Detection of COVID-19 and its pulmonary stage using Bayesian hyperparameter optimization and deep feature selection methods

Nedim Muzoğu¹ | Ahmet Mesrur Hale£oğu² | Muhammed Onur Avci¹ | Melike Kaya Karaaslan³ | Bekir Siddik Binboğa Yarman⁴

¹Department of Biomedical Engineering, Faculty of Engineering, Istanbul University-Cerrahpasa, Istanbul, Turkey
²Department of Radiology, Sisli Hamidiye Etfal Training and Research Hospital, Health Sciences University, Istanbul, Turkey
³Department of Biomedical Sciences, Faculty of Engineering, Kocaeli University, Kocaeli, Turkey
⁴Department of Electrical-Electronics Engineering, Faculty of Engineering, Istanbul University-Cerrahpasa, Istanbul, Turkey

Correspondence
Nedim Muzoğu, Department of Biomedical Engineering, Faculty of Engineering, Istanbul University-Cerrahpasa, Istanbul, Turkey. Email: nmuzoglu@gmail.com

Abstract
Since the first case of COVID-19 was reported in December 2019, many studies have been carried out on artificial intelligence for the rapid diagnosis of the disease to support health services. Therefore, in this study, we present a powerful approach to detect COVID-19 and COVID-19 findings from computed tomography images using pre-trained models using two different datasets. COVID-19, influenza A (H1N1) pneumonia, bacterial pneumonia and healthy lung image classes were used in the first dataset. Consolidation, crazy-paving pattern, ground-glass opacity, ground-glass opacity and consolidation, ground-glass opacity and nodule classes were used in the second dataset. The study consists of four steps. In the first two steps, distinctive features were extracted from the final layers of the pre-trained ShuffleNet, GoogLeNet and MobileNetV2 models trained with the datasets. In the next steps, the most relevant features were selected from the models using the Sine-Cosine optimization algorithm. Then, the hyperparameters of the Support Vector Machines were optimized with the Bayesian optimization algorithm and used to reclassify the feature subset that achieved the highest accuracy in the third step. The overall accuracy obtained for the first and second datasets is 99.46% and 99.82%, respectively. Finally, the performance of the results visualized with Occlusion Sensitivity Maps was compared with Gradient-weighted class activation mapping. The approach proposed in this paper outperformed other methods in detecting COVID-19 from multi-class viral pneumonia. Moreover, detecting the stages of COVID-19 in the lungs was an innovative and successful approach.

KEYWORDS
bacterial pneumonia, Bayesian optimization, chest CT findings, feature selection, H1N1 viral pneumonia, occlusion sensitivity maps, sine-cosine optimization

1 INTRODUCTION

The new type of coronavirus (COVID-19) was detected in Wuhan, China’s Hubei province, as a “mysterious respiratory disease” and spread globally. The new coronavirus (COVID-19), which was declared a global pandemic by the World Health Organization (World Health Organization, 2020a) in early March 2020, has overwhelmed inadequate healthcare systems in many countries. The novel coronavirus SARS-
CoV-2 is an RNA virus, meaning that it is encoded in ribonucleic acid (RNA) unlike other viruses seen in humans and mammals. The virus generally transmits between people who are close to each other when aerosols or droplets containing the virus are inhaled or come directly into contact with the nose, mouth or eyes (Peng et al., 2020).

COVID-19 shows clinical symptoms similar to types of non-COVID-19 pneumonia, including fever, cough, chest pain, but requires different treatment and infection management. Most types of bacterial pneumonia can be treated with antibiotics, whereas antibiotics are not usually used for pneumonia caused by viruses (Ginsburg & Klugman, 2020). Although a definitive treatment for COVID-19 has not been approved yet, some antiviral therapies may be given received, depending on whether the person is hospitalized or not.

Although real-time polymerase chain reaction (RT-PCR) based on viral load is considered the gold standard for the diagnosis of COVID-19, the result is highly dependent on the site from which the sample was taken and the cellular material (Zhang et al., 2021). If the sample is virus-free or has a low viral load, false-negatives are common, and these patients cannot be diagnosed and isolated in a timely manner.

While case numbers are expected to decrease as vaccines continue to be administered, the number of weekly cases this year has increased to 6.813.948, compared to 3.587.213 at the same time last year. (World Health Organization, 2020b). This indicates that the pandemic is not slowing down, and that the rapid spread of the highly contagious COVID-19 continues to pose a serious threat to global health.

For this reason, rapid and accurate detection of the virus continues to play the most crucial role in reducing the speed of the pandemic by isolating the patients and treating them appropriately. Due to the slow turnaround times of PCR tests and the low sensitivity value of around 71%, there is a need for fast and high sensitivity screening techniques in addition to these tests (Fang & Pang, 2020). Studies show that thoracic computed tomography (CT) scans have a sensitivity of 98% and are more sensitive than PCR (Ai et al., 2020). Therefore, CT remains the primary imaging modality and small ground-glass opacity also can be detected in CT findings even in the absence of disease symptoms at the time of first admission to the hospital (Wang et al., 2020).

CT findings in the lung defined by the Fleischer Society vary according to the stage of the disease (Hansell et al., 2008). The pulmonary stages of COVID-19 and observed CT findings are defined as:

1. Early stage: Normal chest CT findings and ground glass opacities (GGO) are observed, 0–5 days after symptoms.
2. Progressive stage: 5–8 days interval, in which GGO and crazy paving patterns (CPP) are observed.
3. Peak stage: progressive consolidation is observed, 9–13 days.
4. Late stage: Mixed consolidation and ground-glass opacity (GGOC) are observed, >14 days.

As there are significant differences between the COVID-19 CT findings of severe/critical patients and patients not severely affected, it is crucial to diagnose pulmonary involvement to effectively combat COVID-19 (Cau et al., 2021; Zhou et al., 2020). Identifying these differences can be used as a rapid triage method to predict the prognosis of COVID-19 patients and to decide whether patients should be admitted to the intensive care unit and/or mechanical ventilation (Khosravi et al., 2021; Leonardi et al., 2020). Therefore, it is necessary to examine the radiological findings to provide the important information that health professionals need to fight COVID-19 more effectively.

Structural similarities of pneumonia types and varying sensitivities of radiologists in classifying pneumonia types are obstacles to the effective use of computed tomography images in the pandemic. For example, in the study examining how well radiologists can distinguish COVID-19 from viral pneumonia, their sensitivities were found to range from 70% to 97%. As you can see, the classification success of radiologists seems to vary (Bai et al., 2020). Distinguishing COVID-19 from other types of pneumonia using computer-aided diagnosis systems (CAD) from patients' initial CT scans and classifying CT findings determines proper triage, hospitalization, and critical care needs, which will make it possible to develop appropriate strategies to combat the pandemic. Therefore, there is urgent need to integrate CAD systems based on CT images with the picture archiving and communication system (PACS) to help radiologists identify suspected cases of COVID-19 and distinguish COVID-19 from non-COVID-19 viral pneumonia.

Recently, artificial intelligence (AI) has been used in many applications such as image segmentation and classification (Kundu et al., 2022; Shah et al., 2021). Deep learning approach can reveal image features that are invisible to the human eye in the original image (Krizhevsky et al., 2017). Convolutional neural networks (CNN) emerged with developments in deep learning and computer vision and have proven to be very successful in feature extraction. They are widely used, and studies continue to perfect them (Canayaz, 2021b; Vural & Akbacak, 2020). Transfer learning techniques have greatly facilitated the deep learning process by allowing quick retrain of a very deep CNN network, even with a relatively small number of images, using pre-trained networks (Chahar & Roy, 2021; Demir et al., 2022).

In many articles, COVID-19 recognition has been performed with the transfer learning approach using X-ray and CT images. Due to the lack of sufficient publicly available COVID-19 and various viral pneumonia CT datasets, it appears that deep-learning classification studies to distinguish COVID-19 from other types of viral pneumonia are mainly performed with X-ray datasets. Few CT studies have been published in which COVID-19 has been classified together with at least two different types of pneumonia (Bacterial, non-COVID viral, COVID-19 viral). The classifications based on X-ray and CT related to COVID-19 detection in the studies are summarized as follows:

Ibrahim et al. (2021) used AlexNet on COVID-19, non-COVID-19 viral pneumonia, bacterial pneumonia and healthy images, but only achieved 89.18% sensitivity. Khan et al. (2020) proposed a fine-tuned network pre-trained with Xception to classify COVID-19, bacterial pneumonia, viral
pneumonia, and healthy X-ray images, called CoroNet. CoroNet achieved precision and recall rates for COVID-19 cases at 93% and 98.2%, respectively. Despite the successful results of deep-learning studies with radiographic imaging, the disadvantage of radiographic imaging is that early diagnosis (stage I) of COVID-19 findings is possible with CT but not with radiographic imaging. Jain et al. (2020) used ResNet-50 to classify X-ray images of bacterial pneumonia, viral pneumonia and healthy lungs. However, a second step with 5-fold cross-validation using ResNet-101 is required to distinguish COVID-19 from the classified viral pneumonia class. Ucar and Korkmaz (2020) proposed the pre-trained SequezeNET model, in which hyperparameter settings were adjusted with Bayesian Optimization Algorithm, for COVID-19 detection. In the study, COVID-19 was detected with 100% accuracy, radiographs of patients with pneumonia and normal patients were also used in the classification. Nour et al. (2020) performed hyperparameter tuning by applying Bayesian optimization with k-nearest neighbour, support vector machine, and decision tree machine learning classifiers on the features obtained from the five-layer convolutional network structure and achieved the best success of 98.7% with SVM classifiers. Krishnaswamy Rangarajan and Ramachandran (2022) fused SequezeNet and ShuffleNet to extract deep features from 17,367 images obtained by data augmentation from a total of 2480 COVID and non-COVID images. The proposed fused model achieved an overall accuracy of 97%. Zhou et al. (2021) proposed an ensemble learning method with majority voting using pre-trained AlexNet, GoogLeNet and ResNet models called EDL_COVID that classified COVID-19, healthy and tumour chest CT images with 99.05% accuracy. The studies are summarized as in the Table 1.

