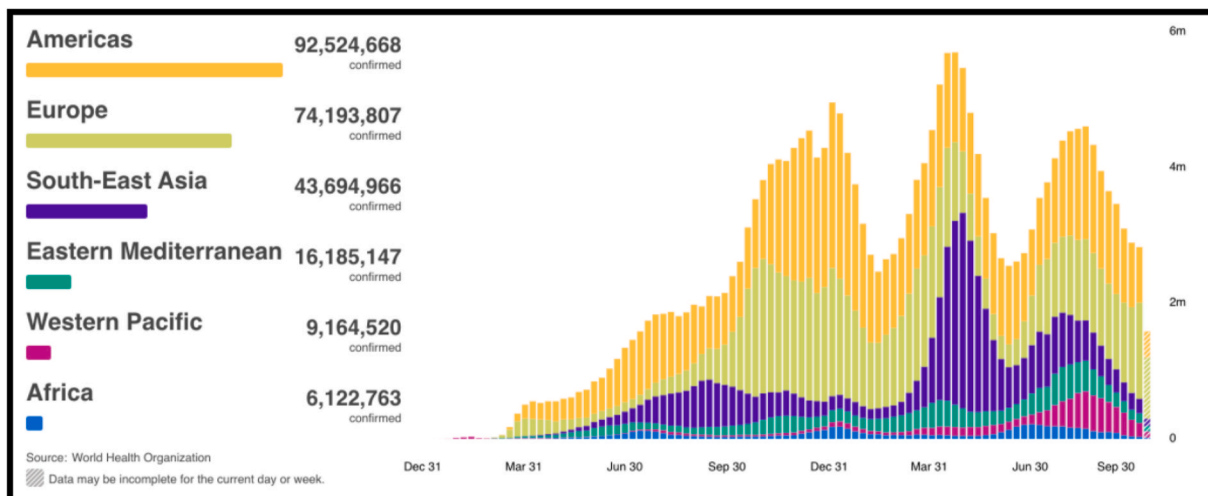


Fig. 1. Statistics of confirmed cases of COVID-19 in some countries until October 18, 2021 [3].

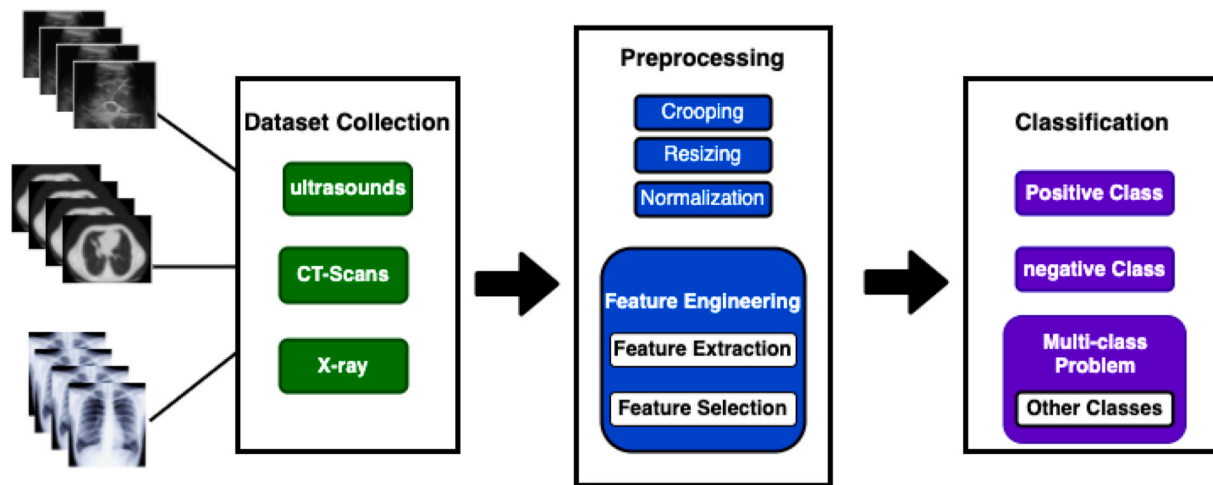


Fig. 2. Image processing basic pipeline.

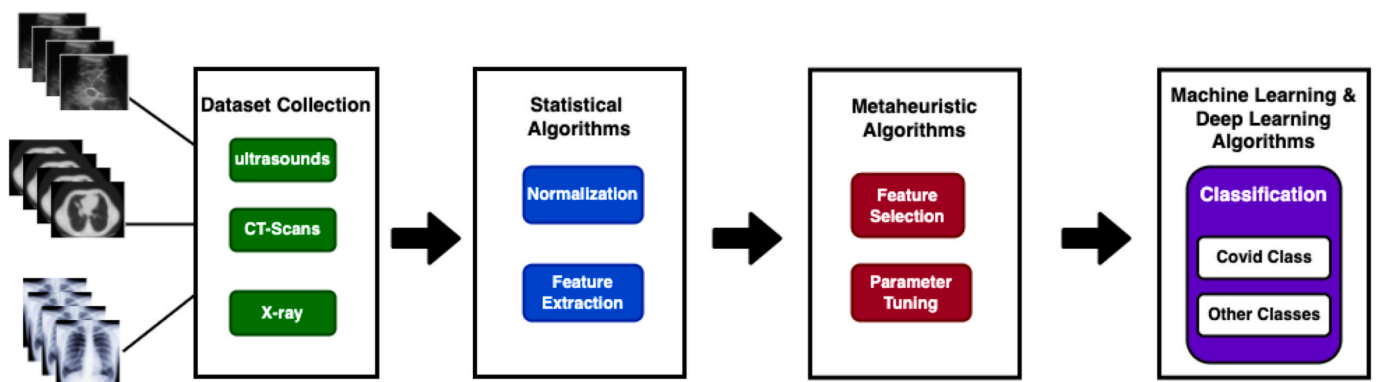


Fig. 3. Metaheuristics use cases in image processing.

scan images. X-ray is a grayscale two dimensional (2D) image that is used to capture dense tissues, while a CT scan is a more enriched 3D colored image that captures more details such as bones, soft tissues, and blood vessels at once. According to previous research, some studies used only X-ray images, while others used CT scans. In addition, some have used both types of images. In this study, the total number of datasets used is 22. The details of these datasets are shown in Table 1 and Fig. 5.

2.1. Dataset references

This section presents the total references or number of times the dataset was used in research. A commonly used dataset shows that the dataset is well defined and contains all features required for image classification. Fig. 6 shows that the dataset (COVID-chestxray-dataset) is at the top, with 16 citations. On the other hand, datasets 3, 5, and 6 are used in three studies each.

2.2. Types of data

As previously mentioned, there are two types of image datasets for COVID-19 research. This section presents the statistics about which type of dataset is more commonly used in previous research. These statistics are shown in Fig. 7. Meanwhile, X-ray images are the most common type of dataset used in past research.

2.3. Classes of dataset

Fig. 8 shows the details about the number of classes of each dataset.

Most datasets have two to four classes. By considering the number of citations of these datasets as shown in Fig. 7, it can be observed that the dataset with the most citations is Dataset 2, which is comprised of six classes. The dataset with a maximum number of classes is Dataset 12, which has only one citation. This shows that the datasets with a smaller number of classes have been used more often in past research. One reason can be that, as the number of classes increases, it becomes more difficult to classify the images with the correct class.

2.4. Count of COVID-19 images

The major concern in each dataset is the count of COVID-19 images, as a reasonable number of COVID-19 images is very important to avoid class imbalance. Table 2 presents the class balance between COVID-19 and other classes. Nonetheless, there is a class imbalance as most datasets have a small number of COVID-19 images compared to other images. There are few datasets with zero COVID-19 images, and there are some datasets with a zero count for other classes. Actually, these special datasets are added to control the class imbalance. From Table 2, we can observe that almost all researchers use more than one dataset for experiments. However, this factor is not sufficient to remove the imbalance issue completely. Many papers propose some augmentation techniques to resolve this issue. These techniques are discussed in the preprocessing section.

From the above discussion, it can be concluded that most of the research was conducted on an X-ray (2D gray-scaled images) dataset with three classes. Therefore, any new researcher can adopt this method to obtain good results. Meanwhile, there is a research gap for new

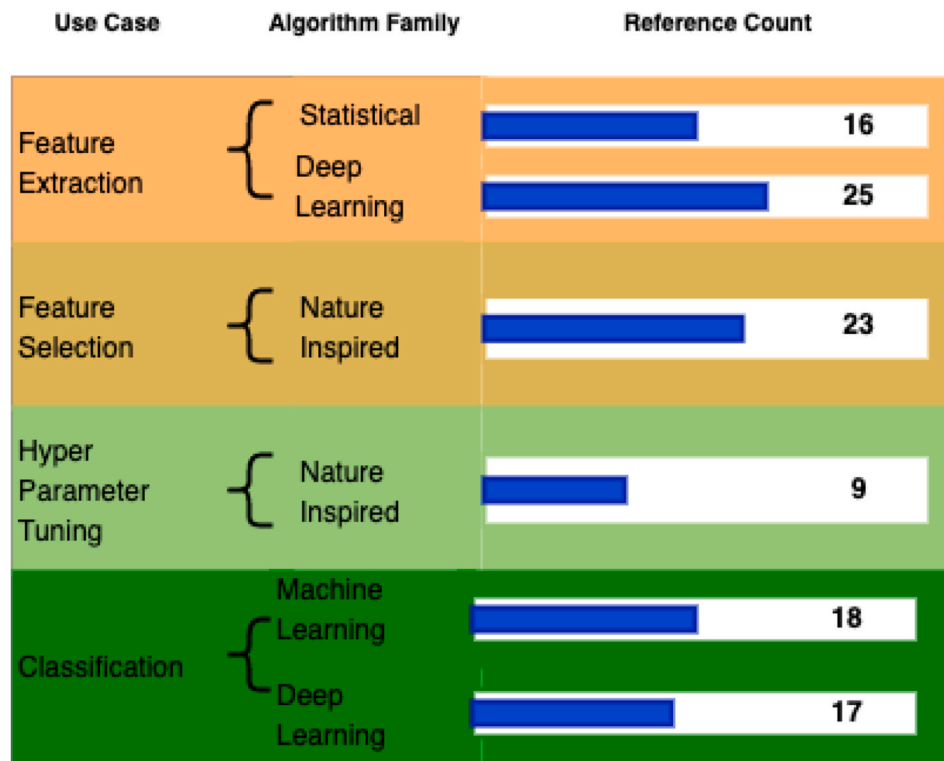


Fig. 4. Number of references for various use cases of nature-inspired algorithms in COVID-19 detection.

Table 1
Titles of datasets referred in papers.

Datasets	References	Title
Dataset1	[6,7]	MH-CovidNet
Dataset2	[8–34]	COVID-chestxray-dataset
Dataset3	[8,17,18,23,35]	Chest X-Ray Images (Pneumonia)
Dataset4	[8]	COVID-19 Ultrasound
Dataset5	[10,12,18,24,27,35,36]	COVID-19 Radiography Database
Dataset6	[10,15,18]	Senza-COVID-19
Dataset7	[11]	A large dataset of real patients CT scans for COVID-19 identification
Dataset8	[28,37,38]	SARS-COV-2 Ct-Scan Dataset
Dataset9	[12,13,26,32]	Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification
Dataset10	[12]	COVID Data Gradient Crescent
Dataset11	[14]	COVID-19 Infection Presenting with CT Halo Sign
Dataset12	[15]	ChestX-ray14
Dataset13	[15]	ChestX-ray8
Dataset14	[39]	COVID-19 CT Lung and Infection Segmentation Dataset
Dataset15	[18]	Chest Imaging
Dataset16	[18,24]	Radiopaedia-COVID-19
Dataset17	[19]	Extensive COVID-19 X-Ray and CT Chest Images Dataset
Dataset18	[19]	Large Dataset of Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images
Dataset19	[35]	DeepCOVID
Dataset20	[35]	COVID-19-Chest-X-ray-Detection
Dataset21	[8,25,28,40]	COVID-CT
Dataset22	[27]	Novel COVID-19 Chest X-ray Repository
Dataset23	[29]	COVID-19 chest X-ray

experiments on CT-scan images with more than three classes.

3. Preprocessing techniques

In some cases, raw images as input can be used, but it is often not feasible to directly apply classification algorithms to original datasets.

There are some other anomalies, such as image imbalance issues, that can affect the performance of an algorithm. Therefore, it is very important to resolve these problems before applying the algorithms. A summary of preprocessing techniques is presented in Table 3. These techniques are discussed in the following sections.

3.1. Normalization

Image size and intensity of pixels play a vital role in describing the structure of the tissue and bones and help to specify important segments that should be examined [44]. The following section discusses the three main techniques of normalization, namely MinMax, Image Resizing, and Gray Scaling.

3.1.1. Min-max

MinMax is the simplest form of normalization. In this normalization, we convert the minimum value of data into zero and maximum to one [45]. All values are bounded in the range [0, 1]. This makes it easier to apply threshold-based algorithms after this normalization is performed on pixel values.

$$\frac{x - \min}{\max - \min} \tag{1}$$

In equation (1), *min* is the minimum intensity value of the pixel, *max* is the maximum intensity value of the pixel, and *x* represents the pixel value we wish to convert. This formula is applied to all pixel values for normalization. One downside about this method is that it cannot handle outliers as it adds skewness to data, allowing all values to lie in a very close range. However, this is very rare in the case of image datasets.

