ORIGINAL ARTICLE

COVID-19 diagnosis on CT images with Bayes optimization-based deep neural networks and machine learning algorithms

Murat Canayaz[1] · Sanem Şehribanoğlu[2] · Recep Özdağ[1] · Murat Demir[3]

Abstract

Early diagnosis of COVID-19, the new coronavirus disease, is considered important for the treatment and control of this disease. The diagnosis of COVID-19 is based on two basic approaches of laboratory and chest radiography, and there has been a significant increase in studies performed in recent months by using chest computed tomography (CT) scans and artificial intelligence techniques. Classification of patient CT scans results in a serious loss of radiology professionals' valuable time. Considering the rapid increase in COVID-19 infections, in order to automate the analysis of CT scans and minimize this loss of time, in this paper a new method is proposed using BO (BO)-based MobilNetv2, ResNet-50 models, SVM and kNN machine learning algorithms. In this method, an accuracy of 99.37% was achieved with an average precision of 99.38%, 99.36% recall and 99.37% F-score on datasets containing COVID and non-COVID classes. When we examine the performance results of the proposed method, it is predicted that it can be used as a decision support mechanism with high classification success for the diagnosis of COVID-19 with CT scans.

Abbreviations

CT	Computed tomography
BO	Bayesian optimization
WHO	The World Health Organization
RT-PCR	Real-time reverse transcription polymerase chain reaction
AI	Artificial intelligence
DL	Deep learning
DNN	Deep neural networks
ML	Machine learning
IRRCNN	Inception recurrent residual neural network
MODE	Multi-objective differential evolution
CNN	Convolutional neural networks
ANN	Artificial neural network
BCNN	Bayesian convolutional neural networks
PNN	Probabilistic neural network
RBFNN	Radial basic function neural network
GRNN	Generalized regression neural network
R	Correlation coefficient
RMSE	Square root of the mean square of the errors
HP	Hyperparameters
SVM	Support vector machine
GS	GridSearch CV
RS	Random Search CV
LSTM	Long short-term memory
BNN	Bayesian neural network
DE	Deep ensemble
AF	Acquisition function
GP	Gaussian processes
RFR	Random forest regression
TPE	Tree parzen estimators
SM	Surrogate model
PI	Probability of improvement
EI	Expected improvement
UCB	Upper confidence bound
kNN	K-nearest neighbor

✉ Murat Canayaz
  mcanayaz@yyu.edu.tr

1   Department of Computer Engineering, Van Yuzuncu Yil University, 65100 Van, Turkey

2   Department of Econometrics, Van Yuzuncu Yil University, 65100 Van, Turkey

3   Department of Software Engineering, Mus Alpaslan University, 49100 Mus, Turkey

1 Introduction

Coronaviruses are among diseases that have threatened our world for years, spreading in the form of epidemics among most human and animal species. An outbreak of coronavirus infection, which severely affected the population in Wuhan the capital of China's Hubei Province, began in December 2019. The World Health Organization (WHO) proclaimed a worldwide emergency on January 30, 2020, officially announced a new coronavirus, named COVID-19, and classified a pandemic on March 11, 2020. The coronavirus can infect birds, mammals, and humans. However, bats do not become infected even though they host the coronavirus [1]. As of October 22, 2021, more than 241 million cases were recorded worldwide, causing the death of more than 4.9 million infected people. The top three regions with the highest number of COVID-19 cases and more than weekly 2 million cases are described on the official WHO web page as the Americas, South-East Asia, and Europe, respectively. At the time of the first spread of COVID-19, the Chinese government announced that the diagnosis of COVID-19 could be confirmed with real-time reverse transcription polymerase chain reaction (RT-PCR) [1]. However, the fact that RT-PCR tests gave extreme false-negative results made the reliability of these tests questionable [2]. The inability to detect infected people and to start the necessary treatment on time increases both the risk of transmission of COVID-19 and the risk of death during this process.

COVID-19 tests are done to detect viruses or antibodies. The diagnosis of COVID-19 is made based on two basic approaches. The first is laboratory-based approaches that include nucleic acid testing, antigen tests, and serology tests. The second approach considers lung imaging-based diagnostic approaches such as X-rays and computed tomography (CT) scans[3]. Laboratory tests are performed on samples obtained through nasopharyngeal swab, throat swab, and sputum. One of the most common diagnostic methods used is nasopharyngeal swab [4]. X-rays and CT scans are used as important diagnostic approaches for the verification of patients suspected of being infected with the virus. Since COVID-19 can affect the lungs in a similar way to many diseases such as pneumonia on images obtained with X-rays and tomography, a definite positive COVID-19 result may not be reached based on findings obtained only from lung images without clinical diagnosis [5]. Along with clinical diagnoses, a chest CT scan has high sensitivity in revealing definitive diagnosis of COVID-19. So, the diagnosis of COVID-19 can be made by combining the symptoms and laboratory findings of the infected person with radiological imaging techniques. The radiological features of COVID-19 can be detected by X-rays and CT scans, which are radiological imaging techniques. Radiologists mostly prefer X-ray chest images for the diagnosis of COVID-19 disease. However, chest CT scans are used for more accurate detection, since X-ray devices cannot accurately distinguish soft tissues in chest images [6].