Metaheuristic algorithms which include machine-learning techniques to increase the efficiency of deep models are now widely used in many studies with transfer learning. Metaheuristic algorithms play an important role in the field of AI by reducing the data size using the feature selection methods (Canayaz, 2021b). Some recently proposed meta-heuristic algorithms for feature selection are as follows:

The Monarch Butterfly Optimization (MBO) algorithm is a metaheuristic swarm intelligence algorithm that simulates the monarch butterfly’s reverse migration behaviour, beginning in Canada and moving south to Mexico, then north. Alweshah et al. (2020) significantly reduced the selection size for 18 benchmark datasets with their proposed model by using the MBO algorithm for feature selection and achieved a high classification accuracy of 93% on average. The Slime Mould Algorithm (SMA) is a population-based metaheuristic algorithm that simulates the positive and negative feedback of the propagation wave of slime mould. Ewees et al. (2021) proposed a modified SMA to improve the exploration of SMA. They maximized classification accuracy by minimizing the number of features of 20 different medical datasets, and a remarkable success was observed in their results compared to existing metaheuristic algorithms. The Colony Predation Algorithm (CPA) represents the cooperative and foraging behaviour used by animal hunting groups, such as containment of prey, dispersal of prey, and support of the most likely successful predator. Shi et al. (2021) identified the optimal parameters of the extreme learning machine with CPA to discriminate and classify the severity of COVID-19. The Hunger Games Search (HGS) is a population-based metaheuristic algorithm that simulates the behaviour and actions of animals to ensure the animals’ chances of survival and food acquisition as the most important motivation for their behaviour and actions. Wang et al. (2021) detected COVID-19 positivity from the computed tomography images with the HGS, which they used to increase the stability of the proposed model. The Moth Search Algorithm (MSA) is a population-based metaheuristic algorithm that simulates the movement of moths towards or away from a light source and their Lévy flight tendency around their position. Elaziz et al. (2019) aimed to minimize the make span to schedule a set of tasks in virtual machines (VMs) by computing the fitness function for a solution, and obtaining the best solution set by using the Moth Search Algorithm (MSA) and Differential Evolution Algorithm (DEA) as a hybrid. MSDE has been shown to provide efficient scheduling for the VM while minimizing the make span. The Harris Hawks optimization (HHO) is a population-based swarm algorithm that simulates the feeding behaviour of hawks, which explore and surprise their prey, attack from different directions, and cooperate in hunting with different tracking methods. Balaha et al. (2020) detected COVID-19 from CT images with an accuracy of 93% to 99% by optimizing the parameters of nine different pre-trained networks with HHO in their proposed structure. SCA is a meta-heuristic algorithm that generates an initial population of agents and requires it to fluctuate to balance exploration and exploitation to find an optimal solution based on trigonometric sine and cosine functions. Sindhu et al. performed feature selection using SCA, tested 10 benchmark medical datasets with extreme learning machine (ELM), compared the results with well-known meta-heuristic algorithms such as Particle Swarm Algorithm (PSO) and Genetic Algorithm (GA), and achieved better classification performance (Abualigah & Diabat, 2021).

In this study, a new dataset was created to successfully differentiate the most common types of viral pneumonia (influenza A (H1N1) viral pneumonia, bacterial pneumonia) with clinical symptoms similar to COVID-19, both from each other and from healthy lung patients. Also, when the COVID-19 images in the dataset were analysed by radiologists, it was found to contain five different classes of chest CT findings according to their stage. Thus, the second data set, the COVID-19 CT findings data set, was created by classifying the COVID-19 images according to their stage. Therefore, this study not only distinguishes COVID-19 but also proposes a system to help intensivists assess the patient’s ventilatory needs by detecting the stages of COVID-19 CT findings. Since the approach is recommended as an auxiliary application for the radiologists in PACS systems, it was developed with the models that use fewer network parameters, considering the hardware requirements of these systems. Consequently, the success in classifying different types of pneumonia and COVID-19 CT findings with the proposed approach was examined. The following are the study’s main innovations and contributions:

1. The main idea is to detect COVID-19 and predict disease prognosis by classifying the most discriminating features with the best-optimized classifier.
**Table 1** Details of the previous studies involving deep learning techniques for COVID-19 diagnosis

| Work reference | Models / methods | Dataset | Train/validation/test ratio (%) | Modality | Year | Overall Acc (%) |
|----------------|------------------|---------|---------------------------------|----------|------|-----------------|
| Ibrahim et al. (2021) | Transfer learning, Data augmentation | 284 COVID-19, 327 Viral, 330 Bacterial, 310 Normal | 70:30 | X-ray | 2020 | 93.42 |
| Khan et al. (2020) | CNNs, Residual structure | 371 COVID-19, 4237 Viral, 4078 Bacterial, 2882 Normal | 4-fold | X-ray | 2020 | 95 |
| Jain et al. (2020) | Transfer learning, Data augmentation | 250 COVID-19, 350 Viral, 300 Bacterial, 315 Normal | 80:20 | X-ray | 2020 | 98.2 |
| Uçar and Korkmaz (2020) | Transfer learning, Data augmentation, Bayesian optimization | 45 COVID-19, 1591 Pneumonia, 1203 Normal | 80:10:10 | X-ray | 2021 | 98.3 |
| Nour et al. (2020) | CNNs, Bayesian optimization | 219 COVID, 1345 Viral, 1341 Normal | 70:30 | X-ray | 2020 | 98.7 |
| Krishnaswamy Rangarajan and Ramachandran (2022) | Transfer learning, Data augmentation | 1252 COVID, 1230 non-COVID | 70:30 | CT | 2021 | 96.98 |
| Zhou et al. (2020) | Transfer learning, Ensemble learning | 2500 COVID-19, 2500 Normal, 2500 Lung tumour | 5-fold | CT | 2020 | 99.05 |

Abbreviation: CNNs, convolutional neural networks.
2. Successful hybrid use of the Sine Cosine Optimization Algorithm for feature selection and Support Vector Machines for classification, in detecting COVID-19 and predicting the prognosis of COVID-19 disease with AI.

3. It has the highest classification accuracy in detecting COVID-19 and predicting the prognosis of COVID-19 disease by determining the best point hyperparameter of the classifier of the hybrid model with the Bayesian Optimization Algorithm.

The organization of the proposed paper is as follows; the datasets, the CNN models and the methods are explained in Section 2. The experimental results are reported in Section 3. There is a discussion and conclusion at the end of the study.

2 | DATASET, MODELS, AND METHODS

2.1 | Datasets

The CT images used in this study belong to four different classes: COVID-19 pneumonia, non-COVID-19 pneumonia (influenza A H1N1), bacterial pneumonia and healthy lung. Pneumonia datasets were obtained from Istanbul Hamidiye Etfal Training and Research Hospital PACS systems with the approval of the Ethics Committee and reviewed by radiologists from the Radiology Department. Of the healthy lung CT images, 236 were obtained from cross-sectional images at https://radiopaedia.org, and 72 from https://github.com/deepimages2021. These healthy lung CT images are published at https://github.com/nilimuzoglu/healthy-lung-computed-tomography-308-images for similar studies. The introduced dataset consists of 614 images from COVID-19 CT (taken between 3/1/2020 and 31/12/2020), 415 images from viral H1N1 pneumonia CT (taken between 1/1/2019 and 10/30/2019), 518 images from bacterial pneumonia CT (taken between 4/1/2019 and 11/30/2020), 236 and 72 images of the healthy CT image dataset (09/12/2021 and 27/11/2021). This dataset is an updated version of the CPB dataset used in the study of Toğaçar et al., (2022). The updates include the addition of healthy lung images and the exclusion of unclassified COVID-19 images in the five CT finding classes from the CPB dataset. For this reason, we will call this dataset CPBH. The images resolutions in the datasets vary from 512 to 1096 pixels and are in .jpg and .jpeg formats. However, since the accepted dimensions for input images of the deep-learning models are 224 x 224, all images have been resized accordingly. The demographic characteristics of the 423 patients in the CPBH dataset are summarized in Table 2, where n represents the number of patients.

To classify the pulmonary stages of the disease in the COVID-19 images of the CPBH dataset, the images were analysed by the radiology department and found to contain five different classes of chest CT findings. The five different classes of chest CT findings included in the COVID-19 dataset are listed below.

1. consolidation,
2. crazy-paving-pattern (CPP)
3. ground-glass opacities (GGO)
4. mixed ground-glass opacities and consolidation (GGOC),
5. mixed ground-glass opacities and nodule (GGON).

Moreover, the COVID-CT dataset published by Yang et al. (2020) has also been classified by the radiologists into CT finding groups to compensate for the unbalanced number of images in the five COVID-19 CT findings classes. Since the number of GGO CT findings from CPBH was sufficient, the findings from COVID-CT were added to the dataset, excluding the GGO class, and the updated version of the CovCT findings dataset used in the publication of Toğaçar et al. (2022) was created. The CT findings dataset created for the first time for the deep learning approach will be referred to as CovCT-Findings, as in the study. The percentage distributions of the data sets by classes and the number of images are shown in Figure 1 and Table 3, respectively.

| Characteristic | COVID-19 (n = 196) | Pneumonia (n = 211) | Bacterial (n = 10) | Healthy (n = 6) |
|----------------|-------------------|--------------------|-------------------|-----------------|
| Age (y) mean   | 53.17 ± 17.58     | 58 ± 11            | 58 ± 11           | 36.84 ± 8.56    |
| <60            | 130 (67%)         | 98 (46%)           | 4 (40%)           | 4 (66%)         |
| ≥60            | 66 (33%)          | 113 (54%)          | 6 (60%)           | 2 (33%)         |
| Male           | 107 (55%)         | 130 (61%)          | 5 (50%)           | 3 (50%)         |
| Female         | 89 (45%)          | 81 (39%)           | 5 (50%)           | 3 (50%)         |

Note: Percentages in parentheses are patient data. Age presentations are mean ± SD.
Figure 2 shows examples of axial lung window CT images from the CPBH dataset, and Figure 3 shows examples of axial lung window CT images from the CovCT-findings dataset.

All pneumonia CT scans obtained from the PACS system were performed with a 128-detector array (Somatom Sensation, Siemens, Germany) helical scanner. Tomography scans were performed with the patient's breath held, and the CT parameters were as follows: minimum and maximum...
maximum tube current (150–250 mAs), 2-mm reconstruction interval, and 5-mm slice thickness. Two radiologists (with 15 and 25 years of experience) interpreted Chest CT scans, blindly and independently. This study was carried out using the Windows 10 64-bit operating system with an AMD Ryzen7@3.20 GHz processor, 32GB of RAM, and an 8GB NVIDIA GeForce RTX 3070 graphics card (GPU). MATLAB 2021a was used for classification.