3.1.2. Image resizing

Image resizing is an increase or decrease in the size of an image or a change in the number of pixels of an image. Neural networks are the most popular technique for image classification. Their input, however, must be a fixed length. The images are available in different sizes and thus it is necessary to resize the images before passing them as input to

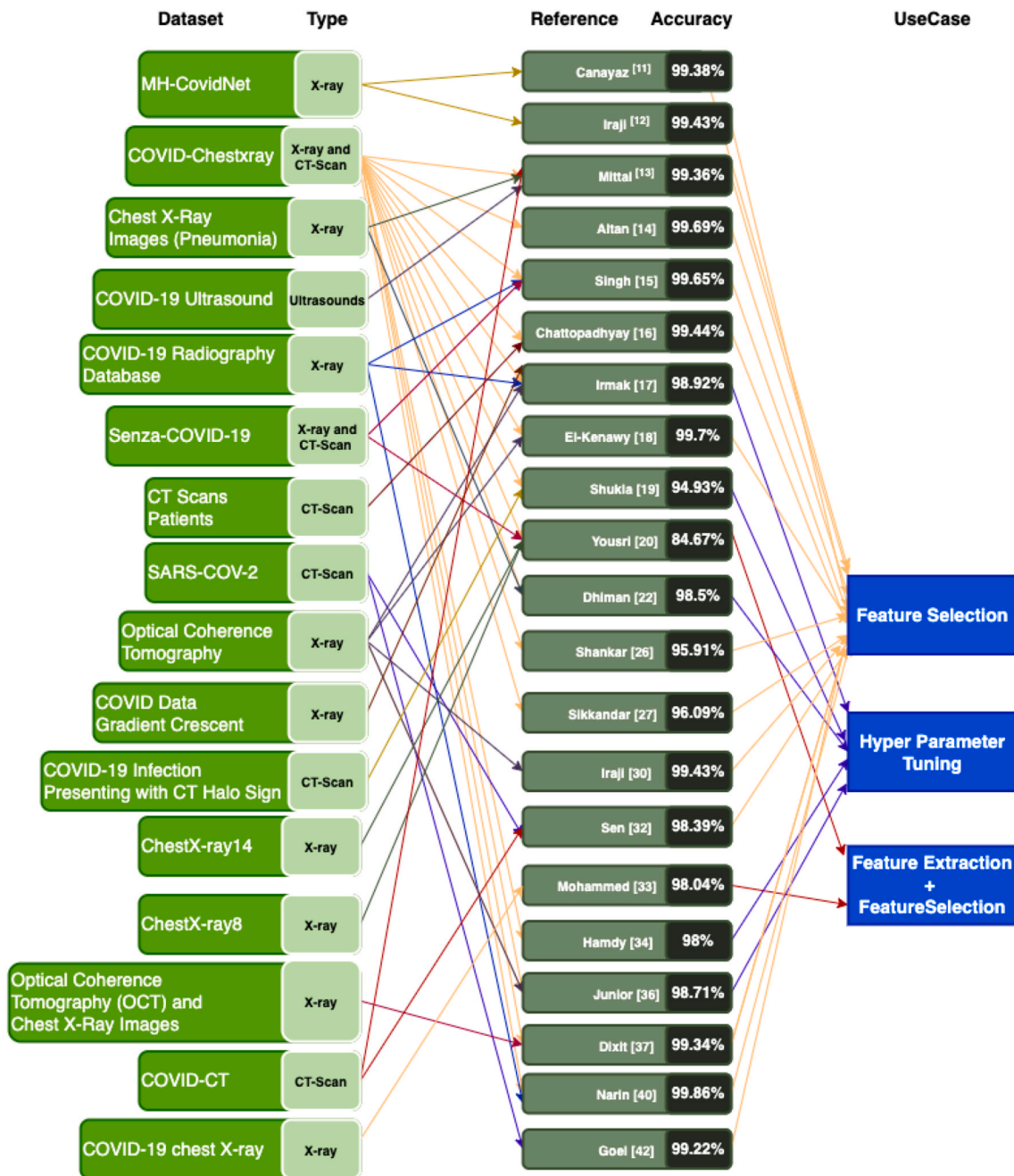


Fig. 5. Datasets vs number of citations of the dataset for different use cases along with the reported accuracy scores.

the neural network. Interpolation is the most commonly used technique for this task. It takes averages of neighboring pixels and fills the values in a matrix template of a given size [46].

3.1.3. Gray scale image

There are two main types of images: 2D X-ray images and 3D CT-scan images. 3D images are colored and have an extra layer or dimension for keeping color values. These color values are viewed as another feature and include some processing costs. In most cases, this extra layer does not play an important role in classification tasks. Therefore, it is advisable to ignore this extra dimension in the preprocessing step. The gray image intensity of RGB (Red Green Blue) is considered equal, and it is mostly stored in an eight-bit integer. Eight-bit integer can store up to 256 possible different shades of gray from black to white [47]. After converting to grayscale, the intensity value of RGB lies in the range [0 –

–255] scale.

3.2. Augmentation

Data augmentation fixes the class imbalance by creating synthetic examples of classes with fewer objects. Data imbalance is a common problem in image processing or computer vision, especially in the medical field. In most cases, the number of images of a particular disease class is less than those of other classes. This class imbalance can introduce some biases in the model. It is therefore a good idea to perform augmentation before training the model. Many augmentation techniques are used to generate synthetic images to balance the count of each class. Data augmentation is applied to all data. First, class labels are balanced, and then feature selection and classification are performed. The following section discusses the two main techniques used by

DataSet VS ReferenceCount

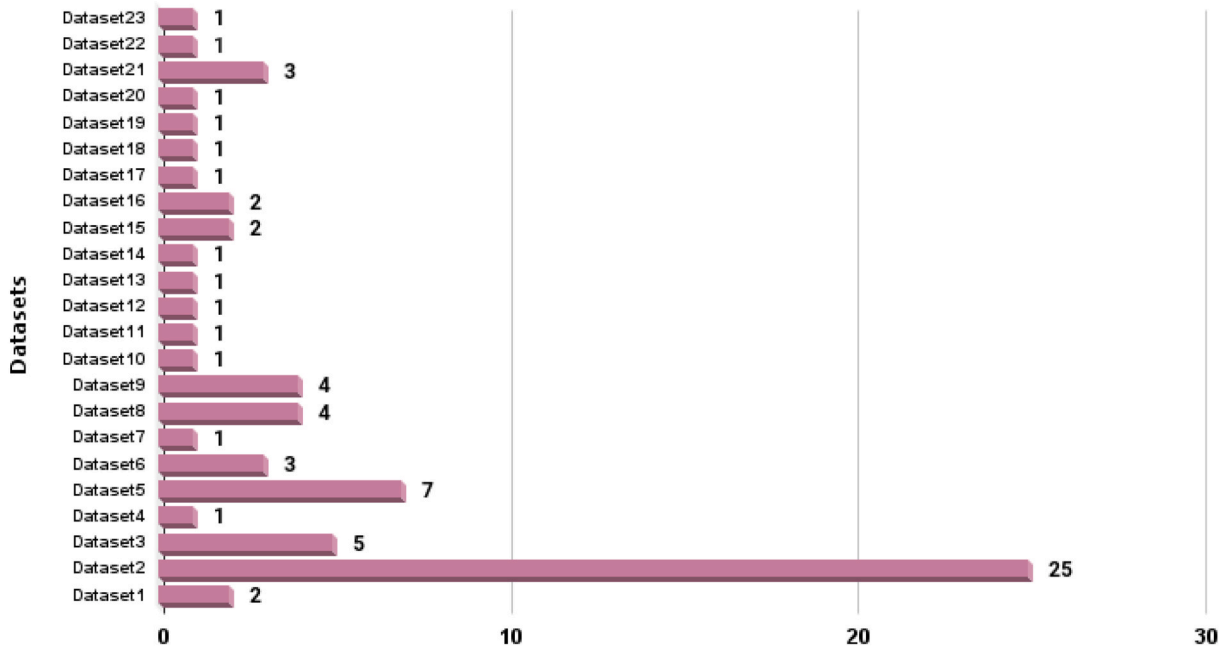


Fig. 6. Datasets vs number of citations of the dataset.

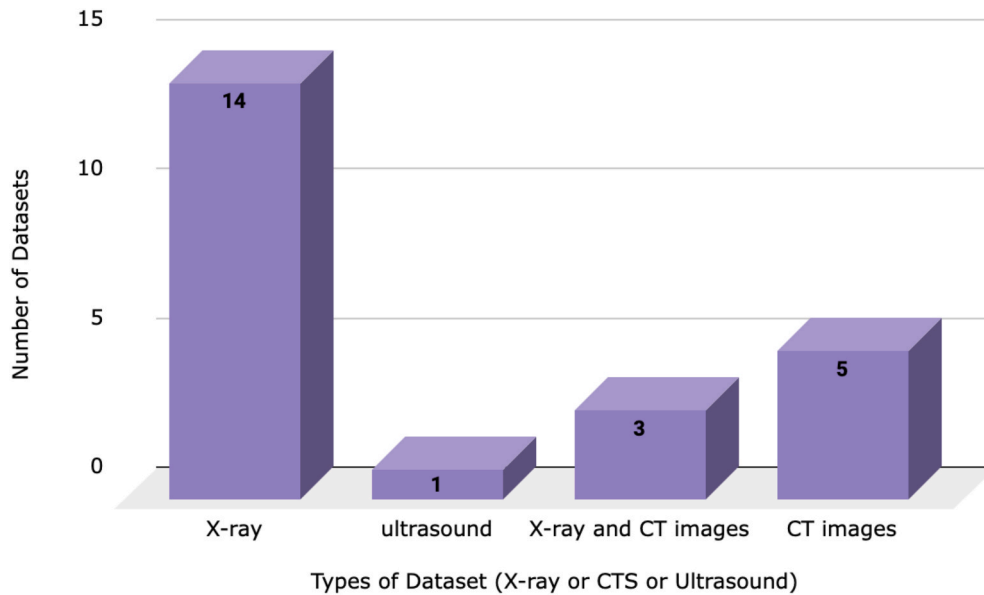


Fig. 7. Types of datasets vs number of datasets.

research related to this study.

3.2.1. Translation and rotation

Translation or rotation of an image changes the shape of the image, which can be observed by the human eye. However, these operations also change the pixel values, making the newly generated image a new instance of training.

Translation is a process of moving images around the x-axis, y-axis, or both axes. This augmentation technique is very simple and effective, as the major characteristics of images remain almost the same. The results are the same regardless of the distance from which the image was captured.

$$x_2 = x_1 + x \tag{2}$$

$$y_2 = y_1 + y \tag{3}$$

Equations (2) and (3) show an example of image translation.

Rotation of any image involves its movement around the axis at some angle. Rotation can generate many images with new pixel values representing the same class. However, the drawback of this method is that the image may lose information if it is not a square image. This issue can be resolved by using the image resizing technique.

$$x_2 = \cos(\theta) * (x_1 - x) - \sin(\theta) * (y_1 - y) + x \tag{4}$$

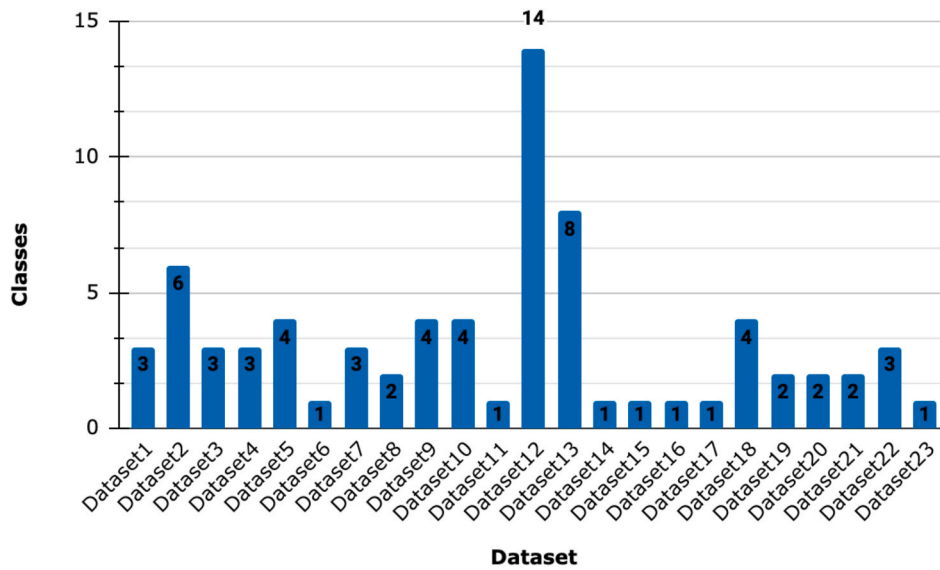


Fig. 8. Datasets Vs Total classes.

Table 2

Count of images from COVID-19 cases and other classes.

Datasets	Covid-cases	Other Images
Dataset1	364	0
Dataset2	468	178
Dataset3	0	5573
Dataset4	8	24
Dataset5	3616	17 549
Dataset6	115	0
Dataset7	2168	2005
Dataset8	1252	1229
Dataset9	0	NAN
Dataset10	69	231
Dataset11	2	0
Dataset12	0	112 120
Dataset13	0	108 948
Dataset14	20	0
Dataset15	132	0
Dataset16	289	0
Dataset17	9471	8128
Dataset18	0	NAN
Dataset19	520	5000
Dataset20	76	17
Dataset21	100	100
Dataset22	752	3223
Dataset23	357	0

Table 3

Preprocessing techniques used by different studies.

Reference	Preprocessing technique	Name
[8,37,41]	Normalization	MinMax
[18]	Normalization	Image Resizing
[20]	Normalization	Gray scaling
[18,42]	Augmentation	Rotation, and Translation
[43]	Augmentation	SMOTE and the LSH-SMOTE

$$y_2 = \sin(\theta) * (x_1 - x) + \cos(\theta) * (y_1 - y) + y \tag{5}$$

Equations (4) and (5) show that rotation takes an angle θ , and then it will move both x and y axis around each pixel (x, y) with θ factor. The new place of the image is different from the original one after rotation. This can be perceived as a camera taking images from different angles, and this new image will still represent a realistic image. Therefore, image rotation can be used to add many images by using different values

of angles.

3.2.2. LSH-SMOTE (Locality Sensitive Hashing synthetic minority Oversampling technique)

This algorithm is a mixture of two different algorithms, namely SMOTE and LSH (Locality Sensitive Hashing).

SMOTE is an augmentation technique that uses statistical techniques to increase the minority class. This algorithm takes k -neighbors and then generates a new synthetic data point (image). The new point will not be the exact copy of any point from the input, but will rather represent a new point in class [48]. This algorithm works better for limited features except for datasets with large features. It uses k -neighbors, and thus there is a possibility that it will add some noisy points to the data.

LSH is a hashing algorithm. It creates cases against each point and assigns similar cases to the points that are very close to each other according to the similarity feature. In the case where two points lie in the same feature space, their locality will be similar as well. It is mostly used to identify the nearest neighbor [49].

LSH-SMOTE is a combination of both techniques. It deals with negative points raised in SMOTE by using K Nearest Neighbors (KNN) for the generation of new candidates. LSH always assigns low scores to the points that are from the majority class. Thus, the class imbalance will be resolved. Many factors need to be taken into account before inputting data to an algorithm, such as verifying all data points are the same size and checking if all classes are equally represented in the dataset, or some majority classes are dominant. These issues can be resolved by using different normalization techniques, such as image resizing, grayscale, min-max normalization, etc. [50]. In the case of imbalanced data, augmentation techniques can be used, such as rotation translation, LSH-SMOTE, etc. [51,52].

4. Feature extraction

A machine learning or deep learning algorithm learns specific patterns or repeated patterns that lie in data points (images) belonging to the targeted class. It is important to clean raw data to ensure meaningless features are reduced. This phase of feature extraction contributes to extracting first-level features or upper-level features, such as edges, corners, or segments of image data. Furthermore, almost all papers report it as the first step after preprocessing. Convolution Neural Network (CNN) is the most popular approach for feature extraction in literature. The following sections present some details about the

different feature extraction methods used for COVID-19 detection from images.