There has been a considerable increase in studies in the literature to diagnose COVID-19 from chest CT scans. Looking at these studies, the diagnostic approaches for COVID-19 are examined in two general categories. The first is based on laboratory-based approaches, while the other is based on medical imaging instruments such as X-rays and CT scans. When the studies performed are examined, chest CT-scans are used as a priority tool in the clinical process because of successful results for the diagnosis of COVID-19 [7]. Today, Artificial Intelligence (AI)-based Machine Learning (ML) and Deep Learning (DL) technologies are used for the diagnosis of SARS-CoV-2 in the medical field by using chest CT scans. ML Algorithms are used to help radiologists make decisions in the process of diagnosing COVID-19 from images on chest CT-scans. In addition, Deep Neural Networks (DNN) are preferred by researchers for imaging-based problems that require feature extraction, such as the diagnosis of COVID-19.

To summarize the literature studies about CT scans related to the diagnosis of COVID-19, Alom et al. [8] studied a total of 425 CT-scans developing the Inception Recurrent Residual Neural Network (IRRCNN) and NABLA-N models for COVID-19 detection and segmentation of CT scans, respectively. Silva et al. [9] developed the Efficient Deep Learning Technique to evaluate each COVID-19 chest CT scan independently and to process CT images of different quality when using different CT devices depending on the environmental conditions. To diagnose COVID-19 from chest CT scans and to classify the lesions by segmenting, the following models were developed: a Multi-task Deep Learning model by Amyar et al. [10], Weakly Supervised DL Framework by Wang et al. [11], a new Deep Transfer Learning model based on DenseNet201 by Jaiswal et al. [12], a new DL model was developed using multi-objective differential evolution (MODE) and convolutional neural networks (CNN) by Singh et al. [3].

In the literature review for our study, we did not come across any study in which a Bayes optimization (BO)-based approach was applied to CT scans. Therefore, the motivation for this paper is to find the hyperparameters (HP) using BO by both DNN and ML algorithms and to be the first to illustrate their high performance. What makes our study important is that it tries to obtain the best models by finding the most optimum results in a particular search area while choosing the most suitable parameters for both DNN and ML. Moreover, it is planned to contribute to real-time

disease diagnosis by implementing these models on web. Generalization of models is important for performance in real-time systems. Ultimately, we have to verify that our work, which is expected to aid expert opinion, is working correctly. For this reason, GridSearchCV (GS) [13] is used as an alternative to BO for parameter optimization of ML algorithms. This is an important argument that shows that our study is usable.

We may summarize the contribution of our study to the field with a few points:

- Providing a decision support mechanism that helps expert opinion with high accuracy using BO-based models.
- Showing that datasets created from CT scans can give different results in terms of model and features.
- Examining the contribution of HP regulation to performance.
- It offers a fast-integrated approach for real-time disease diagnosis.

2 Related work

In this part of our study, literature studies published using BO about COVID-19 data in the field of artificial neural network (ANN) and DL are presented.

Cabras [14] proposed a semi-parametric approach to estimate the evolution of COVID-19 disease in Spain. It combined DL techniques with Bayesian Poisson-Gamma model. The resulting general model enabled prediction of the future variation of the disease sequences in all regions and the results of the final future scenarios. The overall success rate was found to be 95% in this study. Ghoshal et al. [15] studied a large number of PA chest radiography images. They attempted to improve diagnostic performance by using Dropweights-based Bayesian convolutional neural networks (BCNN) and DL methods. In comparison with standard Convolutional Neural Networks and BCNN, the accuracy rate was shown to be higher (over 92%) for BCNN. Dhamodharavadhani et al. [16] used SNN models such as probabilistic neural network (PNN), radial basic function neural network (RBFNN), and generalized regression neural network (GRNN), which include the Bayesian decision rule and the predictors of the Parzen probability density function. They attempted to predict future COVID-19 deaths in India using two separate datasets. In the study, R (correlation coefficient) and RMSE (square root of the mean square of the errors) were studied. As a result, PNN was observed to give better results for both criteria. Ucar et al. [17] adapted Squeeze-Net for the diagnosis of COVID-19 by combining with BO. The BO method was used for the optimization of HP. The