2.2 Deep convolutional neural networks

The deep CNNs used in this paper were trained with ImageNet dataset (Russakovsky et al., 2015), which consists of more than 1 million natural images and 1000 different classes. While there are differences between deep CNNs, typical CNN architectures consist of layers such as convolutional layers, pooling layers, and fully connected layers. Basically, the convolution layers aim to generate different feature maps of the given image using multiple kernels. The pooling layer reduces the dimensionality of the feature maps, retaining only the important features and reducing spatial invariance. The output of the last pooling layer is flattened before being passed to one or more fully connected layers, and each point is transmitted to the last layer via variable weights that have as many output nodes as the number of classes. The SoftMax function normalizes its inputs to target class probabilities for classification tasks. Then, these output values are used as input to the cross-entropy loss function, which measures their distance from the truth-values. Minimizing the difference between outputs and ground truth labels by optimizing the parameters using an optimization algorithm, the models can make better predictions (Gu et al., 2019; Ajit et al., 2020).

In this study, for the reasons previously described, we focused on the deep CNN architectures with fewer parameters, but almost equivalent accuracy compared to well-known models (Vgg16, Vgg19 and ResNets) that have achieved successful results in many studies (Chowdhury et al., 2020; Teodoro et al., 2021). There are significant differences in features, data size and number of classes between medical image datasets that describe pathologies in local tissues and natural image datasets such as ImageNet, where the image has a clear subject (Alzubaidi et al., 2021). Since the characteristics (size, features, etc.) of the datasets are different from the ImageNet dataset, it was determined after some experiments that it is better to train the model from scratch to achieve more successful results, even if it requires high computation. The ImageNet structures trained with the datasets is used for feature extraction and described as follows.

2.2.1 GoogLeNet

GoogLeNet is a deep convolutional neural network developed by researchers at Google (Balagourouchetty et al., 2020). The GoogLeNet architecture consists of nine inception modules. The outputs of this process, which is performed in parallel at the input, are stacked at the output in a structure designed to better handle objects at different scales by using 1 × 1, 3 × 3, 5 × 5 and 3 × 3 convolutional filters of different sizes in the inception module. The architecture, designed based on computational efficiency, is built to be used in everyday devices. The Rectified Linear Units (ReLU) activation function is used to introduce non-linearity to the model, and a 40% dropout layer is used to optimize the weights during training. The addition of two auxiliary classifiers in the middle layers to prevent the network from stopping learning by neglecting small gradient values in

FIGURE 3 Computed tomography (CT) images of COVID-19 pulmonary stages from the CovCT-findings dataset used in the experimental analysis of this study. (a) Crazy-paving-pattern (b) consolidation, (c) ground-glass opacities, (d) ground-glass opacities and nodule, (e) ground-glass opacities and consolidation
the lower layers is another prominent feature of GoogLeNet, called vanishing gradient descent. The auxiliary classifiers calculate the loss during training and add back the total loss of the network, which contributes to the performance of the model. In this study, 1000 significant features were obtained from the fully connected layer named loss3.

2.2.2 | ShuffleNet

ShuffleNet is a convolutional neural network model designed for Deep Learning use in everyday devices (drones, robots, applications on phones) by maintaining high accuracy while reducing computational parameters (Gomes et al., 2021). Group convolution with channel shuffle is the second most important feature of this design. ShuffleNet uses the group convolution feature that allows parallelization of data to create the second most important feature. The Channel Shuffle operation, on the other hand, aims to better organize the feature maps by splitting them into multiple subgroups and feeding them with different subgroups in the next layer. In this study, 1000 significant features were obtained from the fully connected layer named node-202.

2.2.3 | MobileNetV2

MobileNetV2 introduces the reverse residual layer, which skips connections between narrow parts of the network, which is the opposite of an original residual connection (Gang et al., 2021). It also uses the linear bottleneck layer, where the last convolution of the block has a linear output before being added to the first activations to compensate for the loss of information. The new layer is based on a type of factorized convolution for mobile applications, which reduces the computational cost compared to standard convolutions called depth-wise separable convolutions introduced in MobileNetV1. In this study, 1000 significant features were obtained from the fully connected layer called Logits.

2.2.4 | Parameters of the deep convolutional neural networks

Optimization algorithms modify the weights and learning rate at each epoch to minimize the loss function while training deep learning models. The ADAM method (Kingma & Ba, 2015) combines the heuristics of both momentum and RMSProp (Kwon et al., 2021) to calculate the exponential average and squares of the gradient and adaptive learning rates for each parameter. The Adam optimization algorithm, which has led to successful results in many deep learning studies due to its structure based on both momentum and learning rates, was preferred in this study. The necessary parameters to determine the Adam update rule are shown in Equations (1)–(4). There, $m_t$ is the first-order momentum (mean) of the gradient and $v_t$ is the second-order momentum of the gradient, $\beta_1$ and $\beta_2$ are the hyper-parameters control the exponential decay rates of $m_t$ and $v_t$, $b_m$ and $b_v$ are the bias-corrected first and second moment respectively and $U_{t+1}$ is Adam update rule.

\[
m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{1}
\]

\[
v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \tag{2}
\]

\[
b_m = \frac{m_t}{1 - \beta_1^t} \tag{3}
\]

\[
b_v = \frac{v_t}{1 - \beta_2^t} \tag{4}
\]

\[
U_{t+1} = U_t - \frac{\eta}{\sqrt{b_v + \epsilon}} b_m \tag{5}
\]

| Table 4: Models and parameters |
|--------------------------------|
| CNN architecture | Image size | Depth | Optimization | Batch size | Gradient decay factor | Learning rate |
|-------------------|------------|-------|--------------|------------|-----------------------|---------------|
| GoogLeNet         | 224 × 224 | 22    | Adam         | 64         | 0.99                  | 1e-5          |
| MobileNetV2       | 224 × 224 | 53    | Adam         | 64         | 0.99                  | 1e-5          |
| ShuffleNet        | 224 × 224 | 50    | Adam         | 64         | 0.99                  | 1e-5          |

Abbreviation: CNN, convolutional neural network.
where $g_t$ is a gradient vector, $\eta$ is learning rate, and defaults to 0.9 for $\beta_1$ and 0.999 for $\beta_2$, and $10^{-8}$ for $\epsilon$.

The size of the mini batches was determined depending on the size of the dataset and the hardware performance used for training and was set to 64. Since two different datasets are used, the training iterations may differ. Therefore, the epoch value of 50 for CPBH was adjusted to 100 for the CovCT Findings dataset, taking into account experience in the study. The parameters used in this study for the pre-trained models are shown in Table 4.

### 2.3 | Method

#### 2.3.1 | Feature selection method

It is worth noting that the datasets used for artificial intelligence are a group of features (Sayed et al., 2019; Shen & Zhang, 2021). Reducing the dimensionality of the data by removing redundant and irrelevant features from the search space increases the prediction and classification success (Gu, Cheng, & Jin, 2018). Since the main goal of feature selection is to eliminate these features and select the most relevant ones, it is widely used in fields such as artificial intelligence and statistics (Gheyas & Smith, 2010). Therefore, in this study, the most relevant features are selected using the feature selection algorithm to achieve better classification performance.

The sine-cosine algorithm (SCA) used in this study is a population-based metaheuristic feature selection algorithm introduced by Mirjalili in 2016 that uses sine and cosine functions (Mirjalili, 2016). The SCA starts with a set of multiple solutions in each search space and fluctuates them outward or towards the best solution. These solutions are evaluated using the fitness function to obtain the target solution. In a search algorithm, finding completely new fields of the search space is called exploration, and finding these fields of the search space in the neighborhood of previously visited points is defined as exploitation. SCA obtains the destination solution depending on the parameters $r_1$, $r_2$, $r_3$, and $r_4$. The parameter $r_1$ determines the next direction of motion, which may be inside or outside the gap between the solution and the target, as shown in the Figure 4. The search algorithm changes the range of sine and cosine corresponding to $r_1$ specified by the following Equation (6) to balance exploration and exploitation,

$$r_1 = a - t \frac{Xa}{T}$$

(6)

where $T$ is the maximum number of iterations and $a$ is a constant when $t$ is the current iteration. If $r_1 < 1$, the distance between the solution and the target gets smaller. If $r_1 > 1$, distance gets bigger (Taghian & Nadimi-Shahraki, 2019). A higher value of the parameter $r_2$ leads to exploration and a lower value leads to exploitation so that it directs movement towards or away from the target. The parameter $r_3$ introduces a random weighting for the effect of the target in defining distance (Bhadoria et al., 2020). When the parameter $r_4$ is less than 0.5, the sine function and when it is greater than 0.5, the cosine function is effective, as given in Equations (7) and (8).

$$X^{t+1}_i = X^t_i + r_1 x \sin(r_2)x|X^t_i - X^t_{r_4}E^t_i|, r_4 < 0.5$$

(7)

If $r_1 < 1$, the next position for the X solution point is directed to the black-grid region.

If $r_1 > 1$, the next position for the X solution point is directed to the yellow-grid region.

**FIGURE 4** Effects of sine and cosine in Equations (7) and (8) on the next position
Where $X_{i}^{t+1}$ is the current solution value in the $i$-th dimension and $t$-th iteration, $P_i^t$ is the destination point position in $i$-th dimension and at iteration $t$, and $r_4$ is a random number between 0 and 1 (Abualigah & Diabat, 2021; Taghian & Nadimi-Shahraki, 2019) The Effect of sine and cosine functions in Equations (7) and (8) are shown in Figure 4.

As a result, a subset of features was created by selecting the most relevant among 1000 features extracted from the last layers of the models using the SCA feature selection algorithm. These subsets were then classified by the Support Vector Machine classifier.