4.1. Convolution neural network (CNN)

This neural network structure was designed specifically for the image data type. It resembles the connection pattern of nerve cells in the human brain and is inspired by the tissues of the visual cortex. It can capture spatial and temporal patterns, making it favorable for image processing. It requires very little preprocessing [53]. Feature extraction is composed of two types of layers, namely the convolution layer and pooling layer.

Convolution layer is responsible for feature extraction, such as colors, edges, shapes. This layer works by convoluting a kernel or filter on an input image. In the first convolution, it extracts low-level features. By adding more convolution layers, this learning can be extended to high-level features such as objects and events.

Pooling layer can reduce the number of parameters. After extracting the features, many pixels become less significant. This algorithm can perform well without including additional features. Therefore, this layer helps to reduce the size of features. It consists of three types of selections, namely max pooling, min pooling, and average pooling. Max pooling selects pixels with maximum value from the selected window. Unlike max pooling, min pooling selects pixels with minimum value. Average pooling takes the average of all pixels appearing in a window.

Flatten Vector: After learning all features, the final matrix is used to convert into a flat vector, which is passed to the neural network.

Fully Connected Layer: This layer consists of a fully connected neural network. This final layer assigns the class labels and it is used for classification.

The architecture of CNN can be customized by changing the count and arrangement of layers based on the current issue. However, there are some popular CNN architectures tested on different datasets with good performance, e.g., VGGNet (Very Deep Convolutional Networks) [54], ResNet (Residual Network) [55], etc. Table 4 shows the performance of different CNN architectures based on accuracies reported by different researchers for COVID-19 detection from images. Some researchers used customized variants, while some used a combination of multiple architectures. For example, in Ref. [18], it uses both CheXNet (Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning) [56] and DenseNet (Dense Convolutional Network) [57]. The maximum accuracy is reported by customized layers of Inception [58] and EfficientNet [59]. Therefore, these can be considered as best

Table 4
Accuracies reported by different CNN architectures in different references.

CNN architecture	CNN architecture variants	Reference (Use case is COVID-19 detection)	Accuracy %
VGG [54]	VGG19	[6,32]	99.38, 98.71
	VGG16	[23,27]	96.09, 94.17
ResNet [55]	ResNet18	[11]	99.44
	ResNet50	[13,32,36]	99.7, 94.17, 99.86
	ResNet101	[17,36]	98.5, 99.86
EfficientNet [59]	EfficientNet-B0	[9]	99.69
	EfficientNet-B1	[16]	97.62
AlexNet [60]	AlexNet	[8,43]	99.5, 99.36
	AlexNet Modifier	[42]	98.55
CheXNet [56]	CheXNet	[18]	97.94
DenseNet [57]	DenseNet201	[18]	97.94
LeNet [61]	LeNet-5	[35]	99.11
GoogLeNet [62]	GoogLeNet	[14]	98.38
Inception [58]	Inception	[63]	99.8
	InceptionV3	[27,32]	94.17, 98.71
Customised layers	Customised layers	[37,64]	99.99, 97.78
Xception [65]	Xception	[27]	94.17

performers out of the listed CNN architectures.

Fig. 9 shows CNN architectures and the count of research papers on COVID-19 detection, where these are mentioned as part of the proposed solution. ResNet and AlexNet are ranked first, while VGG and EfficientNet are in second place. From these statistics, we can conclude that ResNet and AlexNet are more commonly used for COVID-19 related image processing.

4.2. Generative adversarial networks (GAN)

GAN is composed of two main parts of neural networks. One is a generator, and the other is a discriminator. The generator is used to generate synthetic data of the given classes, while the discriminator is used to classify the data. The discriminator assigns a label to the given record and, in case of misclassification, the discriminator will be penalized and will be forced to tune its weight according to the target. In the event the discriminator continues to generate accurate detections, then the generator model will be penalized for not producing diverse images. The goal of this model is to find a balance between these two networks. For image processing, this model can increase the size of the dataset, thus avoiding data deficiency problems. In the case of COVID-19 datasets, it was observed that images of COVID-19 were less than those of other classes.

A drawback of these algorithms is a large number of parameters, but this can be controlled by using some optimization algorithms. GAN was used for feature extraction and its parameter was reduced and optimized by incorporating Whale Optimization Algorithm (WOA) [38], achieving a 99.22% accuracy.

4.3. Statistical algorithms

Unlike ML, DL, and nature-inspired algorithms, statistical methods are based on fixed mathematical steps or equations, which do not change for any dataset. They require a specific shape of input and return values after passing through functions. In image processing, these kinds of algorithms are mostly used for feature engineering, or for highlighting specific patterns in pixels. Table 5 shows statistics of different studies that used statistical algorithms for feature extraction or feature reduction in COVID-19 detection.

Albadr et al. [20] and Ali et al. [66] applied HOG and PCA for feature extraction and reduction. Albadr et al. [20] applied a variant of Cuckoo Search algorithm, while Ali et al. [66] applied an extreme learning machine approach, they were able to secure 84.67% and 97% accuracies respectively. Yousri et al. [15] applied discrete and gabor wave transformations after preprocessing, and afterward gray level co-occurrence matrix (GLCM) was computed. They were able to achieve 84.67% on dataset1 and 98.95% on dataset2. Grid search and local binary patterns (LBP) are two very important feature extraction techniques. Irmak and Emrah [12] applied grid search to automatically determine hyper-parameters of the CNN model. The best-achieved accuracy was 98.92%. Shankar et, al. [22] applied fusion-based feature extraction with a mixture of gray level run length matrix (GLRM), and local binary patterns (LBP). The artificial neural network was used for classification and 95.91% accuracy was reported. Satapathy et, al. [39] applied Otsu image thresholding technique, which separates the background and reduces the meaningless features. Feature selection was done using Cuckoo-Search-Algorithm (CSA) and segmentation evaluation was computed using Kapur Entropy. The overall accuracy of classification was 97.62%.

5. Nature-inspired algorithms for feature selection

A set of features from any dataset represents its behavior (pattern). Some features are very important for image datasets. The size, color values, intensity value, and existing shapes in the image play an important role in identifying the correct class. Image datasets have

CNN Architectures

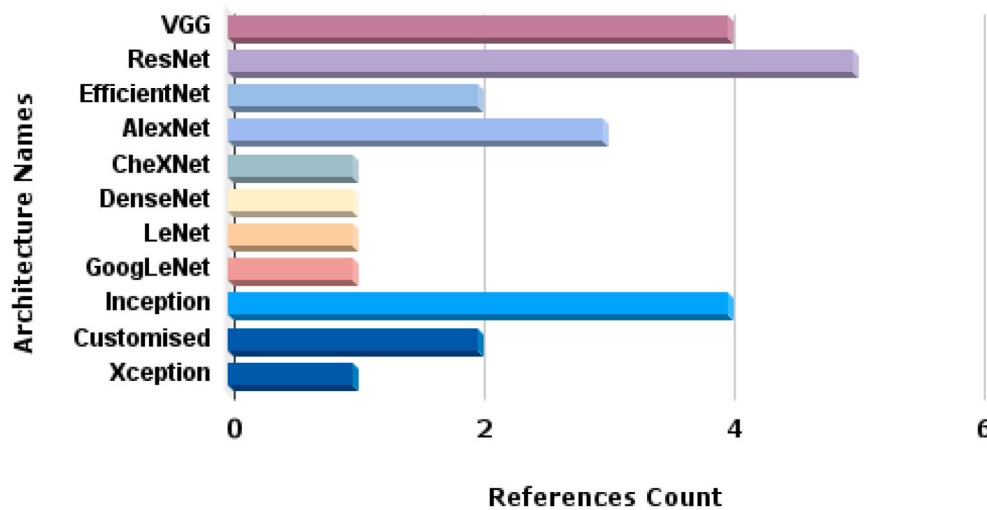


Fig. 9. CNN architectures vs. references count.

Table 5
Statistical algorithms.

Name	Algorithm
Histogram of Oriented Gradients (HOG)	It is used to mark the boundary of an object’s image, which it performs by taking gradient of image [20, 66]
Principal component analysis (PCA)	PCA is used for reducing dataset dimensions, by representing it with eigenvalues and eigenvectors [20,31,66]
Entropy	Entropy can store patterns of features with respect to the target class [67].
Wavelet Transform (WT)	This soft and hard threshold based signal processing approach is used to remove noise from image [15].
Discrete Wavelet Transform (DWT)	This form of WT discretizes signals by dividing it into sets. This optimized method reduced computational time overhead [68].
Gabor Wavelet Transform (GW)	Wavelet transformation with Gabor function that converts image to its eigen version [15].
Skewness	This method is used to figure out the distribution of image pixels [68].
Gray-level difference method (GLDM)	GLDM is mostly used for extracting texture features from images on the basis of partial derivatives of image pixels [23].
Gray Level Co-Occurrence Matrix (GLCM)	GLDM is mostly used for extracting texture features from images on the basis of correlation of image pixels [15,22]
Gray Level Run Length Matrix (GLRLM)	GLRLM can extract high order texture features, like short-run emphasis, long-run emphasis, etc. [40].
SFTA	This texture feature extraction technique consists of two steps, creating image stacks and analysis of each binary image based on boundaries, pixel count, mean gray level [67].
Grid Search	This hyper parameter tuning algorithm evaluates the model for each combination of parameters defined in grid [12].
local binary patterns (LBP)	LBP computes behaviors of each pixel with respect to the central pixel [22].
two-dimensional curvelet transformation	This transformation is used to highlight different shapes along with its direction [9].
Wiener filtering (WF)	It is used to remove image noise by using standard deviation and mean values of image [22].
Kapur Entropy	This image segmentation based method is used to evaluate the algorithm performance [39].
Otsu Image Thresholding	Automatic image thresholding technique used to separate foreground from background [39].

thousands of features due to the vast number of pixels. The quality of the image is directly proportional to the number of pixels. This factor is the major reason for increasing computation complexity. Therefore, it is very important to reduce the number of features from image data before passing it to the classification algorithm. It is also important to retain the core pattern of the class when reducing the features of an image. During feature extraction, it is necessary to select representative features. Feature selection plays an important role not only in dimension reduction but also in improving accuracy. Owing to a large number of features, the problem of feature selection can be viewed as an optimization problem. Hence, many metaheuristic-based optimizers have proven effective for this task. Details of nature-inspired algorithms used for feature selection in COVID-19 image classification are given in the following section.

Fig. 10 shows that 72% of researchers used nature-inspired algorithms for feature selection, 24% used Convolution Neural Network (CNN), and only 4% used statistical methods. Nature-inspired algorithms and statistical methods are discussed as follows. Details of CNN for image classification are discussed in section 4.1.

More detailed information about nature-inspired algorithms for feature selection is provided in Fig. 11.

Nature-inspired algorithms are those algorithms that are similar to or follow the steps of some natural phenomena. These kinds of algorithms are mostly used to resolve optimization problems. Feature selection can also be formulated as an optimization problem. Therefore, many researchers have adopted these algorithms for feature selection. Details of different nature-inspired algorithms used for feature selection and some other tasks in the context of COVID-19 detection from images are given below.

5.1. Genetic algorithm (GA)

This algorithm is based on Darwin’s theory of evolution, which states that “the fittest will survive and the rest of the generation will be destroyed” [69]. This algorithm is categorized in the following main steps: initialize population, create initial solutions (here, any function according to the problem can be defined), select the best members, perform crossover and mutation, and then select members for the next generation. These steps should be repeated until the desired population is created [40,67,69]. Fig. 12 shows a flowchart of the genetic algorithm.

Besides feature selection, GA has been used for some optimization tasks in image classification, which include hyper-parameter

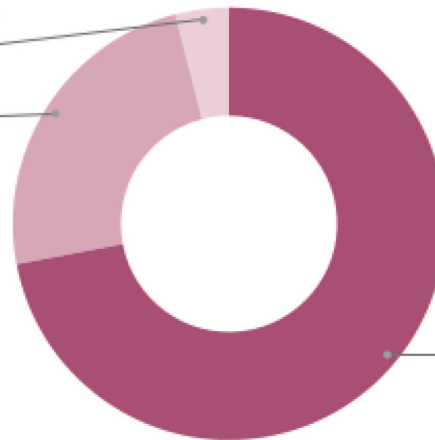
Feature Selection Algorithms

Statistical Method

4.0%

CNN

24.0%



Nature Inspired
72.0%

Fig. 10. Feature selection techniques used in image classification for COVID-19 detection.

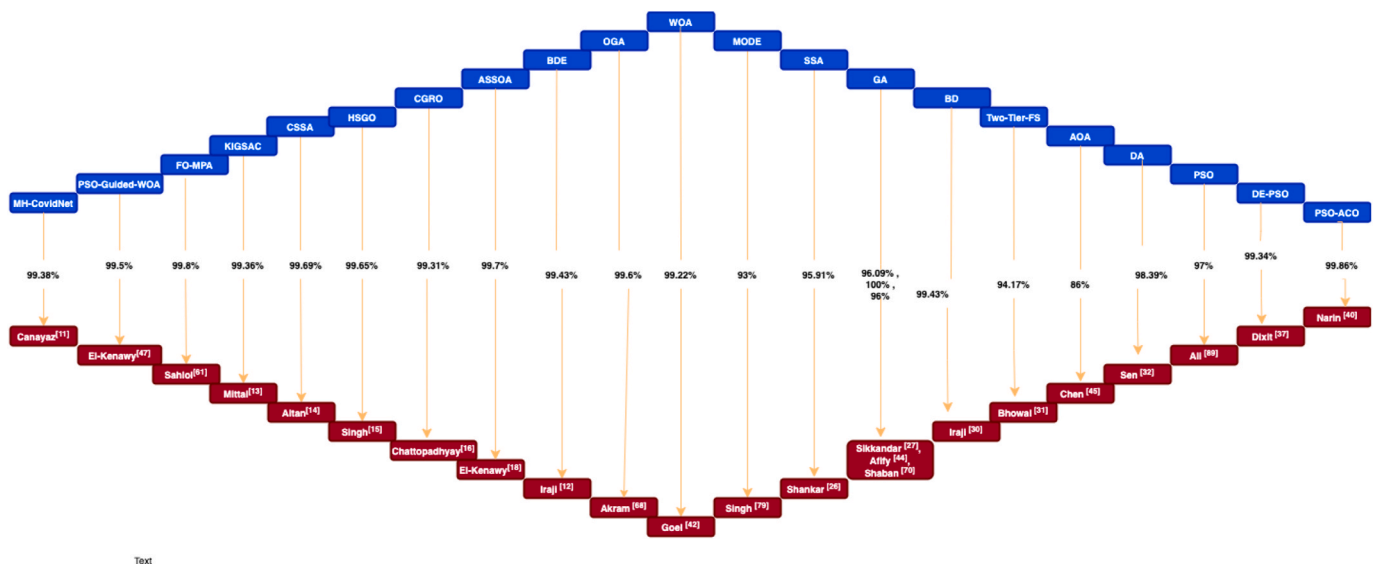


Fig. 11. Relationship between nature-inspired algorithms and applications of feature selection in image classification for COVID-19 detection.

optimization of deep learning models. A variant of GA called a multi-objective genetic algorithm (MOGA) is used for problems with a wide range of variables and a diverse set of possible solutions. Shukla et al. [14] used MOGA for hyper-parameter optimization of CNN. Another study [20] used GA with a combination of extreme learning machines and called it Optimized Genetic Algorithm-Extreme Learning Machine (OGA-ELM). ELM equipped GA with the power to avoid overfitting. Its neural network-like structure works like a kernel-based support vector machine (SVM), which helps to achieve the best performance.