proposed method classified three classes of X-ray images labeled Normal, Pneumonia and COVID. It classified the data in the normal class with 98.04% accuracy, the data in the Pneumonia class with 96.73% accuracy, and the data in the COVID-19 class with 100% accuracy. Arman et al. [18] optimized the HP values of VGG16, MobileNetV2, InceptionV3, and Xception architectures using BO to detect COVID-19 on chest X-ray images. The proposed method classified three classes of X-ray images labeled Normal, Pneumonia and COVID. It classified the data in the normal class with 100% accuracy, the data in the Pneumonia class with 100% accuracy, and the data in the COVID-19 class with 98.3% accuracy. Majid et al. [19] designed a new series network consisting of five convolution layers to replace CNNs. This CNN model was designed as a deep feature extractor. The inferred deep distinguishing features were used to feed ML algorithms, the k-nearest neighbor, support vector machine (SVM), and decision tree. The HPs of ML models were optimized using the Bayes optimization algorithm. The best accuracy rate was achieved at 98.7% with SVM. Stefan et al. [20] processed a large number of reasonable hypothetical scenarios generated by a simulation program with ANN. After completion of the training phase, Bayesian posterior distributions were estimated. The network created has three levels. In the first level, feature extraction was performed from the observation data, in the second, preprocessed time series of different lengths were reduced to fixed-size statistical summaries, and in the third, a Bayesian-based inference network was used to extract parameters from the observations with summary statistics. At the end of the study, the number of newly infected, newly recovered and new deaths was estimated with 95% success. Ratnabali et al. [21] proposed a shallow long short-term memory (LSTM)-based neural network to estimate the COVID-19 risk situation of countries. The BO framework was used to optimize and design country-specific networks. Each network created with BO was trained using a maximum of 5000 iterations. The data for each country were used separately to create a country-specific optimized network and an average of 77.6% accuracy was obtained in country-specific datasets. Ankur et al. [22] showed that the uncertainty estimation decreases when the amount of training data is low with Bayesian Neural Network (BNN) and Deep Ensemble (DE) models. The approach enabled the basic uncertainties of the estimation for the deep K-Nearest neighbor (kNN) classifier to be accurately measured. Diagnosis of COVID-19 from chest X-rays was shown to measure uncertainty in a superior way compared to the latest technology. The proposed model was tested on three different datasets (COVID-19 training, COVID-19 Unseen and Shoulder). It achieved an accuracy rate of 99.9% for the first dataset, 60% for the second and 50.1% for the

third. Gao et al. [23] used a total of 1918 CT scans in their study where they developed an approach called double-branched combination network (DCN) with less attention module for Covid-19 diagnosis and segmentation. The highest accuracy rate for classification was stated as 96.74%. Panwar et al. [24] have considered three datasets known as 1) COVID-chest X- ray, 2) SARS-COV-2 CT-scan, and 3) Chest X-ray Images (Pneumonia). According to the results obtained, the proposed deep learning model can detect COVID-19 positive cases within ≤ 2 s, faster than the currently used RT-PCR tests for the detection of COVID-19 cases. In their study, He et al. [25], in which they created a publicly available dataset containing hundreds of CT scans, developed sample efficient deep learning methods that can obtain high diagnostic accuracy of COVID-19 from CT scans even when the number of CT images is limited. Specifically, they propose a self-transition approach that synergistically integrates comparative self-supervised learning and transfer learning to learn powerful and unbiased feature representations to reduce the risk of overfitting. Wu et al. [26] have developed a new Joint Classification and Segmentation (JCS) system to perform real-time and explainable CT chest CT diagnosis of COVID-19. JCS obtains an average sensitivity of 95.0% and a specificity of 93.0% on the classification test set.

3 Methodology

3.1 Hyperparameter tuning

HPs have an important place in both ML and DL algorithms as they aim to achieve the best performance in ML algorithms [17] because ML algorithms rarely contain parameters. HPs also have an important place in training algorithms [27]. Especially in CNN studies, it can be time consuming considering the size of the model, activation function, optimization algorithm to be used and the structure of the network [28]. BO is a convenient approach in studies that take a long time [27, 29].

The HP optimization method is collected under two headings as manual and automatic search. Manual search is based on an expert's experience. As a result of the increase in the number of hyperparameters and the value range, the possibility of making an error increases [27]. Trial-and-error processes slow down the optimization process [28]. HPs optimization were suggested to reduce the possibility of errors and speed up optimization. HPs optimization aim to reduce human effort in ML algorithms, to increase current performances and to make studies repeatable [27]. Three techniques are often used in ML algorithms to optimize HPs; these are Grid Search CV (GS), Random

Search CV (RS) [27, 28] and Informed search methods, respectively.