2.3.2 | Support vector machine classifier

Support vector machine (SVM) is a classification method developed to find the optimal hyperplane with the largest margin between classes (Cortes et al., 1995). The points, closest to the hyperplane from the classes and support the hyperplane/decision boundary as a column, are called support vectors. Using the training data represented by $\{x_i, y_i\}, i = 1, 2, ..., r, y_i = \{1, -1\}$ in $n$-dimensional space, and the hyperplane equation defined by $w.x + b = 0$, where $x$ is a point extending on the hyperplane, $w$ is the coefficient vector of the hyperplane, and $b$ is the offset of the hyperplane from the origin, and a separating hyperplane can be defined as,

$$y_i(w.x_i + b) - 1 \geq 0$$  \hspace{1cm} (9)

The representation of SVM in two-dimensional space for linearly separable patterns is shown in Figure 5. The width of the maximum distance between the support vectors and the hyperplane is defined with $\frac{2}{\|w\|}$ as the margin of the classifier. Maximizing this margin leads to an optimization solution in which the data points are classified efficiently and with as few errors as possible. SVM is a method based on the structure of a classification function that can distinguish binary classes. There are two different approaches that aim to use the basic SVM by reducing the multi-class problem to a set of binary problems to perform multi-class classification examples (Galar et al., 2011).

1. One versus All: N- class patterns then N binary classifier structure.
2. One versus One: N- class patterns then $\frac{N \times (N - 1)}{2}$ binary classifier structure.

In SVM, we usually need a kernel function to convert the training dataset from a nonlinear decision surface to a linear equation in multi-dimensional space. Therefore, in this study, we used the one-vs-one method with quadratic kernel as a multiclass strategy by evaluating the results of different kernel functions (cubic, linear, quadratic etc.) depending on their success before the hyperparameter optimization process.
2.3.3 | Hyperparameter optimization method

The hyperparameters (epoch number, learning rate, activation function, etc.) control learning process and are set by the analyst before training according to the requirements of the model. The optimization algorithm tries to obtain the most accurate values from the hyperparameter set within a certain range. Thus, the hyperparameter optimization can be expressed as:

$$\lambda^* = \arg\min_{\lambda \in A} f(x)_{X_{train}, X_{test}}$$

(10)

Here, $\lambda$ is a hyperparameter vector, denoted by $\lambda \in A$ for the hyperparameter space $A$, where $f$ is the objective function used with the $X_{dataset}$ whose loss value is to be measured. The target is to discover the optimal $\lambda$-parameter configuration for the lowest loss value we can obtain through a sequence of evaluations of the objective function $f(x)$.

The hyperparameter optimization algorithm can optimize and improve the performance of the deep learning methods (Hutter et al., 2019). To achieve high accuracy in deep learning studies, many parameters of machine learning algorithms are optimized to obtain the global minimum of the objective function considering the time consumption; it is difficult to manually apply the required hyperparameters one by one for the training process. Grid search, random search, and many swarm optimization algorithms (Bayesian, etc.) are employed to automatically optimize the hyper-parameters of machine learning. The grid search evaluates all combinations of the specified hyperparameters sequentially and comprehensively, and is therefore computationally expensive (Wu et al., 2019). In contrast, random search only tries samples from a given list or statistical distribution of the parameter. These methods explore the hyperparameter space blindly and do not learn from previous iterations. On the other hand, Bayesian optimization is a powerful hyperparameter search method that incorporates prior data about hyperparameters for global optimization of expensive black-box functions. Namely, given a prior distribution over the objective function $p(f)$ and after taking into consideration the new information, Bayes’ theorem (Wu et al., 2019) can be used for decision-making in modelling the objective function for the posterior distribution of $f$ given $X_{dataset}$ (Malkomes & Garnett, 2018). Therefore, Bayesian optimization was used in this study to optimize the parameters of the SVM classifier and achieve better classification accuracy.

The Bayesian optimization algorithm uses the Gaussian process as a surrogate model to model the objective function $f(x)$, where the surrogate model fits the $f(x)$ function every new iteration. The second fundamental component of Bayesian optimization is the acquisition function, which calculates the next best point using the predictive distribution of the probabilistic model (Hutter et al., 2019). In minimizing the $f(x)$ function, the next position is obtained by maximizing the acquisition function. Although there are many types of acquisition functions, the preferred acquisition function in this study for configuration $\lambda$ is the Expected Improvement function calculated as follows:

$$[I(\lambda)] = (f_{\min} - \mu(\lambda))\Phi((f_{\min} - \mu(\lambda))/\sigma) + \sigma\phi((f_{\min} - \mu(\lambda))/\sigma)$$

(11)

where $\mu$, $\sigma$, and $f_{\min}$ are mean, variance, and best observed value, respectively. $\Phi$ and $\phi$ are the standard normal distribution and standard normal density function, respectively.

In this study, the hyperparameter search range of the Support Vector Machines classifier to be optimized is as follows:

1. Kernel functions: Cubic, Gaussian, Linear and Quadratic.
2. Box constraint level: [0.001, 1000].
3. Kernel scale: [0.001, 1000].
4. Multiclass method: One-versus-One and One-versus-All.
5. Standardize data: true and false.

2.3.4 | Deep learning visualization method

Visualization methods are used to establish more successful structures by visualizing misclassifications and which parts of the image the network makes a particular decision by looking at. Methods can be local to a particular input image or global to an entire data set, and the common distinction between the methods is whether they are based on gradients or perturbations (MathWorks, 2021). In this study, perturbation-based Occlusion Sensitivity Maps (OSM) were chosen as the deep learning visualization method to visualize the most important features of the image that CNNs uses to make classification decisions (Aminu et al., 2021). OSM is an image visualization method that measures the difference in classification score for the classes when patches of the input image are masked out, and then locates the disease regions in the image. For this, OSM calculates the change in probability score for a given class by moving a mask across the input image that occludes different parts of the input image. Based on this score the most important parts of the image for classification are highlighted.
We preferred OSM, which gives more sensitive results than Gradient-Weighted Class Activation Mapping (Grad-CAM) (Selvaraju et al., 2017) in the interpretation of multifocal lesions, is compared in Section 3.

2.3.5 | Proposed approach

This study was carried out with two different datasets (CPBH and CovCT-Findings). CPBH includes four classes (COVID-19, influenza A (H1N1) Viral Pneumonia, Bacterial Pneumonia, Healthy Lung). CovCT Findings, which are used to classify pulmonary stages of the disease in the lungs from the COVID-19 images in the CPPH and COVID-CT datasets, consists of five classes (GGO, CCP, GGOC, GGON, Consolidation). For easy reading, we thought it appropriate to name our hybrid deep learning approach SCA-BayeSVMNet, which includes the name of the classifier and the optimization algorithms used. SCA-BayeSVMNet includes 4 steps for both datasets the proposed study includes 4 steps for both datasets. In the first step of the approach, the deep learning models were trained with the datasets and classified with Softmax to be compared with the SVM classification results used in the general structure of this study. In the second step, 1000 features were extracted from the last layers ‘loss3’, ‘logits’ and ‘node_202’ of the GoogLeNet, MobileNetV2 and ShuffleNet models for each image and subjected to SVM. The SCO algorithm was used in the third step to obtain the discriminative features from the 1000 features extracted from each model. The selected features were first evaluated separately for each model, then the combined features were classified using SVM. Since both datasets contain unbalanced classes, the combined features are validated by 5-fold cross-validation to test stability and generalization even when estimating minority classes. In the fourth step, it is aimed to achieve better results with hyperparameter optimization by using the most successful features set in the previous step. Figure 6 shows an overall block diagram of the SCA-BayeSVMNet multi-class approach.

2.3.6 | Performance evaluation matrix

The five-performance metrics used to evaluate the performance of multiclass classification in SCA-BayeSVMNet are accuracy (Acc), sensitivity (Se), specificity (Sp), precision (Pre), and F-score (F-Scr.). True Positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN) parameters were used to calculate the performance metrics. These metrics derived from the confusion matrix were calculated using the equations in Equations (12)–(16).

\[
Se = \frac{TP}{TP + FN} \tag{12}
\]

\[
Sp = \frac{TN}{TN + FP} \tag{13}
\]
3 | EXPERIMENTAL RESULTS

In this study, two different datasets were examined. The CPBH dataset includes 4 classes, and CovCT-Findings includes 5 classes as mentioned earlier. All experimental analyses were evaluated with a ratio of 70% for the training set and 30% for the validation set. The results of both datasets are given separately below.

3.1 | CPBH dataset

The classification CT images of COVID-19, influenza A (H1N1), bacterial pneumonia and healthy lungs were performed with three pre-trained CNN models. In the first two steps, the CPBH dataset was trained with GoogLeNet, MobileNetV2 and ShuffleNet models. Figure 7 shows the training and validation accuracy graphs of the models.

The confusion matrices of the models and the obtained metric values are shown in Figure 8 and Table 5, respectively.

\[ \text{Pre} = \frac{TP}{TP + FP} \]  

(14)

\[ F - \text{Scr} = \frac{2 \times TP}{2 \times TP + FP + FN} \]  

(15)

\[ \text{Acc} = \frac{TP + TN}{TP + TN + FP + FN} \]  

(16)
In this step, firstly, the softmax function was used in the classification processes. The softmax classifier has the best success rate with MobilenetV2 model with 91.73%, as shown in Table 5. GoogLeNet also draws attention, which predicts bacterial pneumonia class and healthy lung class with 100% accuracy. Then the trained models were classified by SVM and the results are given in Figure 9 and Table 6.

The overall accuracy of GoogLeNet, MobileNetV2 and ShuffleNet is 94.96%, 96.04%, and 95.50% respectively, and is shown in Table 6. All models succeeded in distinguishing the class of healthy lungs from the other three classes with 100% accuracy. On the other hand, the class of bacterial pneumonia is distinguished from the other three classes with almost 100% accuracy, except that only one image is predicted as viral pneumonia false negative. The confusion matrices show that almost all misclassified images are from COVID-19 pneumonia and viral pneumonia.