COVID-19 detection from images has been reported in five studies with variants of GA. In Table 6, it can be observed that most of the studies used this algorithm for selecting features and they all have accuracy values above 90%, making it a good choice for optimization.

5.2. Whale Optimization Algorithm (WOA)

This nature-inspired algorithm [71] is similar to the hunting pattern of humpback whales. It involves three main steps: encircling prey, bubble-net attack, and searching for prey.

Encircling prey: In real-life scenarios, whales can find the location of its prey and then encircle it. However, in search space, the optimal location is unknown during its first step. In WOA, the current optimal or

selected location is encircled as the optimal location or prey.

Bubble net attack: After marking the target's location, the algorithm tries to encircle the place near the optimal solution, which is known as "shrinking encircling". The distance between the current location and new encircled points is calculated and used to update the location. This is known as "spiral updating position".

Search for prey: Whales continue exploration even during preying. Meanwhile, individual whales search its prey on the basis of the location of other fellow whales [72]. In algorithmic form, the first step is to initialize the whale population, then find the whale with the highest fitness value, and repeat it until desired results are achieved or the iteration limit is exceeded.

Some researchers use this algorithm with some additional steps to create new variants, such as Particle Swarm Optimization (PSO) guided WOA or PSO-WOA. In this modified form, another step is added after updating the solution by using WOA. It passes these solutions to PSO and then gets a newly optimized solution set. Then it updates the solutions again [43].

Table 7 shows studies that used WOA, their use cases, and their respective accuracy. It is evident from this table that WOA is mostly used in feature selection and achieved remarkable results.

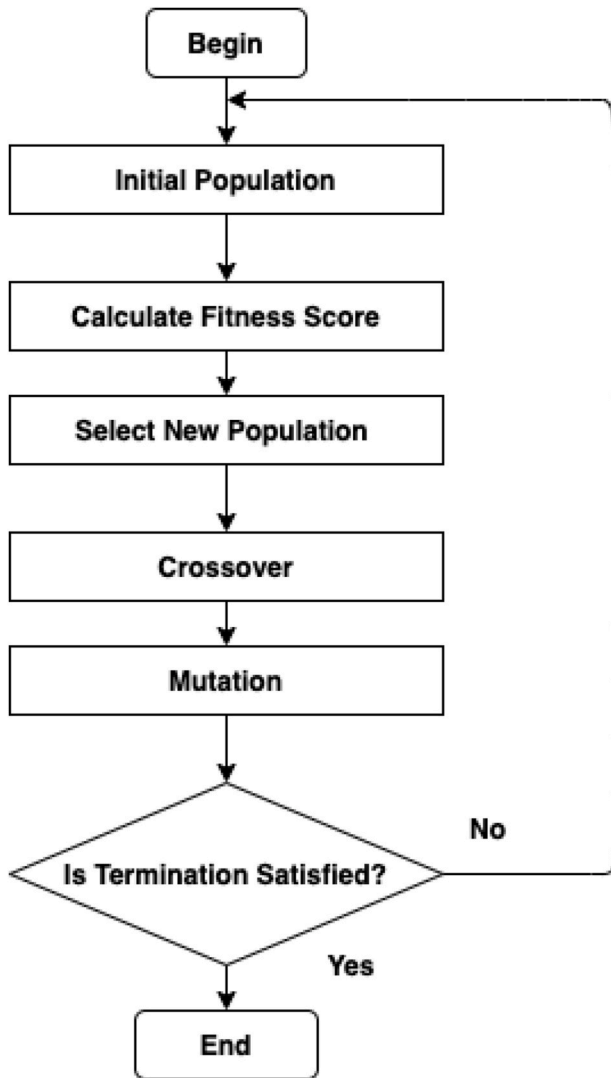


Fig. 12. Flowchart of genetic algorithm [70].

Table 6
GA variants reported in different studies.

Algorithm	Reference	Use Case	Accuracy Measure (testing accuracy)%
GA	[40,68, 69]	Feature selection	92.6, 100, 96
MOGA	[14]	Tune hyper parameter	94
OGA-ELM	[20]	Feature selection	100

Table 7
Accuracy of WOA's variants reported in different studies.

Algorithm	Reference	Use Case	Accuracy Measure (testing accuracy)%
WOA	[38,73]	Feature Selection	NA, 99.22
PSO-WOA	[43]	Feature selection	99.5

5.3. Cuckoo Search algorithm (CSA)

This algorithm [74] is inspired by the egg-laying strategy of cuckoo birds. They keep their eggs in the host's nest for hatching. For this purpose, they search for the best host. They also destroy the host's eggs to increase the probability of hatching. In terms of optimization problem, the eggs can be considered as the optimization problem and the place of eggs can serve as the solution vectors. The similarity in solutions can be measured by finding the difference between the two vectors [75].

Hanon et al. [15] used a modified version of this algorithm with the addition of Fractional Order calculus (FO), known as Fractional Order Cuckoo Search optimizer (FO-CS). This version adds the capability of storing four previous searches of cuckoos, which can help out in improving globally optimized solutions. Table 8 shows that the modified form of CS outperformed the original CS.

5.4. Differential evolution (DE)

This algorithm [76] is very similar to the GA and consists of two different steps. First, it performs mutation before crossover. Secondly, its solution searching includes information from all individuals of the population to complete the search process [77]. It consists of three steps: mutation, crossover, and selection [5]. The initial population is mostly selected randomly or by following some distribution.

DE has a variant named Multi-Objective Differential Evolution (MODE). MODE selects multiple best options. It also controls parameters such as magnification and crossover rate. MODE is more powerful than the original DE [5,78]. Another variation in Binary Differential Evolution (BDE) is that this version helps to create a map from continuous space to binary space [7].

Table 9 shows that BED performs best as compared to MODE, but it includes other factors too, e.g. main architecture, etc.

5.5. Salp swarm algorithm (SSA)

This algorithm [76] is very similar to the GA and consists of two different steps. First, it performs mutation before crossover. Secondly, its solution searching includes information from all individuals of the population to complete the search process [77]. It consists of three steps: mutation, crossover, and selection [5]. The initial population is mostly selected randomly or by following some distribution.

DE has a variant named Multi-Objective Differential Evolution (MODE). MODE selects multiple best options. It also controls parameters such as magnification and crossover rate. MODE is more powerful than the original DE [5,78]. Another variation in Binary Differential Evolution (BDE) is that this version helps to create a map from continuous space to binary space [7].

Table 10 shows the variants of SSA and respective accuracy reports by different studies on COVID-19 detection. This algorithm and its variants are used for feature selection and hyper-parameter optimization in COVID-19 detection.

5.6. Marine predators algorithm (MPA)

This is another nature-inspired algorithm that follows the natural hunting strategy between predator and prey in marine ecosystems [79].

Table 8
CS's variants reported in different studies.

Algorithm	Reference	Use Case	Accuracy Measure (testing accuracy)%
CS	[39]	Feature Selection	96.72
FO-CS	[15]	Feature selection	98.95

Table 9
DE's variants reported in different studies.

Algorithm	Reference	Use Case	Accuracy Measure (testing accuracy)%
BDE	[7]	Feature selection	99.43
MODE	[5,78]	Tune hyper parameters, Feature selection	94.0, 93.0

Table 10
SSA's variants reported in different studies.

Algorithm	Reference	Use Case	Accuracy Measure (testing accuracy)%
SSA	[22]	Feature selection	95.91
CSSA (Chaotic Salp Swarm Algorithm)	[21]	Tune hyper parameters + Feature selection	99.69

Similar to most of the nature-inspired algorithms, it initializes the first population randomly in a range of 0–1. Afterward, the fitness of all candidates is calculated, and the fittest one is selected (top predator). As a result, a matrix of predators is formed in this way. Another matrix of the same dimension named “prey”, is used by predators to update the position. Predators try to find the best location to search for their food [67].

Its variant Fractional Order Marine Predators Algorithm (FO-MPA), combines the memory storage support of fractional order methods to store previous location information [63]. Another variant is the Improved Marine Predators Algorithm (IMPA), which includes a Ranking-based Diversity Reduction Technique for improving the search. This technique keeps track of consecutive iterations where no particle can find the best solution [67].

Table 11 shows different versions of MPA and its reported accuracy. This algorithm has been used for feature selection in COVID-19 image datasets. The accuracy value is written as “NA” in Table 11 for IMPA, as this study [67] did not classify the images but only applied image segmentation.

5.7. Flower pollination algorithm (FPA)

This algorithm [80] is modeled after the pollination process of flowers. Meanwhile, the position of a pollen particle can be considered as a solution vector. It has two parts, namely local pollination and global pollination. Chakraborty et al. [81] proposed a modified form of FPA by mixing type 2 fuzzy system and named it Superpixel based Fuzzy Modified Flower Pollination Algorithm (SuFMoFPA). This modification improves the computation by selecting and processing the most relevant pixel of the image and ignoring the irrelevant pixels. It can be used for feature selection in COVID-19 detection.

5.8. Slime mold algorithm (SMA)

This algorithm [82] is inspired by the food collection pattern of yellow-colored slime mold. The micro-organism is used to create connectivity from the current position to food or host by making an optimal link. Searching for the best path for food can be mapped to solution

Table 11
MPA's variants reported in different studies. NA mean Not applicable.

Algorithm	Reference	Use Case	Accuracy Measure (testing accuracy)%
IMPA	[67]	Feature selection	NA
FO-MPA	[63]	Feature selection	99.8

vectors. This solution can be computed by calculating the positive and negative feedback of the progress toward the destination. Compared to other algorithms, this algorithm has been used for feature selection [73] and has achieved the highest Kapur's entropy.

5.9. Social group optimization (SGO)

It is a very common behavior observed in humans that their problem-solving skills improve when working in groups compared to their individual performance. The social group optimization algorithm [83] is based on the social behavior of problem-solving. Their population consists of feature vectors, and each feature vector presents a unique person. Each feature vector needs to be optimized to get the best solution. Additionally, this algorithm has a variant called Hybrid Social Group Optimization (HSGO). To improve the SGO, this variant adds a wrapper-based approach. This algorithm includes six main steps: population initialization, calculating fitness of each member, separating best and worst candidates, performing mutation, an improving phase where each person learns from previous experience, and an acquiring phase used to acquire knowledge from other individuals.

According to the reported results, HSGO outperformed the original SGO that was used for feature selection [10].

5.10. Emperor penguin optimizer (EPO)

This metaheuristic-based optimizer [84] is inspired by the huddling behavior of emperor penguins, which they show during forging. They move in groups with the help of fast movers of the swarm. Therefore, in a mathematical model, this algorithm aims to find the quickest mover. Its modified version [17], called Multi-Objective Emperor Penguin Optimizer (MOEPO) includes archive and grid mechanisms. In addition, it also performs an update step at the group level, which makes it more efficient than the simple EPO. MOEPO [17] has been used for feature selection in COVID-19 detection.

5.11. Gray wolf optimization (GWO)

This algorithm [85] is inspired by the gray wolf's hunting policies and the leadership role. Gray wolves divide the population into four levels: alpha, beta, delta, and omega. Alphas are the leaders that make decisions about hunting, living and moving from one place to another. Beta assists the alpha group, while Delta can simply rule the omega. Omega is the last class that has to obey leaders (Alphas). They follow four steps: hunting, searching, encircling, and then attacking the prey. The searching task is done by following the location of alpha, beta, and delta. This algorithm [85] has been used for feature selection in COVID-19 detection.

5.12. Squirrel search optimization algorithm (SSOA)

This algorithm is inspired by the food search mechanism of squirrels. Squirrels divide all trees into three types, namely normal, oak, and hickory. Normal trees are not useful to them, while, oak and hickory are used for food and making nests. Mathematically, it considers search space as a matrix of squirrels that are flying to search for food. They search for hickory trees and oak trees. A modified version of the algorithm [13] assumes that squirrels search for one suitable hickory tree and three suitable oak trees. It also expands the movement space. This is known as Advanced Squirrel Search Optimization Algorithm (ASSOA). This was used for the feature selection step and provided better results compared to the original SSOA.

6. Nature-inspired algorithms for parameters and architectural optimization

The computational cost for image classification is very high, as the

feature set of images is usually very large compared to the text data. This can be optimized using nature-inspired algorithms at different stages. It is mostly considered as part of feature selection (feature reduction). A few researchers have improved the results by optimizing hyper-parameters or weights of the main algorithm. The following subsections talk about how nature-inspired algorithms are used for the optimization of parameters of machine learning and deep learning models.

6.1. Parameter optimization of machine learning algorithms

Chimp Optimization Algorithm (ChOA) [86] is inspired by the hunting procedure of chimps. Chimps can be categorized into four types: driver, chaser, barrier, and attackers. The chimps' population can be presented as all possible solution vectors are divided into four categories. Each solution will act according to the properties of the category associated with it. All four types will mark the position of the prey or the best solution. A deep convolution neural network (CNN) is used to extract the features, and the ChOA is applied to optimize weights and biases for extreme learning machines (ELM) classification [35].

Support vector machine (SVM) is a machine learning algorithm (also called a maximum margin classifier). SVM uses kernel functions for solving non-linear classification problems. The performance of the SVM classifier heavily depends on the selection of kernel parameters. Due to the huge search space, the optimal selection of these parameters is time-consuming. Hanon et al. [34] proposed a whale optimizer-based improved partial swarm optimization algorithm (WO-IPSO) to optimize the parameters of SVM.

6.2. Architecture (hyper-parameters) optimization of deep learning models

CNN is the most commonly used method for feature extraction in image classification tasks, and about 54% of studies reported in this survey have adopted CNN. CNN has millions of parameters that can increase the computational cost. This cost can be reduced by using metaheuristic optimization algorithms. Three studies by Hamdy et al. [30], Júnior et al. [32], Hanon et al. [34] used PSO for parameter optimization.