![Confusion matrix of the pre-trained models using Softmax on CPBH dataset. (a) GoogLeNet, (b) MobileNetV2 and (c) ShuffleNet](image)

**TABLE 5** Performance metrics obtained with SoftMax classifier for the CPBH dataset

| CNN models | Classes          | Se (%) | Sp (%) | Pre (%) | F-Scr (%) | Acc (%) | Overall Acc (%) |
|------------|------------------|--------|--------|---------|-----------|---------|-----------------|
| GoogLeNet  | Bacterial        | 100    | 100    | 100     | 100       | 100     | 91.19           |
|            | COVID-19         | 85.95  | 93.80  | 87.36   | 86.65     | 91.19   |
|            | Healthy          | 100    | 99.78  | 99.46   | 99.82     | 99.82   |
|            | Viral (H1N1)     | 81.45  | 93.98  | 79.53   | 80.48     | 91.19   |
| MobileNetV2| Bacterial        | 100    | 100    | 100     | 100       | 100     | 91.73           |
|            | COVID-19         | 89.73  | 92.72  | 86.01   | 87.83     | 91.73   |
|            | Healthy          | 100    | 99.78  | 98.92   | 99.46     | 99.82   |
|            | Viral (H1N1)     | 78.23  | 95.83  | 84.35   | 81.17     | 91.91   |
| ShuffleNet | Bacterial        | 100    | 98.75  | 96.88   | 98.41     | 99.10   | 91.19           |
|            | COVID-19         | 85.95  | 94.65  | 88.83   | 87.36     | 91.77   |
|            | Healthy          | 100    | 99.78  | 98.92   | 99.46     | 99.82   |
|            | Viral (H1N1)     | 81.45  | 94.68  | 81.45   | 81.45     | 91.73   |

Abbreviation: CNN, convolutional neural network.

![Confusion matrix of the pre-trained models using support vector machine (SVM) on CPBH dataset. (a) GoogLeNet, (b) MobileNetV2 and (c) ShuffleNet](image)
classes. Although MobilenetV2 is two times less computationally intensive, it achieved higher accuracy than GoogLeNet on all F-Scr, which is an important evaluation criterion for imbalanced datasets. As a result of the second step, MobileNetV2 was found to be the most successful. The analysis results of the second step are shown in Table 6.

In the third step, the features obtained from the models of the first two steps were used for feature selection. SCA was used to reduce the dimensionality of the data in the obtained features, improve the classification performance and obtain optimal feature subsets. The K value is set to 3 in SCA so that the KNN classifier calculates the fitness function for each feature subset. The other selected parameters that are important for the performance of SCA are: the number of solutions is 5, the iteration value is 100 and the alpha value used to calculate the r1 value is 3.

In the feature selection process, SCA selected the most effective 142, 208 and 165 out of 1000 features of GoogLeNet, MobileNetV2 and ShuffleNet, respectively. We thank to SCA for being able to achieve the overall accuracy of GoogLeNet, MobileNetV2, and ShuffleNet is 97.84%, 96.22% and 95.86%, respectively, which is higher than the first two steps with the classification of the selected effective features. By combining a total of 515 effective features selected from the three models and applying SVM, the highest overall accuracy of 99.10% was obtained. At the end of the third step, the classification of images of healthy lungs and bacterial pneumonia was achieved with 100% accuracy. The confusion matrix obtained with the combined features shows that there are only five misclassified images belonging to COVID-19 pneumonia and viral pneumonia classes, and an increase in all performance metrics for viral pneumonia. According to the results of the third step, the feature selection methods improved the classification success of the proposed models. Confusion matrices obtained from the classification of the CPBH dataset using the SCA method are shown in Figure 10. The performance metrics obtained by SVM using the SCA method for the CPBH dataset are given in Table 7. If there is a class that cannot be adequately represented in a classification problem (by comparing the minority class with the majority class), we can speak of an unbalanced dataset (Lopez et al., 2012). By examining the number of representations of the classes in the CPBH dataset, we can consider it unbalanced. To avoid overfitting during training and to check the consistency of the previous results, the dataset was tested with a 5-fold cross-validation, yielding an overall accuracy of 98.71%. The analysis results of the fourth step are shown in Table 7.

In the fourth step, it is aimed to obtain better results by using the combined 515 features obtained from CPBH dataset, which we achieved the highest success in the previous step, with the hyperparameter optimization method. The method searches for the minimum classification error during 30 iterations, and tuned by Bayesian optimization algorithm to find a point that minimizes an objective function.

In this step, the best hyperparameter solution set with the lowest classification error was obtained with the Bayesian optimization algorithm, using the feature subset with the best results in the previous steps. The best hyperparameter optimization method achieved the best overall accuracy of 99.46%, and the results are shown in Figure 11. The analysis results of the fourth steps are also shown in Table 8.

### 3.2 CovCT findings dataset

The CovCT findings dataset includes 785 COVID-19 chest CT images from CPB and COVID-CT datasets to balance minority classes. These images were then analysed by the radiology department and found to contain five different classes of CT findings. Classification of these five classes (GGO, GGON, GGOC, CPP and Consolidation) was performed with three different pre-trained CNNs. Figure 12 shows the training and validation accuracy graphs of the models.
In the first two steps, the models were trained with the dataset and classified with the Softmax function, and the features extracted from the trained models were also classified with SVM. The training and validation accuracy graphs of the models trained with the dataset and the obtained confusion matrices are shown in Figures 12 and 13, respectively. At the end of the first step, when classifying the models with Softmax, MobileNetV2 had the highest overall accuracy success with 93.62%, as shown in Table 9. In the F-scr metric, which is an important parameter for unbalanced datasets, the MobilenetV2 has the highest success with 97.67% for the Consolidation class.

Then the trained models were classified by SVM and the results are given in Figure 14 and Table 10. As seen from the performance metrics table in Table 10, after the first two steps, ShuffleNet obtained the lowest result of 92.34%, followed by GoogLeNet with 92.76% while MobileNet with 94.46% achieved the highest result in the overall accuracy metric. When the class-based F-Scr values in the confusion matrices in Figure 14 are examined, GGOC and GGON have the lowest values with both MobileNetV2 and GoogLeNet, while CCP and Consolidation have the lowest F-Scr value with the ShufleNet model.

In the third step, the features obtained from the models of the first two steps were selected using SCA and the same selection parameters as for the CPBH dataset were used. With the help of SCA, 151, 268 and 264 more effective features were selected from 1000 features of GoogLeNet, MobileNet and ShuffleNet, respectively. As you can see from the confusion matrices in Figure 15 and from Table 11, metrics for overall accuracy achieved statistically significant performance, but MobileNetV2 got the best percentage with 96.60%. The class with the lowest accuracy is GGO as shown in Table 11. The fact that the sum of the FP and FN values of the GGO class is higher than the other class calculations caused these results. According to the results of the third step, the feature selection methods improved the classification success of the proposed models. Subsequently, when the selected 683 effective features from the three models were classified using SVM, the highest classification success of 98.72% was obtained. The consistency of the previous results was tested using 5-fold cross validation as the dataset-included minority and majority classes. As a result, classification success was 97.58%, emphasizing the effect of selected and combined features on classification success, as shown in the Table 11.

In the fourth step, we aimed to obtain better results by using the combined 683 features, which we achieved the highest success in the third step, with the hyperparameter optimization method. The method searches for the minimum classification error during 60 iterations, and tuned by Bayesian optimization algorithm to find a point that minimizes an objective function. It uses the best hyperparameter point, which is the best set of hyperparameter values, to obtain the minimum classification error. In this step, by using the best feature subset, the best hyperparameter solution set with the lowest classification error was obtained with the Bayesian optimization algorithm. Consequently, the best overall accuracy of 99.82% was achieved by using the best hyperparameter search range at the 60th iteration. The analysis and results of the fourth steps are shown in Figure 16 and Table 12.
### TABLE 7: Performance metrics obtained with SVM using the SCA method for the CPBH dataset

| CNN models                  | Feature number | Classes              | Se (%) | Sp (%) | Pre (%) | F-Scr (%) | Acc (%) | Overall Acc (%) |
|-----------------------------|----------------|----------------------|--------|--------|---------|-----------|---------|-----------------|
| GoogLeNet                   | 142            | Bacterial            | 100    | 100    | 100     | 100       | 100     | 97.84           |
|                             |                | COVID-19             | 95.68  | 98.92  | 97.79   | 96.72     | 97.84   |                 |
|                             |                | Healthy              | 100    | 99.78  | 98.92   | 99.46     | 99.82   |                 |
|                             |                | Viral (H1N1)         | 96.77  | 98.38  | 94.49   | 95.62     | 98.02   |                 |
| MobileNetV2                 | 208            | Bacterial            | 100    | 100    | 100     | 100       | 100     | 96.22           |
|                             |                | COVID-19             | 94.59  | 97.04  | 94.09   | 94.34     | 96.22   |                 |
|                             |                | Healthy              | 100    | 100    | 100     | 100       | 100     |                 |
|                             |                | Viral (H1N1)         | 91.13  | 97.69  | 91.87   | 91.50     | 96.22   |                 |
| ShuffleNet                  | 165            | Bacterial            | 99.35  | 100    | 100     | 99.68     | 99.82   | 95.86           |
|                             |                | COVID-19             | 94.05  | 97.04  | 94.05   | 94.05     | 96.04   |                 |
|                             |                | Healthy              | 100    | 100    | 100     | 100       | 100     |                 |
|                             |                | Viral (H1N1)         | 91.13  | 97.22  | 90.40   | 90.76     | 95.86   |                 |
| GoogLeNet & MobileNetV2 & ShuffleNet | 515 | Bacterial            | 100    | 100    | 100     | 100       | 100     | 99.10           |
|                             |                | COVID-19             | 98.92  | 99.19  | 98.39   | 98.65     | 99.10   |                 |
|                             |                | Healthy              | 100    | 100    | 100     | 100       | 100     |                 |
|                             |                | Viral (H1N1)         | 97.58  | 99.54  | 98.37   | 97.98     | 99.10   |                 |
| GoogLeNet & MobileNetV2 & ShuffleNet (5-fold cross-validation) | 515 | Bacterial            | 100    | 99.93  | 99.81   | 99.99     | 99.95   | 98.71           |
|                             |                | COVID-19             | 98.05  | 99.11  | 98.21   | 98.13     | 98.76   |                 |
|                             |                | Healthy              | 100    | 100    | 100     | 100       | 100     |                 |
|                             |                | Viral (H1N1)         | 97.11  | 99.17  | 97.11   | 97.11     | 98.71   |                 |

Abbreviations: CNN, convolutional neural network; SCA, sine-cosine algorithm; SVM, support vector machine.
The results obtained from the MobilnetV2 model, which achieved the highest overall accuracy, are visualized in Figure 17 with the OCS and Grad-CAM. OCS and Grad-CAM are methods that visualize the classification decisions of the network. Occlusion Sensitivity Maps focus more on specific regions that may be completely covered by the occlusion patch (Murugan et al., 2021). Since Covid-19 CT findings are mostly seen in both lobes of the lung, OCS was processed in this study to visualize multifocal CT findings. The success of the occlusion method in focusing multifocal lesions is highlighted in Figure 17. Figure 17 also shows that the occlusion method provides more sensitive results in the interpretation of multifocal lesions than Grad-CAM known from many studies focusing on the large main lesion (Oh et al., 2020).

![Fig 11](a) Min classification error graph using Bayesian optimization algorithm with combined 515 features, (b) best point hyperparameter optimization confusion matrix of 515 combined features

| TABLE 8 | Performance metrics obtained with optimizable SVM using the Bayesian optimization method for the CPBH dataset |
| --- | --- |
| CNN models | Feature number | Classes | Se (%) | Sp (%) | Pre (%) | F-Scr (%) | Acc (%) | Overall Acc (%) |
| GoogLeNet & MobileNetV2 & ShuffleNet | 515 | Bacterial | 100 | 100 | 100 | 100 | 100 | 99.46 |
| | | COVID-19 | 98.46 | 99.73 | 99.19 | 99.46 |
| | | Healthy | 100 | 100 | 100 | 100 | 100 |
| | | Viral (H1N1) | 98.39 | 99.77 | 99.19 | 98.79 | 99.46 |

Abbreviations: CNN, convolutional neural network; SVM, support vector machine.

The results obtained from the MobilnetV2 model, which achieved the highest overall accuracy, are visualized in Figure 17 with the OCS and Grad-CAM. OCS and Grad-CAM are methods that visualize the classification decisions of the network. Occlusion Sensitivity Maps focus more on specific regions that may be completely covered by the occlusion patch (Murugan et al., 2021). Since Covid-19 CT findings are mostly seen in both lobes of the lung, OCS was processed in this study to visualize multifocal CT findings. The success of the occlusion method in focusing multifocal lesions is highlighted in Figure 17. Figure 17 also shows that the occlusion method provides more sensitive results in the interpretation of multifocal lesions than Grad-CAM known from many studies focusing on the large main lesion (Oh et al., 2020).

4 | DISCUSSION

In this study, a new four-class dataset was created to distinguish COVID-19 from other types of pneumonia. When evaluating the performance metrics of the CPBH dataset, the accuracy is 99.46% for both the overall and COVID-19 class. It is well known that in similar studies, success is increased with various image enhancement techniques (Toğaçar et al., 2020). We do not resort to them in this study because we want to keep complexity at the most reasonable level by considering the ease of adaptation to the proposed applications. MobilNetV2, which has the lowest parameters and is generally created for mobile phone applications, is more successful in classifying 1000 extracted features, while GoogLeNet's success stands out after feature selection with SCA.

SCA is used in many studies for optimization purposes in applications such as engineering problems, network and classifier parameter optimization, image processing, real-world applications, object tracking and financial market forecasting. (Abualigah & Diabat, 2021). In this study, SCA is used to select the most distinctive features of the datasets and is often used to select the optimal network or classifier parameters when the optimization studies are examined. Many studies have used the variants of SCA, often supplemented by a second optimization algorithm, to avoid early convergence, determine the optimal solution, and improve exploration and exploitation capabilities (Abualigah & Diabat, 2021). The success of SCA, on the other hand, is demonstrated in Table 13 by the results obtained by selecting the concatenated features from the models proposed in this study using state-of-the-art metaheuristic optimization algorithms, and no SCA variants were required for this study. Since the classification
of SCA-BayeSVMNet is processed in the Matlab Classification Learner interface, where the accuracy of many classifiers is computed and ready for the user, the classification success of these data sets with SVM has been easily determined. As a result, the SCA-SVM model was developed, with the most distinguishing features and classified by the best successful classifier. Table 13 compares the efficiency of the proposed feature selection method by performing feature selection with the Genetic Algorithm (GA), Particle Swarm Algorithm (PSO), SMA and SCA according to the third step.

**FIGURE 12** Training and validation accuracy graphs of the CovCT-findings dataset with convolutional neural network (CNN) models. (a) GoogLeNet, (b) MobileNetV2 and (c) ShuffleNet

**FIGURE 13** Confusion matrix of the pre-trained models using Softmax on CovCT-findings dataset. (a) GoogLeNet, (b) MobileNetV2 and (c) ShuffleNet
When the feature selection results of SCA are evaluated based on the model in the SCA-BayeSVMNet, we see that the GoogLeNet model increases its accuracy by 2.95% by using 85% reduced features. Examining the publications in the literature that contain data comparable to SCA-BayeSVMNet, we find that SCA outperforms the feature selection algorithms in Table 14.

In the last step, the best results were obtained in the classification performed with the best point hyperparameters. As a result, SCA-BayeSVMNet achieved the most successful results, with an increase of 2.2% when hyper-parameter tuning was performed with Bayesian optimization of the selected most successful feature subset, which is 4.4% in Canayaz’s study (Canayaz, 2021a). However, a direct comparison of other studies with SCA-BayeSVMNet is not possible due to differences in the dataset, model structure, hyperparameters, and so forth. Comparison of CT modality studies is as follows in Table 15:

Li et al. (2020a) proposed a three-dimensional deep learning framework called COVNet to classify COVID-19, Community Acquired Pneumonia (CAP), and other non-pneumonia chest CT images. COVNet has a sensitivity of 96% and a specificity of 98% in detecting COVID-19 with an AUC of 0.96. Song et al. (2021) introduced an architecture by combining the ResNet50 backbone with a squeeze-and-excitation block (SE) to highlight important channels and suppress fewer of them. Using this structure, they distinguished COVID-19 CT images from bacterial pneumonia, healthy lung images, and typical viral pneumonia. An overall accuracy of 94% was achieved with the proposed approach. This article is the only publication in which we found that multiple types of pneumonia are classified with COVID-19 CT images with significant success. On the other hand, we would expect the proportions of CT finding types in the COVID-19 dataset to be specified.

The CNNs structure designed by Sen et al. (2021) is used to select features using a hybrid feature selection algorithm. It first selects features with MI and relief and then uses the Dragonfly algorithm (DA) as a wrapper filter. They achieved 98.39% accuracy in 2-class classification of...
TABLE 10 | Performance metrics obtained with SVM method for the CovCT-findings dataset

| CNN models     | Feature number | Classes      | Se (%) | Sp (%) | Pre (%) | F-Scr (%) | Acc (%) | Overall Acc (%) |
|----------------|----------------|--------------|--------|--------|---------|-----------|---------|-----------------|
| GoogLeNet      | 1000           | CCP          | 1      | 99.54  | 94.12   | 96.97     | 99.57   | 92.76           |
|                |                | Consolidation| 97.67  | 94.17  | 94.63   | 93.33     | 99.32   |                 |
|                |                | GGO          | 95.10  | 95.49  | 96.15   | 97.09     | 98.45   |                 |
|                |                | GGOC         | 86.67  | 100    | 100     | 92.86     | 99.15   |                 |
|                |                | GGON         | 84.75  | 97.16  | 90.91   | 87.72     | 94.04   |                 |
| MobileNetV2    | 1000           | CCP          | 98.28  | 99.54  | 99.13   | 98.70     | 99.10   | 94.46           |
|                |                | Consolidation| 95.35  | 98.44  | 93.18   | 94.25     | 97.87   |                 |
|                |                | GGO          | 98.04  | 96.99  | 96.15   | 97.09     | 94.83   |                 |
|                |                | GGOC         | 86.67  | 100    | 100     | 92.86     | 99.15   |                 |
|                |                | GGON         | 91.53  | 97.16  | 91.53   | 91.53     | 95.74   |                 |
| ShuffleNet     | 1000           | CCP          | 81.25  | 99.09  | 86.67   | 83.87     | 97.87   | 92.34           |
|                |                | Consolidation| 95.35  | 96.88  | 87.23   | 91.11     | 96.60   |                 |
|                |                | GGO          | 92.16  | 96.24  | 94.95   | 93.53     | 94.47   |                 |
|                |                | GGOC         | 86.67  | 100    | 100     | 92.86     | 99.15   |                 |
|                |                | GGON         | 94.92  | 97.16  | 91.80   | 93.33     | 96.60   |                 |

Abbreviations: CNN, convolutional neural network; GGO, ground glass opacities; GGOC, consolidation and ground-glass opacity; GGON, ground-glass opacities and nodule; SVM, support vector machine.

FIGURE 15 | Confusion matrices from classification of CovCT-findings dataset using the sine-cosine algorithm (SCA) method. (a) GoogLeNet, (b) MobileNetV2, (c) ShuffleNet, (d) combined 683 features, (e) 5-fold cross validation for combined 683 features

COVID-19 in SARS-CoV-2 dataset. Canayaz, (2021a) proposed a hybrid model for feature extraction by combining the Convolutional Block Attention Module (CBAM) and the first four versions of the EfficientNet models. The most relevant of the extracted features were then selected using Principal Component Analysis (PCA), and an overall accuracy of 98.45% was achieved when classified using SVM. Chattopadhyay et al. (2021) extracted features from different deep learning models and used two different public datasets for methodology. Using a meta-heuristic approach for feature selection, the Clustering-based Golden Ratio Optimizer, an accuracy of 99.31% was achieved for the deep features of ResNet18 for 2 classes. Carvalho et al. (2021) determined the most successful one among the 4 different CNN structures they offered, and then optimized the
## Table 11: Performance metrics obtained with SVM using the SCA method for the CovCT-findings dataset