The first CNN-based solution for COVID classification was reported by Irmak et al. [12]. The main contribution of this paper was to use a large dataset of COVID with 1524 COVID-19 images and also deal with a large number of hyperparameters associated with CNN architecture using grid search. Irmak et al. adopted two different models to classify the images. In the first stage, a model was used to detect whether the image is COVID or non-COVID. In the second stage, the specific class of the image was determined. This paper deals with a three-class classification problem and achieves a 98.27% average accuracy.

According to Kaur et al./cite [bib28], although CNN is widely used for image processing, it suffers from hyper-parameter tuning issues due to a large number of parameters. The authors considered the following four classes: COVID, healthy, pneumonia, and tuberculosis, and proposed a modified approach. The AlexNet architecture was adopted, and hyper-parameters of this network were tuned using Strength Pareto Evolutionary Algorithm-II (SPEA-II). The final layer of architecture was used as a classifier. Meanwhile, the proposed solution was able to achieve an accuracy of 99.97%.

Singh et al. [5] focused on the same issue of hyper-parameter tuning of CNN using the same pipeline. However, this study worked with a binary class (COVID, non-COVID) classification problem and used multi-objective adaptive differential evolution (MADE) for tuning. The experimental results were compared to a set of machine learning algorithms. The proposed approach outperformed all other algorithms and obtained an accuracy of 94.48%.

Shukla et al. [14] dealt with a four-class classification problem and followed a very similar pipeline as discussed above. However, this paper

used the GoogLeNet architecture of CNN and tuned the hyper-parameters of this model using a multi-objective genetic algorithm (MOGA). The experimental results using 20-fold cross-validation were presented and discussed. The proposed GA-based method was able to obtain results with a 94.93% accuracy.

Dhiman et al. [17] applied eleven deep learning architectures (AlexNet, VGG16, VGG19, GoogleNet, ResNet18, ResNet50, ResNet101, InceptionV3, InceptionResNetV2, DenseNet201 and XceptionNet) and performed binary class classification. Furthermore, this study optimizes parameters (weights) of CNN using multi-objective emperor penguin optimizer (MOEPO). The best accuracy is reported with ResNet101.

Hamdy et al. [30] and Júnior et al. [32] have used particle swarm optimization (PSO) to solve image classification problem for COVID-19 diagnosis. Hamdy et al. [30] worked with CT images, and solved a binary class problem by taking into account 1050 COVID-19 and 1500 normal images. The CNN's hyper-parameters were tuned using PSO and obtained an accuracy of 98.04%. Júnior et al. [32] also experimented with a binary classification problem. However, an X-ray dataset was used with only 206 COVID images, which were very few compared to Ref. [30]. Unlike [30], Júnior et al. proposed a different algorithm for classification called eXtreme Gradient Boosting (XGBoost). The hyper-parameters of this suggested that the algorithms were tuned using PSO. The reported accuracy of this method was 98.71%. Although the nature of images was different in both studies, the approaches are quite similar (both use PSO for hyper-parameter tuning), hence the reported accuracies are very close.

Goel et al. [64] suggest Gray Wolf Optimizer (GWO) for the COVID-19 detection using medical images. The proposed study handles a three-class classification problem of X-ray images. A CNN was used with a customized architecture. In this study, the hyper-parameter of CNN was tuned using GWO and the last layer was used as a classifier. This suggested method reported results with an accuracy of 97.78%. Another research work conducted by Goel et al. [38] provides a solution for a binary classification problem for COVID-19. The suggested model was trained on CT images using a generative adversarial network (GAN), and the hyper-parameters were optimized by Whale Optimization Algorithm (WOA).

More detailed information about meta-heuristics algorithms for hyper-parameter optimization of deep learning architectures is provided in Fig. 13.

7. Classification

Classification of images into different classes is the last step after preprocessing, feature extraction, and feature selection. There are two types of classification problems: (i) binary class classification and (ii) multi-class classification. According to Fig. 14, 30 studies used binary class classification, 14 studies used three classes, and only two studies used four classes. There is a bar in Fig. 14 with label zero, which shows the count of papers that used metaheuristics for only feature selection. Binary classification is simpler than multi-class classification. As a result, most studies have only dealt with binary classification tasks.

Unlike the statistical model, machine learning (ML) algorithms learn from data. These algorithms extract patterns from data and yield a function that represents the behavior of the dataset. The learning is evolved by increasing learning examples. The following section discusses the different classification methods used in literature for COVID-19 detection from images.

7.1. Fully connected neural networks

Neural Network (NN) structure is inspired by biological neurons of the brain. It consists of connections of units and has two main layers, namely input, and output. In deep neural networks, many hidden layers are added to enhance the performance. For each unit, it needs to find a

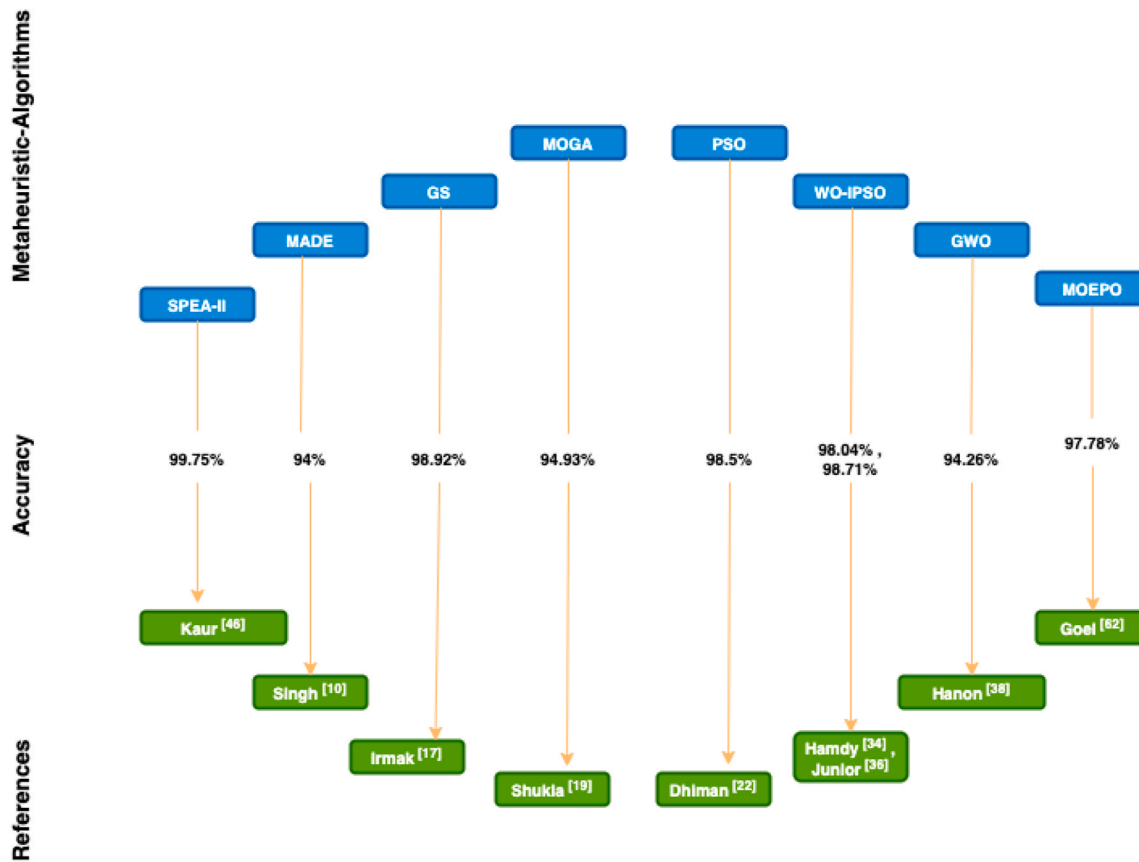


Fig. 13. Relationship between meta-heuristics algorithms and applications of hyper-parameter optimization in image classification for COVID-19 detection.

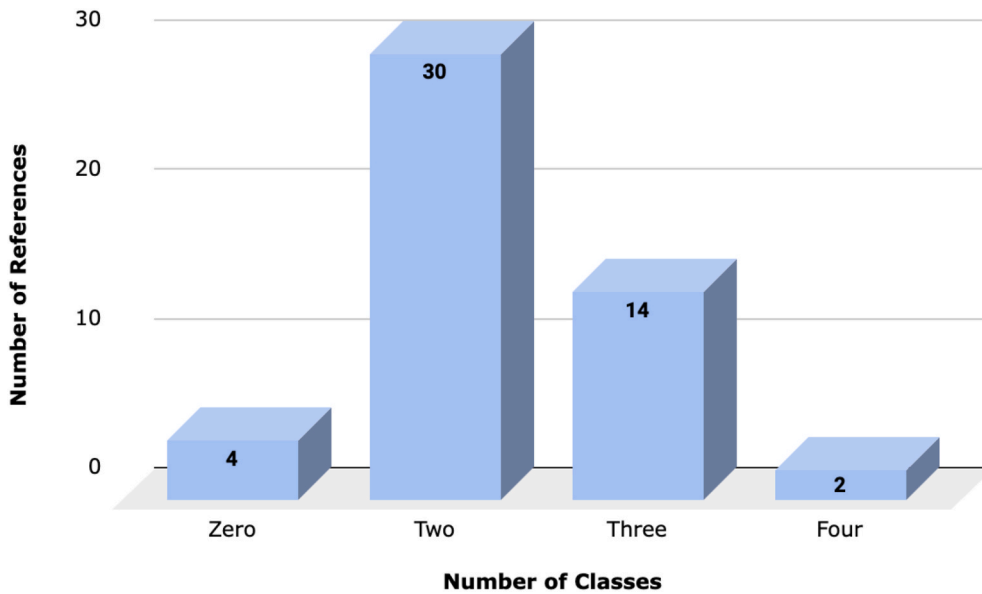


Fig. 14. Number of class VS Number references.

suitable weight vector. The forward pass is used to make the prediction, a loss is calculated for the training set, and backpropagation is used to update the weights to minimize loss or error. A popular approach for optimizing weights in backpropagation is a gradient descent algorithm [87]. Several hidden layers can be added and this version of NN is called Multi-Layer Neural Network (MLNN) or Multi-Layer Perceptron (MLP). Many studies [9,12,13,18,22,27] have used fully connected neural

networks for classification and have achieved more than 90% accuracy on test data.

Kaur et al. [42] adopted the fully connected layer as a classifier. In this study, AlexNet was used and hyper-parameters were optimized using SPEA-II. This paper solved a four-class problem with an accuracy of 99.75%. Singh et al. [5] optimized the parameters of CNN using a nature-inspired algorithm MADE. This paper focuses on the binary class

problem and achieves a 94% accuracy. Hamdy et al. [30] suggest PSO for optimizing the hyper-parameters of CNN. The X-ray image data with binary classification is adopted.

Irmak and Emrah [12] used a CNN to solve the COVID-19 binary class classification problem using X-ray images. A set of experiments were performed using a mixture of datasets 2, 5, 9, and 10. The feature extraction and selection were performed using CNN, while the parameters (mini-batch size, learning rate, momentum, and regularization) were optimized with help of GS. The classification was done using a fully connected layer. This proposed method was able to achieve maximum accuracy of 98.92%. Shukla et al. [14] suggested that (CNN-GoogLeNet) the hyper-parameters were optimized with MOGA. It worked on a binary classification problem. The mixture of datasets 2 and 11 with X-ray images was adopted. Test train split was 70%–30% and this method produced results with an accuracy of 94.93%.

Chowdhury et al. [18] suggested CNN with CheXNet and DenseNet201 for feature extraction and selection steps. Image normalization was performed using image resizing, and image augmentation was done using rotation and translation. The final experiment was performed for both augmented and original data. This study deals with the three-class problem. The proposed solution was investigated with a mixture of six X-ray datasets (datasets 2, 3, 5, 6, 15, and 16). The classification was performed with a fully connected layer of neural network. The reported accuracies with and without augmentation were 97.94%, and 97.74%, respectively. Goel et al. [38] proposed CNN-ResNet50, and its hyper-parameters were optimized using WOA. It was used to perform binary classification for CT images (dataset 8). Feature extraction was performed using GAN, and a three-class COVID-19 classification problem was solved. Meanwhile, the reported accuracy was 99.22%. Bhowal et al. [27] investigated three different deep learning architectures, namely VGG16, Xception, and InceptionV3. This work performed experiments using X-ray images from datasets 2, 5, and 22. The feature selection step was completed using a metaheuristic-based Two-Tier-FS-Coalition game. The last fully-connected layer performed a three-class classification with a 94.27% accuracy.

Altan et al. [9] worked with the EfficientNet-B0 architecture of CNN and used Chaotic Salp Swarm Algorithm (CSSA) for feature selection. This proposed method was benchmarked against the X-ray image dataset (dataset 2). The suggested approach was able to achieve 99.69% accuracy for the three-class problem. Goel et al. [23] worked on X-ray images and a mixture of two datasets 2 and 3 were adopted. Nevertheless, the VGG16 architecture of CNN was suggested. Feature extraction and selection were performed using the gray-level difference method, while a fully connected layer of neural network was used for the classification. The maximum reported testing accuracy was 96.09%.

7.2. Naive Bayes (NB)

This probabilistic algorithm is based on Bayes' theorem. It creates the assumption of independence (i.e., the behavior of any feature is independent of all other features). Due to this assumption, it is called Naive Bayes. This algorithm calculates probabilities of attributes given in a class (e.g., the probability of occurring urgency words in spam email is high compared to the normal email). Tallha et al. [68] worked on a binary class detection problem and experimented with an X-ray dataset. Feature extraction was performed using discrete wavelet transform (DWT) and segmentation-based fractal texture analysis (SFTA), while feature selection was performed using a metaheuristic-based optimized genetic algorithm (OGA). The last step of classification was performed using NB, achieving a 92.6% accuracy.

7.3. Support vector machine (SVM)

It is a supervised algorithm that performs regression analysis and tries to find a vector or line that can separate two classes. The record (example vector) that decides the boundary between two classes is

called a support vector. It performs best for the linearly separable binary class dataset. It can be improved for non-linear datasets by applying a kernel function, which attempts to transform the data into a linearly separable form. This algorithm not only finds a separating line but also searches for it by maintaining the maximum margin between two classes.

Twelve studies [6,43,10, 11,16,7,26,28,29,38,33,36], and [34] applied SVM for classification. These papers adopt metaheuristic algorithm MH-CovidNet, PSO-WOA, HSGO, CGRO, BDE, BD, DF, PSO, PSO, DE-PSO, PSO-ACO, and WO-IPSO for feature selection step, respectively. For the last classification step, they prefer to use SVM. Among all [6,7, 26], and [36] work with multi-class, while [10,11,28,29,31,33,34,43] deal with binary class classification problem. All studies report accuracy above 90%.

Canayaz and Murat [6] have adopted X-ray images of three classes labeled as normal, COVID, and pneumonia patients. The main algorithm was CNN (VGG19) and in the classification pipeline, binary PSO and binary GWO were used for feature selection. These selected features were classified using SVM. It was able to achieve an accuracy of 99.38%.

El-Kenawy et al. [43] proposed a binary classification system build on CT chest images. AlexNet architecture of CNN was used as the main method. CNN's first layer was used as a feature extractor. Afterward, guided WO based on Stochastic Fractal Search (SFS-Guided WOA) was used as a feature selector. These selected features were passed to the SVM classifier. The best-reported testing accuracy was 99.5%. Singh et al. [10] also provided a binary class classifier trained on chest X-rays, of which HSGO worked as a feature selector. These selected features were tried with multiple classifiers, and SVM outperformed with an accuracy of 99.65%.

Chattopadhyay et al. [11] adopted ResNet18 (CNN) for feature extraction and then the extracted features were passed to the CGRO for selecting the most relevant features. After selecting a subset of the important features, SVM was used to perform the classification. This model was tested on both X-ray and CT images. The experiments were performed on SARS-COV-2 dataset, Chest X-Ray, and CT-dataset and obtained results with 98.65%, 99.44%, and 99.31% accuracies, respectively.

Iraji et al. [7] applied CNN and extracted deep features from X-ray images, and then the feature selection was performed using BDE. The final features were classified using SVM, and this three-class system was able to achieve a 99.43% accuracy. Iraji et al. [26] modified their previous work for new datasets. They used a very similar pipeline to Ref. [7] and dealt with a three-class classification problem. Features were extracted using CNN, then BD was used for the selection step, and the final set of the feature was sent to SVM for classification. The main difference in both approaches was the datasets [7] has worked with dataset 1, while [26] has worked with datasets 2 and 9.

Sen et al. [28] worked with CT images for binary classification. Feature extraction was performed using CNN, and selection was done using DF. These feature vectors were classified using SVM. This approach was tested on datasets 8 and 21, and thus the reported accuracies were 98.39% and 90.0%, respectively.

Mohammed et al. [29] adopted PSO for both feature extraction and feature selection. Then the final feature set was passed to SVM, which performed binary classification on the chest X-ray dataset with a 98.04% average accuracy. Asghar et al. [31] also used PSO for both feature extraction and feature selection and then performed binary classification using SVM. The experiments were performed on X-ray dataset 2 and the reported accuracy of this system was 99.81%. Narin and Ali [36] also used PSO, but it only worked with CNN. Meanwhile, PSO served as a feature selector, and SVM was trained for classification. The three-class classification was performed on X-ray images. It was implemented on dataset 2 and was able to yield a 99.83% accuracy.

Dixit et al. [33] suggested a different pipeline by adding K-mean clustering as a preprocessing step. Moreover, PSO and DE are used for feature selection and an SVM classifier is trained. The proposed method

was benchmarked against X-ray dataset 2 and achieved a 99.34% accuracy.

Hanon et al. [34] proposed an improved PSO (IPSO) in order to optimize the parameters of an SVM kernel function. The experiments were performed on X-ray images and the trained binary-class classifier outperformed other compared approaches with an accuracy of 94.26%.

7.4. *K-nearest neighbor (KNN)*

This data-driven nonparametric classifier is best known for its ability to handle multi-class classification problems. It is very simple, yet very effective. This algorithm finds k to be the closest objects to a given test set object and assigns the test set object to the majority of class out of its neighbors. The main drawback of KNN is its high memory consumption and its sensitivity toward outliers. Few studies by Afify et al. [40] and Shaban et al. [69] have used it as the last step of the classification pipeline (COVID-19 classification), and also adopted nature-inspired algorithm GA for feature selection. One study [69] tried to reduce the drawback of sensitivity toward outliers by introducing a heuristic-based approach to decide. This new variant is called Enhanced K-Nearest Neighbor (EKNN). EKNN [69] not only decides on the basis of voting of neighbors but also adds the degree of membership of points with its class. Meanwhile, both studies [40,69] have reported accuracies above 95%.

7.5. *K-means*

This unsupervised algorithm assigns points to a cluster on the basis of its closeness with the center of the cluster. The center of each cluster mainly represents its cluster. This non-parametric algorithm simply takes the value of k (number of clusters) as its input based on the following steps. Select k random points as centers, then compute the difference between all data points and centers. Each point is assigned to its closest cluster center. The cluster centers are recomputed for each iteration. This process is repeated until the cluster centers stop changing. Mittal et al. [8] and Chakraborty et al. [81] adopted this process for the classification of COVID-19 images. The work [8] improved the performance of K-mean by adding gravitational search optimizer and applied K-mean after completing the pipeline of CNN. This solution reported a 64% and 99.36% accuracy for CT scans and Ultrasounds, respectively. Chakraborty and Mali [81] proposed a superpixel-based fuzzy modified flower pollination algorithm for better feature selection and K-means for classification. The proposed method outperformed other compared methods.

7.6. *Decision tree (DT)*

DT is a supervised classifier that works on the basis of rules that are created using data patterns. It contains a root node that is population-representative, a decision node that divides the next nodes, and a leaf node (the last node or class label). Initially, DT considers all the features as roots and finds the best attribute of the dataset by computing some measures, and then takes a decision of partition. This process ends after reaching leaf nodes. One of its versions, J48 [78], adds some extra features, such as pruning trees of less value and derivation of rules, thus reducing the impurity of data. This study J48 was used at the end of CNN's pipeline, and the parameter of this CNN was tuned using MOEPO. The proposed method classified the test data with an accuracy of 98.50%.

8. Discussion

In this survey, we have performed a detailed review of metaheuristics-based image classification specific to the COVID-19 classification task. Several perspectives have been discussed in the published literature, ranging from feature extraction to feature

reduction, as well as feature selection and hyperparameter tuning and classification techniques. Although all studies cited in this survey have adopted metaheuristics in their research, different studies used these algorithms at different stages of image classification. Two major use cases of metaheuristics for COVID-19 image classification are feature selection and hyperparameter optimization. Most studies used 23 publicly available datasets discussed in this survey. The following subsections talk about the results on different datasets and use cases of metaheuristic-based algorithms.

8.1. *Datasets and reported accuracies*

In this survey, the following three types of datasets were observed: (i) X-ray images, (ii) CT scans, and (iii) Ultrasounds. Moreover, some studies also use both X-rays and CT scans. The following paragraphs talk about the accuracy scores on different datasets and the experimentation behind them.

Table shows the accuracies and number of studies on different datasets for COVID-19 image classification. According to the table, Dataset 2 is very popular among researchers and was adopted in 25 studies. This dataset contains 468 COVID-19 images, while 178 belong to other 19 diseases, including 59 unknown images. Albadr et al. [20] reported the highest accuracy on this dataset as 100%. The accuracy is perfect, but they used less than 200 X-ray images. The model was trained using Optimized Genetic Algorithm-Extreme Learning Machine (OGA-ELM), and the data was preprocessed by PCA and HOG feature extraction methods. Since the majority of images belong to the COVID-19 class, this accuracy can be a result of an over-fitted model.

The second most popular dataset is Dataset 5, which was used in six studies. The highest accuracy for this dataset is 99.86% [36]. CNN architecture was used along with PSO-ACO for feature selection. This dataset has 219 COVID-19 images, 1345 normal, and 1341 images from pneumonia patients. The COVID-19 images have a smaller count compared to the other classes.

Table 12 shows odd results, as reported by Mittal et al. [8]. They experimented with four different datasets, namely Dataset 2, Dataset 3, Dataset 4, and Dataset 21. Out of these four datasets, two were X-rays, while Dataset 4 and Dataset 21 were CT scans and ultrasounds, respectively. The approach used by this paper was CNN and improved KIGSA-C for classification. This paper reported accuracies of 98.30% for the mixture of Dataset 2 and Dataset 3, while 99.36% for Dataset 21 and only 64.41% on Dataset 4 (CT-scan images). These results show that, although the approach proposed by this research is useful for X-rays and ultrasounds, it is not effective for CT chest images.

A number of studies have experimented with diverse datasets. One study [10] experimented with three datasets, some ([12,15]) used four datasets, and one study [18] used six different datasets in their experiments. Muhammad et al. [18] used a mixture of six datasets and trained the model with 423 COVID-19, 1579 normal chest X-rays, and 1485 viral pneumonia images. They also used data augmentation techniques to remove the problem causing COVID-19 class imbalance. The reported accuracy was 97%+. Although many studies have used imbalanced datasets, some have used a mixture of multiple datasets as shown in the table. Therefore, we can conclude that there is still room for improvement with balanced datasets (balancing the COVID-19 class by mixing multiple datasets).

8.2. *Use cases of metaheuristic-based algorithms*

The selection of important features not only plays a critical role in achieving high accuracy in image classification but also helps to reduce the computational cost. Metaheuristics have shown to be very useful for feature selection tasks in the literature. Table 13 shows the use case of nature-inspired algorithms for feature selection and parameter optimization in COVID-19 image classification. It is evident from Table 13 that most of the researchers prefer to perform feature selection using nature-

Table 12
Datasets and reported testing accuracies.

Dataset	Data Type	References	Accuracy %
Dataset1	X-ray	[6]	99.38
		[7]	99.43
Dataset2	X-ray and CT-Scan	[8]	99.36
		[9]	99.69
		[10]	99.65
		[11]	99.44
		[12]	98.92
		[13]	99.7
		[14]	94.93
		[15]	84.67
		[16]	98
		[17]	98.5
		[18]	97.74
		[19]	97.6
		[20]	100
		[22]	95.91
		[23]	96.09
		[70]	98.7
		[71]	99.43
		[75]	94.17
		[30]	98.04
		[31]	99.81
[32]	98.71		
[33]	99.34		
[36]	94.26		
Dataset3	X-ray	[8]	99.36
		[17]	98.5
		[18]	97.94
Dataset4	CT-Scans	[8]	64.41
Dataset5	X-ray	[10]	99.65
		[12]	98.92
		[18]	97.94
		[24]	98.7
		[27]	94.17
		[36]	99.86
Dataset6	X-ray and CT-Scan	[10]	99.65
		[15]	84.67
		[18]	97.94
Dataset7	CT-Scan	[11]	99.44
Dataset8	CT-Scan	[37]	99.99
		[38]	99.22
		[28]	98.39
Dataset9	X-ray	[12]	98.92
		[13]	99.7
		[26]	99.43
		[32]	98.71
		[12]	98.92
Dataset10	X-ray	[12]	98.92
Dataset11	CT-Scan	[14]	94.93
Dataset12	X-ray	[15]	84.67
Dataset13	X-ray	[15]	84.67
Dataset14	CT-Scan	[39]	97.62
Dataset15	X-ray	[18]	97.94
Dataset16	X-ray and CT-Scan	[18]	97.94
Dataset17	X-ray and CT-Scan	[19]	97.6
Dataset18	X-ray	[19]	97.6
		[33]	99.34
		[35]	99.11
Dataset20	X-ray	[35]	98.25
Dataset21	CT-Scan	[8]	99.36
		[40]	100
		[28]	90
Dataset22	X-ray	[75]	94.17
Dataset23	X-ray	[29]	98.04

inspired metaheuristic algorithms. In the use case of feature selection (FS), the highest accuracy was 100% [40]. The work [40] performed the image segmentation step, and then selected 64 most relevant features using GA. After performing high-level filtering, the data is classified using KNN. KNN all yields accuracy, precision, recall, and F-score of 100%, while the decision tree is classified with an accuracy of 95%. These accuracy scores are very impressive, but they used a very small amount of data. As mentioned in the datasets section, they used only 200 CT images. Therefore, we can infer that this accuracy might be due to the

over-fitting of the model on a small dataset.

The second highest accuracy of 99.5% with 99% F-score was reported by Ref. [63]. The proposed study used fractional-order and marine predators (FO-MPA) for feature selection. Meanwhile, the experiments were performed on two different datasets. One of the datasets was the X-ray images published by Kaggle, which is comprised of 200 COVID-19 images and 1675 negative images. The second dataset was collected from Qatar University and the University of Dhaka, including collaborators from Pakistan. This data consists of 219 positive X-ray images and 1341 X-ray images. The proposed method was compared with WOA, HGSO, SCA, SMA, PSO, GWO, HHO, GA, and MPA. The proposed algorithm outperforms all others on both datasets. Therefore, it can be concluded that the feature selection algorithm in this paper seems to be the most suitable for this task.

The second most common use case of metaheuristics is hyperparameter tuning. The best result was reported by Emrah [12] using GS for tuning the hyperparameter of CNN. This research used 1524 X-ray images of COVID-19, pneumonia, and normal patients.

Many studies ([13,15,30], and [39]) in Table 13 have reported very high accuracy scores. However, they did not mention the other measures, such as precision, recall, and F-score. There are high chances that these high accuracy scores are due to class bias or model bias.

Nature-inspired algorithms are computationally expensive; therefore, it is very important to decide which step addition of these optimizing algorithms can provide a good aid. This survey can help researchers in making these kinds of decisions. Further details about use cases of metaheuristics in the context of COVID-19 image classification can be explored in the references reported in Table 13.

9. Future directions

This survey has performed an extensive review of metaheuristic approaches for COVID-19 image classification. Although the reported accuracies are high, there are still some research gaps. The following subsection discusses some future research directions in COVID-19 image classification research.

9.1. Multi-class classification

Binary classification is easier to handle. This is the major reason why almost 60% of studies dealt with the binary classification problem. Therefore, a research gap exists for multi-class classification. Selecting multiple classes of viral lungs diseases along with COVID-19 images can be a very interesting and useful topic of research.

9.2. Dataset difficulty

The reported accuracies of the reviewed studies are very high and are above 95%. The reason for such high accuracies is the nature of the dataset. Most datasets have two classes, where one class has COVID-19 affected lung images and another has lung images of healthy people. While it is relatively easy to determine the health of COVID-19 infected lungs, it is a more challenging and useful task to identify if the lung images are healthy or if there is another lung disease, such as pneumonia. A possible future research direction is the creation of challenging datasets that are more helpful for building useful classifiers in the medical domain.

9.3. Feature enriched images

In medical image processing, especially for COVID-19 chest images, there are two major types of datasets. One is **x-ray** images (a grayscale, 2D image with lesser number of features), and the other is **CT-Scan** (a colored, 3D image with 360 views, covering all minor details). CT scans are very clear and feature-enriched data sources. However, it can be observed from previous studies that only 20% of studies had explored

Table 13
Use cases of Nature Inspired Algorithms and Reported Accuracies.