| CNN models | Feature number | Classes      | Se (%) | Sp (%) | Pre (%) | F-Scr (%) | Acc (%) | Overall Acc (%) |
|------------|----------------|--------------|--------|--------|---------|-----------|---------|-----------------|
| GoogLeNet  | 151            | CCP          | 87.50  | 100    | 100     | 93.33     | 99.15   | 94.04           |
|            |                | Consolidation| 95.35  | 99.05  | 95.35   | 95.35     | 98.42   |
|            |                | GGO          | 100    | 92.48  | 91.07   | 95.33     | 95.74   |
|            |                | GGOC         | 93.33  | 100    | 100     | 96.55     | 99.57   |
|            |                | GGON         | 84.75  | 98.86  | 96.15   | 90.09     | 95.32   |
| MobileNetV2| 268            | CCP          | 87.50  | 100    | 100     | 93.33     | 99.15   | 96.60           |
|            |                | Consolidation| 100    | 99.48  | 97.73   | 98.85     | 99.57   |
|            |                | GGO          | 97.06  | 97.74  | 97.06   | 97.06     | 97.45   |
|            |                | GGOC         | 93.33  | 100    | 100     | 96.55     | 99.57   |
|            |                | GGON         | 96.61  | 97.73  | 93.44   | 95.00     | 97.45   |
| ShuffleNet | 264            | CCP          | 87.50  | 99.54  | 93.33   | 90.32     | 98.72   | 95.32           |
|            |                | Consolidation| 95.35  | 99.48  | 97.62   | 96.47     | 98.72   |
|            |                | GGO          | 98.04  | 95.59  | 94.34   | 96.15     | 96.64   |
|            |                | GGOC         | 93.33  | 100    | 100     | 96.55     | 99.57   |
|            |                | GGON         | 93.22  | 98.30  | 94.83   | 94.02     | 97.02   |
| GoogLeNet & MobileNet&2 & ShuffleNet | 683            | CCP          | 100    | 100    | 100     | 100       | 100     | 98.72           |
|            |                | Consolidation| 97.67  | 99.48  | 97.67   | 97.67     | 99.15   |
|            |                | GGO          | 100    | 98.50  | 98.08   | 99.03     | 99.15   |
|            |                | GGOC         | 100    | 100    | 100     | 100       | 100     |
|            |                | GGON         | 96.61  | 100    | 100     | 98.28     | 99.15   |
| GoogLeNet & MobileNet&2 & ShuffleNet (5-fold cross-validation) | 683            | CCP          | 96.43  | 100    | 100     | 98.18     | 99.75   | 97.58           |
|            |                | Consolidation| 95.83  | 99.53  | 97.87   | 96.84     | 98.85   |
|            |                | GGO          | 100    | 96.88  | 96.01   | 97.97     | 98.22   |
|            |                | GGOC         | 94.00  | 100    | 100     | 96.91     | 99.62   |
|            |                | GGON         | 95.96  | 99.66  | 98.96   | 97.44     | 98.73   |

Abbreviations: CNN, convolutional neural network; GGO, ground glass opacities; GGOC, consolidation and ground-glass opacity; GGON, ground-glass opacities and nodule; SCA, sine-cosine algorithm; SVM, support vector machine.
hyperparameters of the model using the Parzen method. Finally, the feature selection was made with Genetic Algorithm, and the obtained features were classified with four different classifiers (Multi-layer Perceptron classifier, SVM, XGBoost, Random Forest) and the overall accuracy of 99.7% was obtained. Toğaçar et al. (2022) proposed an approach using the ResNet models as feature extractors and selecting dominant activations from each model before SoftMax. They achieved an overall accuracy of 99.15% using the CPB (COVID-19, Viral Pneumonia and Bacterial Pneumonia) dataset.

**TABLE 12** Performance metrics obtained with Bayesian optimization method for the CovCT-findings dataset

| CNN models                      | Feature number | Classes  | Se (%) | Sp (%) | Pre (%) | F-Scr (%) | Acc (%) | Overall Acc (%) |
|---------------------------------|---------------|---------|--------|--------|---------|-----------|---------|-----------------|
| GoogLeNet & MobileNetV2 & ShuffleNet | 683           | CCP     | 100    | 100    | 100     | 100       | 100     | 99.82           |
|                                 |               | Consolidation | 100    | 99.48  | 97.73   | 98.85     | 99.57   |                 |
|                                 |               | GGO     | 100    | 100    | 100     | 100       | 100     |                 |
|                                 |               | GGOC    | 100    | 100    | 100     | 100       | 100     |                 |
|                                 |               | GGON    | 98.31  | 100    | 100     | 99.15     | 99.57   |                 |

Abbreviations: CNN, convolutional neural network; GGO, ground glass opacities; GGOC, consolidation and ground-glass opacity; GGON, ground-glass opacities and nodule.

**FIGURE 16** (a) Min classification error graph using Bayesian optimization algorithm with combined 683 features, (b) best point hyperparameter optimization confusion matrix of 683 combined features

**FIGURE 17** Comparison of occlusion sensitivity maps and grad-CAM in differentiating multifocal lesions in COVID-19 and CovCT findings classes. (a) Ground-glass opacities, (b) ground-glass opacities and nodule (c) crazy-paving-pattern (d) consolidation, (e) bacterial pneumonia (f) H1N1 viral pneumonia, (g) healthy lung
Pneumonia) dataset. On the other hand, SCA-BayeSVMNet also classified healthy lungs, bacterial pneumonia and H1N1 pneumonia from COVID-19 with an overall accuracy of 99.46%. Although GoogLeNet, which has the highest number of parameters among the models in SCA-BayeSVMNet, and has six times fewer parameters than the ResNet 101 model used by Toğuçar et al. (2022) with the highest parameters, SCA-BayeSVMNet achieved about 1% better overall accuracy, even without using any image processing methods.

On the other hand, as shown in the following studies, the prognosis of COVID-19 patients can be determined by examining the radiological findings. Some studies on predicting the prognosis of COVID-19 patients by examining chest CT findings at different stages of COVID-19 disease are as follows. Tekcan Sanli et al. (2021) investigated chest CT findings on initial hospital admission in patients who did not require intensive care for COVID-19 and patients who required at least 1 day of intensive care for COVID-19 or those patients who did not survive. In the results, consolidation and the increase in the number of lobes were thought to be more common in patients with severe disease and also crazy-paving-pattern (CPP), while central lung involvement were observed more frequently in the patients with severe disease. Li et al. (2020b) investigated the CT findings of 25 severe/critical cases and 58 ordinary COVID-19 patients and found that consolidation, linear opacities and crazy paving patterns were more frequent in severe/critical patients and significantly different from ordinary patients. Cau et al. (2021) studied 195 patients from clinical services and 23 patients from intensive care units. They found on CT imaging that ICU patients had higher incidence of lung consolidation mixed with GGO and bilateral lung involvement than non-ICU patients. Thus, the effectiveness of patient triage in the COVID-19 pandemic is not only to distinguish between COVID-19 and non-COVID-19 patients but also to determine whether patients require intensive care or outpatient treatment according to the classification of the CT findings identified. Therefore, this study was proposed as an application to help decide the patient’s intensive care or outpatient status by classifying chest CT findings into five different classes. Consequently, an overall accuracy of 99.82% was obtained with SCA-BayeSVMNet. On the other hand, CCP, GGO and GGOC scored 100% in F-Scr, followed by GGON, and Consolidation with 99.15% and 98.85%, respectively. Although the CovCT Findings dataset used in this study is similar to the dataset used by Toğaçar et al. (2022) except for the number of GGO images, SCA-BayeSVMNet achieved about 1% better overall accuracy, even without using any image processing methods.

### Table 13: Comparison among different metaheuristic optimization algorithms on the datasets

| Feature selection method | Dataset            | Number of selected features from the models | Overall Acc (%) |
|--------------------------|--------------------|--------------------------------------------|-----------------|
| GA                       | CPBH               | 1287                                       | 98.2            |
|                          | CovCT findings     | 1144                                       | 96.2            |
| PSO                      | CPBH               | 1398                                       | 98.4            |
|                          | CovCT findings     | 1357                                       | 96.8            |
| SMA                      | CPBH               | 182                                        | 98.4            |
|                          | CovCT findings     | 502                                        | 97.9            |
| SCA                      | CPBH               | 515                                        | 99.1            |
|                          | CovCT findings     | 683                                        | 98.72           |

Abbreviations: GA, genetic algorithm; PSO, particle swarm algorithm; SCA, sine-cosine algorithm; SMA, slime mould algorithm.

### Table 14: Comparison of feature selection methods in CT imaging COVID-19 studies

| Work reference          | Models / methods | Dataset         | Feature selection method | Reduced feature dimension (%) | Increased Overall Acc (%) |
|-------------------------|------------------|-----------------|--------------------------|-------------------------------|---------------------------|
| Sen et al. (2021)       | CNNs             | SARS-CoV-2 CT   | Mutual information (MI)  | 64.5                          | 1.61                      |
|                         |                  |                 | Relief-F dragonfly algorithm | 37                           | 2.51                      |
| Carvalho et al. (2021)  | CNNs             | SARS-CoV-2 CT   | Genetic algorithm        | 55.9                          | 0.4                       |
| SCA-BayeSVMNet          | Transfer learning| CPBH            | Sine-Cosine algorithm    | 85.8                          | 2.95                      |
|                         |                  | CovCT findings  |                          | 73.6                          | 3.2                       |

Abbreviations: CNN, convolutional neural network; CT, computed tomography; SCA, sine-cosine algorithm.

Pneumonia) dataset. On the other hand, SCA-BayeSVMNet also classified healthy lungs, bacterial pneumonia and H1N1 pneumonia from COVID-19 with an overall accuracy of 99.46%. Although GoogLeNet, which has the highest number of parameters among the models in SCA-BayeSVMNet, and has six times fewer parameters than the ResNet 101 model used by Toğuçar et al. (2022) with the highest parameters, SCA-BayeSVMNet achieved about 1% better overall accuracy, even without using any image processing methods.