Ref.	Nature Inspired Algorithm	Use case	Accuracy %	Precision %	Recall %	F-Score %	
[6]	MH-CovidNet	Feature Selection	99.38	100	–	99.07	
[43]	PSO-Guided-WOA		99.5	100	100	–	
[63]	FO-MPA		99.8	–	–	99	
[8]	KIGSA-C		99.36	100	100	99.37	
[9]	CSSA		99.69	99.62	99.44	99.53	
[10]	HSGO		99.65	99.66	99.65	99.65	
[11]	CGRO		99.31	99	100	98	
[13]	ASSOA		99.7	–	–	–	
[7]	BDE		99.43	99.16	99.57	–	
[68]	OGA		92.6	92.5	92.5	–	
[38]	WOA		99.22	97.78	99.78	98.79	
[78]	MODE		93	91	90.5	90	
[22]	SSA		95.91	95.97	95.1	95.29	
[23]	GA		96.09	93.85	91.54	90	
[40]	GA		100	100	100	100	
[69]	GA		96	76	64	–	
[26]	BD		99.43	99.57	99.16	–	
[27]	Two-Tier-FS -Coalition game		94.17	87	98	–	
[41]	AOA		86	96	96	96	
[28]	DA		98.39	98.21	97.78	98	
[66]	PSO		97	95.44	100	–	
[33]	DE-PSO		99.34	–	–	99.3	
[36]	PSO-ACO		99.86	100	98.17	99.08	
[16]	MV		98	100	100	100	
[42]	SPEA-II		Tune Hyper-parameters	99.75	99.97	99.89	99.89
[5]	MADE		94	93.41	94.86	95.37	
[12]	GS		98.92	98.72	98.72	–	
[14]	MOGA	94.93	98.29	97.64	98.31		
[17]	MOEPO	98.5	100	100	98.4		
[30]	PSO	98.04	–	–	–		
[32]	PSO	98.71	98.89	99.63	99.25		
[34]	WO-IPSO	94.26	85.91	84.9	85.4		
[64]	GWO	97.78	96.25	97.75	92.88		
[15]	FO-CS	Feature Extraction, Feature Selection	98.95	–	–	–	
[29]	PSO	98.04	91.67	100	–		
[31]	PSO	99.81	–	88	98		
[20]	OGA-ELM	Feature Selection, Classification	100	100	97.37	98.67	
[21]	CSSA	99.69	99.62	99.44	99.53		
[39]	CSA	Feature Extraction, Feature Selection, Classification	97.62	–	–	–	
[24]	GNRCS	98.70	97.6	97.9	97.8		

CT-scan images, while the rest of the studies focused on X-ray data or experimented with both types. The CT images are very feature-enriched, and thus they require more processing to train any model. A possible future direction is by reducing the image parameters at different levels of image processing. Metaheuristics can be useful for reducing image features at different levels.

9.4. Noisy data handling

The X-ray screening process passes some radiation through the human body. Some of the radiations are scattered during this process, and this sometimes produces noise on X-ray images. Three types of noises are observed, namely Poisson noise, salt and pepper noise, and speckle noise. The Poisson noise is due to the uneven spread of X-rays on the surface, while salt and pepper noise appears as white and black noisy pixels. It forms due to sharp and sudden rays on the surface. Speckle noise is a result of some external fluctuations in X-rays when it returns from an object or human body. It appears as more intense gray pixels on an image [88]. Dealing with such noisy data is very important, but none of the studies reviewed in this survey tackle this problem. All the used datasets were clear versions of X-rays. Therefore, this can be a very useful contribution to remove such noise by passing data through suitable filters. For more details, you can explore the different types of filters from image processing, such as Gaussian filters or blurry filters. By improving accuracy over noisy datasets, we can create a model applicable to the real-time system.

9.5. Computational cost reduction

According to our literature review, almost 70% of papers used nature-inspired algorithms as feature selectors. Feature selection is very important and has a huge impact on the performance of the model. Image datasets have thousands of features that take a large amount of processing time. Despite producing highly accurate results using metaheuristic algorithms, the research on reducing the computational cost will remain an open problem, and thus future research can focus on addressing this issue.

9.6. Metaheuristic approach for other medical images

In this survey, we have reviewed the work done for COVID-19 chest images, but these findings can be useful for any other kind of medical image. Therefore, the researchers can use these methodologies and research to diagnose different disease images, such as cancer.

9.7. Other metaheuristic approaches

There are hundreds of nature-inspired algorithms that have been proposed in the last decade. This survey shows that only a few of them have been tested for feature selection in COVID-19 images. Many state-of-the-art optimization algorithms such as political optimizer [89] and heap base optimizer [90] can also be tested for feature selection in COVID-19 images.

9.8. Parameter optimization

A few studies have used metaheuristics for parameter optimization and architectural optimization of deep learning models. There are many recent studies that are developing the deep learning models [91] in order to learn better features and reduce the number of network parameters. These can also be tested for COVID-19 image classification.

9.9. Scalability

Machine learning and deep learning approaches learn patterns from the given dataset. A large number of training data guarantees a more realistic and informative model. However, this advantage comes with the drawback of more processing and more memory consumption. This is the reason why there are no research studies that used a big size of data in this survey. Therefore, this research gap can be covered by processing big COVID-19 datasets using distributed systems, such as Hadoop and Spark.

9.10. Detect new variant of COVID-19 (delta and omicron)

Current trends of COVID-19 spread show that it not only spreads very fast but also changes its shape, and the variants can be more fatal. The most recent variant is Omicron, which was discovered for the first time in November 2021 in South Africa. Omicron has been spreading at a very fast rate (double as compared to COVID-19). This survey provides a base study for the research community that wants to work on the detection of COVID variants using meta-heuristics and deep learning.

10. Limitations

Although many studies have been published on COVID-19 detection using meta-heuristics and deep learning, the research in this domain is still limited. Many limitations need to be addressed for more fruitful research in COVID-19 detection. The following are the major limitations.

10.1. Size of dataset

As observed in the discussion section 8.1, promising accuracies are reported on several datasets. However, on the other hand, the number of images in these datasets is very low. Dataset 2 is the most widely used dataset among all datasets reported in this survey, as shown in Table 1. Albadr et al. [20] reported a 100% accuracy for this dataset, but the experiments were conducted only on 200 images. Similarly, Dataset 5 has only 219 images [36]. The experiments performed on these datasets used deep learning models, which can produce biased results if only given limited data. These deep learning models might fail on new kinds of examples since they are trained on a limited number of images. To date, there is no large size annotated image dataset available for COVID-19 detection. A major challenge in creating a large dataset is the manual annotation of the images, which is time-consuming and involves human effort.

10.2. Class imbalance

Most of the datasets used for COVID-19 detection have a serious problem with class imbalance, as shown in Table 2. It is relatively easy to obtain positive examples for another lung disease since the disease is not new and there is plenty of published work available for those diseases. However, COVID-19 is a very recent pandemic that emerged in late 2019. The collection of positive examples for COVID-19 is still an ongoing task and more data is needed for deep learning models to function effectively. Due to the lack of positive examples, the deep learning models will not be able to learn the best features for the accurate classification of new images.

10.3. Noisy data

Some of the X-ray datasets used in COVID-19 studies are very noisy and it is difficult to detect any problem in those images, even with the human eye. There is a need for clear images of the lungs of COVID-19 patients as well as patients suffering from other lung diseases such as pneumonia. While it is relatively easy to detect COVID-19 affected lungs from healthy lungs, it is more difficult to differentiate between two unhealthy lung images suffering from different diseases. The negative examples must include lung images of pneumonia patients. This is to ensure that the deep learning models can learn features specific to COVID-19 affected lungs.

10.4. Time complexity of meta-heuristics

Metaheuristic algorithms tried to find the global optimal solution by searching for a large space for solutions. In every iteration, the fitness of each solution is evaluated, which makes these algorithms very time-consuming. Using meta-heuristics on large-size datasets will be a challenging task.

10.5. Lack of interdisciplinary knowledge

COVID-19 detection using metaheuristics and deep learning requires knowledge from the field of computer science, biology, virology, and medical imaging. There is a lack of interaction between computer science researchers and biologists, virologists, and medical imaging personnel. AI experts need in-depth biological and medical imaging knowledge about COVID-19 in order to apply their tools to solve this problem. It is necessary to coordinate the efforts being made in all of these domains in order to make progress in COVID-19 detection.

11. Conclusion

COVID-19 is an exponentially spreading disease that is now declared a pandemic. Since there is no permanent cure for this disease yet, the only option is to prevent it by keeping the affected person in isolation. It is very important to detect the patient at the early stages. This can be done by conducting medical tests that are costly, time-consuming, and not possible to perform at a large level. Automated classification of chest images can help in the early detection of COVID-19. Although deep learning algorithms are a popular choice for image classification, they have a large number of parameters to train, and this increases the computational cost. The computational cost can be reduced by optimizing these parameters using nature-inspired algorithms. In this study, we have conducted a detailed review of metaheuristic-based methods and their use cases in COVID-19 image classification. This is the first study on use case of metaheuristics for COVID-19 detection. Metaheuristics have been mostly used in feature selection and hyper-parameter optimization of deep learning models. This study presents the statistics of different types of datasets used in COVID-19 image classification. Details of different nature-inspired algorithms and their use cases are also reported in this study. It can be concluded from the survey that most researchers focused on the binary classification of X-ray datasets. CNN is the most common choice for feature extraction, while nature-inspired algorithms are the most popular for feature selection. For the task of classification, machine learning algorithms are a popular choice. Finally, we present several future research directions in the field of image classification for medical diseases. Some of the future research directions include increasing the size of the COVID-19 datasets, reducing the computational cost of deep learning methods, improving parameter optimization using new metaheuristics, and detection of new COVID-19 variants, etc. We firmly believe, this survey will inspire researchers to tackle the remaining challenges in COVID-19 image classification.