On the other hand, as shown in the following studies, the prognosis of COVID-19 patients can be determined by examining the radiological findings. Some studies on predicting the prognosis of COVID-19 patients by examining chest CT findings at different stages of COVID-19 disease are as follows. Tekcan Sanli et al. (2021) investigated chest CT findings on initial hospital admission in patients who did not require intensive care for COVID-19 and patients who required at least 1 day of intensive care for COVID-19 or those patients who did not survive. In the results, consolidation and the increase in the number of lobes were thought to be more common in patients with severe disease and also crazy-paving-pattern (CPP), while central lung involvement were observed more frequently in the patients with severe disease. Li et al. (2020b) investigated the CT findings of 25 severe/critical cases and 58 ordinary COVID-19 patients and found that consolidation, linear opacities and crazy paving patterns were more frequent in severe/critical patients and significantly different from ordinary patients. Cau et al. (2021) studied 195 patients from clinical services and 23 patients from intensive care units. They found on CT imaging that ICU patients had higher incidence of lung consolidation mixed with GGO and bilateral lung involvement than non-ICU patients. Thus, the effectiveness of patient triage in the COVID-19 pandemic is not only to distinguish between COVID-19 and non-COVID-19 patients but also to determine whether patients require intensive care or outpatient treatment according to the classification of the CT findings identified. Therefore, this study was proposed as an application to help decide the patient’s intensive care or outpatient status by classifying chest CT findings into five different classes. Consequently, an overall accuracy of 99.82% was obtained with SCA-BayeSVMNet. On the other hand, CCP, GGO and GGOC scored 100% in F-Scr, followed by GGON, and Consolidation with 99.15% and 98.85%, respectively. Although the CovCT Findings dataset used in this study is similar to the dataset used by Toğaçar et al. (2022) except for the number of GGO images, SCA-BayeSVMNet achieved about 1% better overall accuracy, even without using any image processing methods.
| Work reference       | Models / methods                                | Dataset                                      | Train/test split ratio (%) | Modality | Year | Overall Acc (%) |
|---------------------|-------------------------------------------------|----------------------------------------------|----------------------------|----------|------|-----------------|
| Li et al. (2020a)   | Transfer learning Shared weights                | 1735 COVID-19                               | 80:10:10                   | CT       | 2020 | 96              |
|                     |                                                 | 1325 non-Pneumonia                          |                            |          |      |                 |
|                     |                                                 | 1735 Community-acquired Pneumonia            |                            |          |      |                 |
| Song et al. (2021)  | Transfer learning ResNet blocks                 | 2119 COVID                                  | 80:10:10                   | CT       | 2021 | 94              |
|                     |                                                 | 2158 Viral                                  |                            |          |      |                 |
|                     |                                                 | 1345 Bacterial                               |                            |          |      |                 |
|                     |                                                 | 504 Normal                                  |                            |          |      |                 |
| Sen et al. (2021)   | CNNs, hybrid feature selection, DA             | 1252 COVID                                  | 5-Fold                     | CT       | 2021 | 98.39           |
|                     |                                                 | 1230 Normal                                 |                            |          |      |                 |
| Canayaz (2021a)     | Transfer learning, CBAM model, Bayesian optimization | 1601 COVID                               | 70:30                      | CT       | 2021 | 98.45           |
|                     |                                                 | 1626 non-COVID                              |                            |          |      |                 |
| Chattopadhyay et al. | Transfer learning, feature selection, CGRO     | 1611 COVID                                  | 80:20                      | CT       | 2021 | 99.31           |
|                     |                                                 | 1627 Normal                                 |                            |          |      |                 |
| Carvalho et al. (2021) | CNNs, Hyperparameter optimization, Feature selection, GA | 1252 COVID                               | 80:20                      | CT       | 2021 | 99.70           |
|                     |                                                 | 1230 non-COVID                              |                            |          |      |                 |
| Toğaçar et al. (2022) | Transfer learning, Dominant activations selections, LIME method | 631 COVID                               | 70:30                      | CT       | 2021 | 99.15           |
|                     |                                                 | 417 Viral                                  |                            |          |      | 99.62           |
|                     |                                                 | 518 Bacterial                               |                            |          |      |                 |
|                     |                                                 | 436 GGO                                    |                            |          |      |                 |
|                     |                                                 | 144 Consolidation                           |                            |          |      |                 |
|                     |                                                 | 198 GGON                                   |                            |          |      |                 |
|                     |                                                 | 50 GGOC                                    |                            |          |      |                 |
|                     |                                                 | 56 CCP                                     |                            |          |      |                 |
| SCA-BayeSVMNet      | Transfer learning, feature selection-SCA, Bayesian optimization | 614 COVID                               | 70:30                      | CT       | 2022 | 99.46           |
|                     |                                                 | 415 Viral                                  |                            |          |      | 98.71           |
|                     |                                                 | 518 Bacterial                               |                            |          |      | 99.82           |
|                     |                                                 | 308 Normal                                 |                            |          |      | 97.58           |
|                     |                                                 | 337 GGO                                    |                            |          |      |                 |
|                     |                                                 | 144 Consolidation                           |                            |          |      |                 |
|                     |                                                 | 198 GGON                                   |                            |          |      |                 |
|                     |                                                 | 50 GGOC                                    |                            |          |      |                 |
|                     |                                                 | 56 CCP                                     |                            |          |      |                 |

Abbreviations: CNN, convolutional neural network; CT, computed tomography; GGO, ground glass opacities; GGOC, consolidation and ground-glass opacity; G Gon, ground-glass opacities and nodule; SCA, sine-cosine algorithm.
Despite all the successful results, the major drawback of the proposed structure is consisting of more than one model and is not an end-to-end model. Although the proposed method is a cost-effective tool for Covid-19 detection, one of the limitations of the paper is that the Bayesian optimization applied in this approach takes 240 and 290 s for the CPBH and CovCT Findings datasets, respectively. For this reason, we recommend that the optimization be automated so that it can be used as a user-friendly application. However, the successful results obtained with the proposed approach outweighed these disadvantages. Furthermore, an important innovation of our study is the determination of the prognosis of the disease by the successful classification of COVID-19 CT findings into five different classes.

5 | CONCLUSION

We are in the second half of 2022 and COVID-19 is in its third year with deaths and new cases. Despite the precautions and vaccines, the pace of the pandemic has not permanently slowed down and has caused many social and economic problems in the last 2 years. It is a fact that rapid detection of COVID-19 and isolation of the patient remains the most important factor in the fight against the pandemic. In this study, influenza A (H1N1), and bacterial pneumonia, which are difficult and common for physicians to distinguish from COVID-19, were successfully classified using the generated CPBH dataset. In addition, disease stage in the lungs was classified using the CT findings dataset obtained by classifying COVID-19 images according to CT findings (Consolidation, CCP, GGO, GGON, GGOC) by radiologists, and an attempt was made to create a decision support application for the responsible physicians, just like a report from the radiology department. For this purpose, the SCA algorithm was used to select the most valuable features from three pre-trained models developed mainly for mobile applications. Thus, with SCA-BayeSVMNet, the distinctive feature subset with the highest accuracy selected from the models was classified by SVM using the best point hyperparameter method. As far as we know, the hybrid deep learning structure in which the sine-cosine feature selection and SVM parameters are optimized with Bayes was used for the first time together with SCA-BayeSVMNet. The overall accuracy for the CPBH and CovCT Findings datasets was 99.46% and 99.82%, respectively. To see the effect of minority classes in the data sets on the stability of the validation, we performed 5-fold cross validation and achieved successful results.

Our primary goal in future studies is to examine the classification success of chest CT findings in different CNN structures, because of the lack of studies on chest CT findings. We also plan to test the success of the SCA-BayesSVMNet deep learning structure with a variant of SCAs on benchmark datasets. In conclusion, we believe that SCA-BayesSVMNet is a powerful model that can be used in the detection of various diseases and will be very useful for radiology departments when integrated with PACS systems.

ACKNOWLEDGEMENTS

The authors wish to thank Prof Dr Mehmet Ruhi Onur, an expert radiologist from Hacettepe University Radiology Department for his valuable help and contributions, we would like to thank the Department of Radiology Şişli Hamidiye Etfal Training and Research Hospital. We also like to thank Ms Bridget Veronica Donofrio for her significant contributions.

FUNDING INFORMATION

There is no funding source for this article.

CONFLICT OF INTEREST

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

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

ORCID

Nedim Muzoğlu https://orcid.org/0000-0003-1591-2806
Ahmet Mesrur Halefoğlu https://orcid.org/0000-0002-2054-3550
Muhammed Onur Avci https://orcid.org/0000-0001-5211-2619
Melike Kaya Karaaslan https://orcid.org/0000-0001-9078-8468
Bekir Siddik Binboga Yarman https://orcid.org/0000-0003-1562-5524

REFERENCES

Abualigah, L., & Diabat, A. (2021). Advances in sine cosine algorithm: A comprehensive survey. In Artificial Intelligence Review (Vol. 54, pp. 2567–2608). Springer Netherlands. https://doi.org/10.1007/s10462-020-09909-3
AUTHOR BIOGRAPHIES

Nedim Muzoğlu, who received his undergraduate degree in Electronics Engineering from the University of Uludag, Turkey in 2000, and completed his master's degree in Biomedical and Electronic Engineering. He studied Artificial Intelligence, Machine Learning, Deep Learning and Clinical Engineering during his doctoral studies. In 2002/2010, he worked in the IT departments in the field of telecommunications. He has been working as a Clinical Engineering Department Manager in Public Hospitals Institutions since 2010.

Ahmet Mesrur Halefoglu is currently working as a resident specialist of Radiology and a Professor at Sisli Etfal Training and Research Hospital, Istanbul, Turkey. He completed his residency between 1992 and 1997 at Sisli Etfal Training and Research Hospital, Istanbul, Turkey. He has served as a postdoctoral research fellowship at Johns Hopkins Hospital, Baltimore, MD, USA, between 1998 and 1999. He has also been a research observer at Johns Hopkins Hospital in the years 2002 and 2004. He is a member of the Turkish Society of Radiology and the European Society of Radiology.

Muhammed Onur Avci completed his MSc and PhD degrees in Biomedical Engineering. He has studied in the field of Biomaterials during his MSc and PhD. He has been working as a Biomedical Engineer in Public Hospitals Institutions.

Melike Kaya Karaaslan, who received her undergraduate degree in Physics Engineering from the University of Ankara, Turkey in 2004, and her MSc in Biomedical Engineering from the University of Kocaeli. She still continues her PhD at Kocaeli University. Her research focuses on the diagnostic radiology and medical physics. She has been working as a Physics Engineer in Public Hospitals Institutions since 2013 and carries out quality control and acceptance tests of diagnostic radiology devices.

Siddik Yarman, who completed his graduate studies in Electronics and Mathematics at Stevens Technical University, and received his PhD in Electronic Systems Theory and Mathematics from Cornell University. He still continues his scientific studies at Istanbul University, Istanbul Technical University, Wuhan University of Technology in China, University of Malaysia and Lincoln University in England.

How to cite this article: Muzoğlu, N., Halefoglu, A. M., Avci, M. O., Karaaslan, M., & Yarman, B. S. B. (2023). Detection of COVID-19 and its pulmonary stage using Bayesian hyperparameter optimization and deep feature selection methods. Expert Systems, 40(1), e13141. https://doi.org/10.1111/exsy.13141