References

- [1] F. Wu, S. Zhao, B. Yu, Y.-M. Chen, W. Wang, Z.-G. Song, Y. Hu, Z.-W. Tao, J.-H. Tian, Y.-Y. Pei, et al., A new coronavirus associated with human respiratory disease in China, *Nature* 579 (7798) (2020) 265–269.
- [2] T. Singhal, A review of coronavirus disease-2019 (covid-19), *Indian J. Pediatr.* 87 (4) (2020) 281–286.
- [3] WHO Coronavirus (COVID-19) Dashboard, Retrieved 2021-12-11, <https://www.covid19.who.int/>.
- [4] T.B. Chandra, K. Verma, B.K. Singh, D. Jain, S.S. Netam, Coronavirus disease (covid-19) detection in chest x-ray images using majority voting based classifier ensemble, *Expert Syst. Appl.* 165 (2021) 113909.
- [5] D. Singh, V. Kumar, V. Yadav, M. Kaur, Deep neural network-based screening model for covid-19-infected patients using chest x-ray images, *Int. J. Pattern Recogn. Artif. Intell.* 35 (2021) 2151004, 03.
- [6] M. Canayaz, Mh-covidnet: diagnosis of covid-19 using deep neural networks and meta-heuristic-based feature selection on x-ray images, *Biomed. Signal Process Control* 64 (2021) 102257.
- [7] M. S. Iraj, M.-R. Feizi-Derakhshi, J. Tanha, Deep Learning for Covid-19 Diagnosis Based Feature Selection Using Binary Differential Evolution Algorithm, *arXiv preprint arXiv:2104.07279*.
- [8] H. Mittal, A.C. Pandey, R. Pal, A. Tripathi, A new clustering method for the diagnosis of covid19 using medical images, *Appl. Intell.* 51 (5) (2021) 2988–3011.
- [9] A. Altan, S. Karasu, Recognition of covid-19 disease from x-ray images by hybrid model consisting of 2d curvet transform, chaotic salp swarm algorithm and deep learning technique, *Chaos, Solitons & Fractals* 140 (2020) 110071.
- [10] A.K. Singh, A. Kumar, M. Mahmud, M.S. Kaiser, A. Kishore, Covid-19 infection detection from chest x-ray images using hybrid social group optimization and support vector classifier, *Cognitive Computation* (2021) 1–13.
- [11] S. Chattopadhyay, A. Dey, P.K. Singh, Z.W. Geem, R. Sarkar, Covid-19 detection by optimizing deep residual features with improved clustering-based golden ratio optimizer, *Diagnostics* 11 (2) (2021) 315.
- [12] E. Irmak, Implementation of convolutional neural network approach for covid-19 disease detection, *Physiol. Genom.* 52 (12) (2020) 590–601.
- [13] E.-S.M. El-Kenawy, S. Mirjalili, A. Ibrahim, M. Alrahmawy, M. El-Said, R.M. Zaki, M.M. Eid, Advanced meta-heuristics, convolutional neural networks, and feature selectors for efficient covid-19 x-ray chest image classification, *IEEE Access* 9 (2021) 36019–36037.
- [14] P.K. Shukla, J.K. Sandhu, A. Ahirwar, D. Ghai, P. Maheshwary, P.K. Shukla, Multiobjective genetic algorithm and convolutional neural network based covid-19 identification in chest x-ray images, *Math. Probl Eng.* 2021 (Meta-Heuristic Techniques for Solving Computational Engineering Problems) (2021), 7804540.
- [15] D. Younsri, M. Abd Elaziz, L. Abualigah, D. Oliva, M.A. Al-Qaness, A.A. Ewees, Covid-19 x-ray images classification based on enhanced fractional-order cuckoo search optimizer using heavy-tailed distributions, *Appl. Soft Comput.* 101 (2021) 107052.
- [16] M. Ben Jabra, A. Koubaa, B. Benjdira, A. Ammar, H. Hamam, Covid-19 diagnosis in chest x-rays using deep learning and majority voting, *Appl. Sci.* 11 (6) (2021) 2884.
- [17] G. Dhiman, V.V. Kumar, A. Kaur, A. Sharma, Don: deep learning and optimization-based framework for detection of novel coronavirus disease using x-ray images, *Interdiscipl. Sci. Comput. Life Sci.* (2021) 1–13.
- [18] M.E. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M.A. Kadir, Z.B. Mahbub, K.R. Islam, M.S. Khan, A. Iqbal, N. Al Emadi, et al., Can ai help in screening viral and covid-19 pneumonia? *IEEE Access* 8 (2020) 132665–132676.
- [19] E. H. Houssein, Z. Abohashima, M. Elhoseny, W. M. Mohamed, Hybrid Quantum Convolutional Neural Networks Model for Covid-19 Prediction Using Chest X-Ray Images, *arXiv preprint arXiv:2102.06535*.
- [20] M.A.A. Albadr, S. Tiun, M. Ayob, F.T. Al-Dhief, K. Omar, F.A. Hamzah, Optimised genetic algorithm-extreme learning machine approach for automatic covid-19 detection, *PLoS One* 15 (12) (2020), e0242899.
- [21] G.I. Sayed, G. Khoriba, M.H. Haggag, A novel chaotic salp swarm algorithm for global optimization and feature selection, *Appl. Intell.* 48 (10) (2018) 3462–3481.
- [22] K. Shankar, E. Perumal, P. Tiwari, M. Shorfuazzaman, D. Gupta, Deep learning and evolutionary intelligence with fusion-based feature extraction for detection of covid-19 from chest x-ray images, *Multimed. Syst.* (2021) 1–13.
- [23] M.Y. Sikkandar, Deep learning model for image-based diagnosis of covid-19 classification using convolution neural network, *Annals of the Romanian Society for Cell Biology* (2021) 2160–2174.
- [24] S.H. Basha, A.M. Anter, A.E. Hassanien, A. Abdalla, Hybrid intelligent model for classifying chest x-ray images of covid-19 patients using genetic algorithm and neutrosophic logic, *Soft Comput.* (2021) 1–16.
- [25] E.H. Houssein, M.M. Emam, A.A. Ali, Improved manta ray foraging optimization for multi-level thresholding using covid-19 ct images, *Neural Comput. Appl.* (2021) 1–21.
- [26] M.S. Iraj, M.-R. Feizi-Derakhshi, J. Tanha, Covid-19 detection using deep convolutional neural networks and binary differential algorithm-based feature selection from x-ray images, *Complexity* 2021 (Complexity and Robustness Trade-Off for Traditional and Deep Models) (2021), 9973277.
- [27] P. Bhowal, S. Sen, R. Sarkar, A two-tier feature selection method using coalition game and nystrom sampling for screening covid-19 from chest x-ray images, *J. Ambient Intell. Hum. Comput.* (2021) 1–16.
- [28] S. Sen, S. Saha, S. Chatterjee, S. Mirjalili, R. Sarkar, A bi-stage feature selection approach for covid-19 prediction using chest ct images, *Appl. Intell.* (2021) 1–16.
- [29] S. Mohammed, F. Alkinani, Y. Hassan, Automatic computer aided diagnostic for covid-19 based on chest x-ray image and particle swarm intelligence, *Int. J. Intell. Eng. Syst.* 13 (5) (2020) 63–73.
- [30] W. Hamdy, I. Elansary, A. Darwish, A.E. Hassanien, An Optimized Classification Model for Covid-19 Pandemic Based on Convolutional Neural Networks and Particle Swarm Optimization Algorithm, 2021.
- [31] M.A. Asghar, S. Razaq, S. Rasheed, et al., A robust technique for detecting sars-cov-2 from x-ray image using 2d convolutional neural network and particle swarm optimization, in: 2020 14th International Conference on Open Source Systems and Technologies (ICOSST), IEEE, 2020, pp. 1–6.
- [32] D.A.D. Júnior, L.B. da Cruz, J.O.B. Diniz, G.L.F. da Silva, G.B. Junior, A.C. Silva, A. C. de Paiva, R.A. Nunes, M. Gattass, Automatic method for classifying covid-19 patients based on chest x-ray images, using deep features and pso-optimized xgboost, *Expert Syst. Appl.* 183 (2021) 115452.
- [33] A. Dixit, A. Mani, R. Bansal, CoV2-Detect-Net: Design of COVID-19 prediction model based on hybrid DE-PSO with SVM using chest X-ray images, *Info. Sci.* 571 (2021) 676–692.
- [34] W.A. Hanon, et al., Svm-knn-ipo ensemble method for diagnosis of novel coronavirus (covid-19) with ct images, *Turkish J. Comput. Math. Educ. (TURCOMAT)* 12 (8) (2021) 51–59.
- [35] T. Hu, M. Khishe, M. Mohammadi, G.-R. Parvizi, S.H.T. Karim, T.A. Rashid, Real-time covid-19 diagnosis from x-ray images using deep cnn and extreme learning machines stabilized by chimp optimization algorithm, *Biomed. Signal Process Control* 68 (2021) 102764.
- [36] A. Narin, Accurate detection of covid-19 using deep features based on x-ray images and feature selection methods, *Comput. Biol. Med.* 137 (2021) 104771.
- [37] A. Castiglione, P. Vijayakumar, M. Nappi, S. Sadiq, M. Umer, Covid-19: Automatic detection of the novel coronavirus disease from ct images using an optimized convolutional neural network, *IEEE Transactions on Industrial Informatics* 17 (9) (2021) 6480–6488.
- [38] T. Goel, R. Murugan, S. Mirjalili, D.K. Chakraborty, Automatic screening of covid-19 using an optimized generative adversarial network, *Cogn. Comput.* (2021) 1–16.
- [39] S. C. Satapathy, D. J. Hemanth, S. Kadry, G. Manogaran, N. M. Hannon, V. Rajinikanth, Segmentation and Evaluation of Covid-19 Lesion from Ct Scan Slices-A Study with Kapur/otsu Function and Cuckoo Search Algorithm.
- [40] H.M. Afify, A. Darwish, K.K. Mohammed, A.E. Hassanien, An automated cad system of ct chest images for covid-19 based on genetic algorithm and k-nearest neighbor classifier, *Ingénierie des Systèmes d'Inf.* 25 (5) (2020) 589–594.
- [41] L. Chen, T. Rezaei, A new optimal diagnosis system for coronavirus (covid-19) diagnosis based on archimedes optimization algorithm on chest x-ray images, *Comput. Intell. Neurosci.* 2021 (Swarm Intelligence and Neural Network Schemes for Biomedical Data Evaluation) (2021), 7788491.
- [42] M. Kaur, V. Kumar, V. Yadav, D. Singh, N. Kumar, N.N. Das, Metaheuristic-based deep covid-19 screening model from chest x-ray images, *J. Healthcare Eng.* 2021 (2021), 8829829, <https://doi.org/10.1155/2021/8829829>.
- [43] E.-S.M. El-Kenawy, A. Ibrahim, S. Mirjalili, M.M. Eid, S.E. Hussein, Novel feature selection and voting classifier algorithms for covid-19 classification in ct images, *IEEE Access* 8 (2020) 179317–179335.
- [44] M. Kociotek, M. Strzelecki, R. Obuchowicz, Does image normalization and intensity resolution impact texture classification? *Comput. Med. Imag. Graph.* 81 (2020) 101716.
- [45] MinMax normalization, Retrieved 2021-09-11. URL <https://www.codecademy.com/>.
- [46] Resizing image, from digital image processing, Retrieved 2021-09-11. URL <http://www.sisu.ut.ee/>.
- [47] Grayscale Images, Retrieved 2021-09-11. URL <https://www.inf.ed.ac.uk/>.
- [48] SMOTE augmentation from Diving Deep with Imbalanced Data, Retrieved 2021-09-11. URL <https://www.datacamp.com/>.
- [49] Locality Sensitive Hashing for Similar Item Search, Retrieved 2021-09-11. URL <https://www.towardsdatascience.com/>.
- [50] Equations Normalization, Retrieved 2021-09-11. URL <https://www.developers.google.com/>.
- [51] Data Augmentation, Retrieved 2021-09-11. URL <https://www.nanonets.com/>.
- [52] E. Hassib, A. Eldesokey, L. Labib, S. Elghamrawy, Lsh-smote: a modified smote algorithm for imbalanced data-sets, *Ciência Técnica Vitivinícola* 33 (4) (2018) 50–65.
- [53] A Comprehensive Guide to Convolutional Neural Networks, Retrieved 2021-09-11. URL <http://www.towardsdatascience.com/>.
- [54] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, *arXiv preprint arXiv:1409.1556*.
- [55] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.
- [56] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya, et al., Chexnet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning, *arXiv preprint arXiv:1711.05225*.
- [57] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4700–4708.
- [58] C. Szegedy, S. Ioffe, V. Vanhoucke, A.A. Alemi, Inception-v4, inception-resnet and the impact of residual connections on learning, in: *Thirty-first AAAI Conference on Artificial Intelligence*, 2017.
- [59] M. Tan, Q. Le, Efficientnet, Rethinking model scaling for convolutional neural networks, in: *International Conference on Machine Learning*, PMLR, 2019, pp. 6105–6114.

- [60] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, *Adv. Neural Inf. Process. Syst.* 25.
- [61] Y. LeCun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L. D. Jackel, Backpropagation applied to handwritten zip code recognition, *Neural Comput.* 1 (4) (1989) 541–551.
- [62] I. MODULE, Googlenet: Going Deeper with Convolutions.
- [63] A.T. Sahlol, D. Yousri, A.A. Ewees, M.A. Al-Qaness, R. Damasevicius, M. Abd Elaziz, Covid-19 image classification using deep features and fractional-order marine predators algorithm, *Sci. Rep.* 10 (1) (2020) 1–15.
- [64] T. Goel, R. Murugan, S. Mirjalili, D.K. Chakrabarty, Optconet: an optimized convolutional neural network for an automatic diagnosis of covid-19, *Appl. Intell.* 51 (3) (2021) 1351–1366.
- [65] F. Chollet, Xception: deep learning with depthwise separable convolutions, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017*, pp. 1251–1258.
- [66] S.S.M. Ali, A.H. Alsaeedi, D. Al-Shammary, H.H. Alsaeedi, H.W. Abid, Efficient intelligent system for diagnosis pneumonia (sars-covid19) in x-ray images empowered with initial clustering, *Indonesian J. Elect. Eng. Comput. Sci.* 22 (1) (2021) 241–251.
- [67] T. Akram, M. Attique, S. Gul, A. Shahzad, M. Altaf, S.S.R. Naqvi, R. Damaševičius, R. Maskeliūnas, A Novel Framework for Rapid Diagnosis of Covid-19 on Computed Tomography Scans, *Pattern Analysis and Applications*, 2021, pp. 1–14.
- [68] T. Akram, M. Attique, S. Gul, A. Shahzad, M. Altaf, S.S.R. Naqvi, R. Damaševičius, R. Maskeliūnas, A Novel Framework for Rapid Diagnosis of Covid-19 on Computed Tomography Scans, *Pattern Analysis and Applications*, 2021, pp. 1–14.
- [69] W.M. Shaban, A.H. Rabie, A.I. Saleh, M. Abo-Elsoud, A new covid-19 patients detection strategy (cpds) based on hybrid feature selection and enhanced knn classifier, *Knowl. Base Syst.* 205 (2020) 106270.
- [70] M.A. Albadr, S. Tiun, M. Ayob, F. AL-Dhief, Genetic algorithm based on natural selection theory for optimization problems, *Symmetry* 12 (11) (2020) 1758.
- [71] M. Abdel-Basset, L. Abdle-Fatah, A.K. Sangaiyah, An improved lévy based whale optimization algorithm for bandwidth-efficient virtual machine placement in cloud computing environment, *Cluster Comput.* 22 (4) (2019) 8319–8334.
- [72] Z. Yan, S. Wang, B. Liu, X. Li, Application of Whale Optimization Algorithm in Optimal Allocation of Water Resources, vol. 53, 2018, 04019.
- [73] M. Abdel-Basset, V. Chang, R. Mohamed, Hsma woa: a hybrid novel slime mould algorithm with whale optimization algorithm for tackling the image segmentation problem of chest x-ray images, *Appl. Soft Comput.* 95 (2020) 106642.
- [74] X. Li, Z. Zhang, C. Huang, An epc forecasting method for stock index based on integrating empirical mode decomposition, svm and cuckoo search algorithm, *J. Syst. Sci. Inf.* 2 (6) (2014) 481–504.
- [75] X.-S. Yang, M. Karamanoglu, Nature-inspired computation and swarm intelligence: a state-of-the-art overview, *Nature-Inspired Comput. Swarm Intell.* (2020) 3–18.
- [76] A. Biju, T. Victoire, K. Mohanasundaram, An improved differential evolution solution for software project scheduling problem, *Sci. World J.* 2015 (2015), 232193, <https://doi.org/10.1155/2015/232193>.
- [77] V. Kachitvichyanukul, Comparison of three evolutionary algorithms: Ga, pso, and de, *Ind. Eng. Manag. Syst.* 11 (3) (2012) 215–223.
- [78] D. Singh, V. Kumar, M. Kaur, et al., Classification of covid-19 patients from chest ct images using multi-objective differential evolution-based convolutional neural networks, *Eur. J. Clin. Microbiol. Infect. Dis.* 39 (7) (2020) 1379–1389.
- [79] A. Faramarzi, M. Heidarinejad, S. Mirjalili, A.H. Gandomi, Marine predators algorithm: a nature-inspired metaheuristic, *Expert Syst. Appl.* 152 (2020) 113377.
- [80] D. Potnuru, K.A. Mary, C.S. Babu, Experimental implementation of flower pollination algorithm for speed controller of a bldc motor, *Ain Shams Eng. J.* 10 (2) (2019) 287–295.
- [81] S. Chakraborty, K. Mali, Sufmofpa: a superpixel and meta-heuristic based fuzzy image segmentation approach to explicate covid-19 radiological images, *Expert Syst. Appl.* 167 (2021) 114142.
- [82] Slime Mould Algorithm (SMA): A Method for Optimization, Retrieved 2021-09-11. URL <http://www.mathworks.com/>.
- [83] N. Dey, V. Rajinikanth, A.S. Ashour, J.M.R. Tavares, Social group optimization supported segmentation and evaluation of skin melanoma images, *Symmetry* 10 (2) (2018) 51.
- [84] G. Dhiman, V. Kumar, Emperor penguin optimizer: a bio-inspired algorithm for engineering problems, *Knowl. Base Syst.* 159 (2018) 20–50.
- [85] G. Zhao, H. Wang, D. Jia, Q. Wang, Feature selection of grey wolf optimizer based on quantum computing and uncertain symmetry rough set, *Symmetry* 11 (12) (2019) 1470.
- [86] M. Khishe, M.R. Mosavi, Chimp optimization algorithm, *Expert Syst. Appl.* 149 (2020) 113338.
- [87] P. s. i. r. t. d. l. mlfromscratchcom, *Neural Networks: Feedforward and Backpropagation Explained & Optimization*, 2021.
- [88] E. Manson, V. A. Ampoh, E. Fiagbedzi, J. Amuasi, J. J. Flether, C. Schandorf, *Image Noise in Radiography and Tomography: Causes, Effects and Reduction Techniques*.
- [89] Q. Askari, I. Younas, M. Saeed, Political optimizer: a novel socio-inspired meta-heuristic for global optimization, *Knowl. Base Syst.* 195 (2020) 105709.
- [90] Q. Askari, M. Saeed, I. Younas, Heap-based optimizer inspired by corporate rank hierarchy for global optimization, *Expert Syst. Appl.* 161 (2020) 113702.
- [91] S. Kanwal, I. Younas, M. Bashir, Evolving convolutional autoencoders using multi-objective particle swarm optimization, *Comput. Electr. Eng.* 91 (2021) 107108.