3.1.1 GridSearch CV

The GS method is a full factorial design. It checks all possible states to optimize parameters [27, 28]. A finite set of values is created for each HP and the Cartesian products of these sets are evaluated [27]. Large numbers of HPs and the search field cause an increase in time [28]. RS is more efficient than GS in a high-dimensional space. However, the RS method is unsuccessful in training complex models [27].

3.1.2 Bayesian optimization

Bayesian optimization is the most popular informed search method. It is faster than GS and RS methods. BO [28, 29] is preferred, especially considering the computational density encountered in DL algorithms. BO is an approach to optimize objective functions that take a long time to evaluate [29–31]. BO is a model-based HP optimization algorithm [31–33] based on the iterative update of the function to be optimized.

If we define $f : x \rightarrow y$, y = f(x) and f as $D = \{x_i, y_i\}_{=1}$ a black-box function, BO is a probability-based surrogate model (SM) to maximize an acquisition function (AF) that will decide which point to select [32–34]. An unknown model (f) is considered to have a black box property if it does not have a functional form [34, 35], and the optimization problem related to the HPs of this model is as in Eq. 1.

$$x^* = \underset{x \in X}{\operatorname{argmax}} f(x) \tag{1}$$

The purpose of this optimization problem is to find global maximization (or minimization) at the sampling point for the function f. Here X represents the search space of x. BO is essentially a Bayesian approach based on Bayes' theorem. The purpose of Bayesian approaches is to use the information obtained from the data as prior information and to reveal how the existing information will be updated with the obtained posterior information [36, 37].

Using the Bayesian approach, an SM is created in BO [27, 28]. As an SM, it usually uses one of the gaussian processes (GP), random forest regression (RFR) or tree Parzen estimators (TPE) methods. In studies, GP is preferred which takes advantage of the properties of normal distribution and has a stochastic process. GPs are preferred due to their smooth and well-calibrated uncertainty estimates and closed-form computability properties [33, 33]. GPs predict a distribution for each HP setting rather than a single value [27, 28]. GP is considered to be the mean of function μ, covariance kernel K, $f \sim GP(\mu, K)$. In this

study, the kernel function of the Matern (v = 5/2), which is widely used to define the covariance of two points at $d(x_i, x_j)$ unit distance, was preferred.

$$K = \left(1 + \frac{\sqrt{5}d}{\rho} + \frac{5d^2}{3\rho^2}\right) \exp\left(-\frac{\sqrt{5}d}{\rho}\right) \quad (2)$$

where $d(x_i, x_j)$ is Euclidean distance and ρ and ν are covariance parameters.

The process of maximizing the posterior process obtained by combining SM and prior knowledge [27, 28, 32] in BO is called AF (u). AF enables BO to make educated predictions [37, 38]. A proper AF should be easy to assess or maximize, and there should be a tradeoff between exploration and exploitation. Probability of Improvement (PI), Expected Improvement (EI) and Upper Confidence Bound (UCB) are commonly used for AF. PI was used in this study. If we define the best available observation as (x^+), which maximizes the possibility of improvement, Eq. 3 written as

$$x^+ = \operatorname*{argmax}_{x \in X} u(x|D) = \operatorname*{argmax}_{x \in X} f(x) \quad (3)$$

PI tries to find points that will prevail over the best available value. The search is terminated when the repeat count of the algorithm reaches the maximum, and where $\Phi(\cdot)$ is the normal cumulative distribution function. This function, defined as PI, tries to find a point where improvement probability is maximized [27, 28] by adding a ε trade-off parameter [38, 39].

$$PI(x) = P(f(x) \geq f(x^+) + \varepsilon) = \Phi\left(\frac{\mu(x) - f(x^+) - \varepsilon}{\sigma(x)}\right) \quad (4)$$

where ε is a parameter that tunes the tradeoff between exploration and exploitation.

The BO process continues to iterate until the maximum value is reached. BO makes this search efficient, using all the information it gets from the optimization history [39]. The pseudo code of BO is given in Algorithm 1.

Algorithm 1 Bayesian optimization

1: **for** n=1,2,… **do**
2: select new x_{n+1} by optimizing acquisition function u
$$x_{n+1} = \operatorname*{argmax}_{x} u(x|D_n)$$
3: query objective fuction to obtain y_{n+1}
4: augment data $D_{n+1} = \{D_n(x_{n+1}, y_{n+1})\}$
5: update statistical model
6: **end for**

3.2 Deep neural networks

DL is an ML subfield about algorithms inspired by the structure and function of the brain called NNA. DNN, on

the other hand, is a tool in which DL applications that contain layer structures such as convolution, pooling, and fully connected layer are carried out. Many models were developed through these layers. The models we used in our study were developed using these layers. Detailed information about these layers can be found in [40]. The models developed have their own features rather than these layers.

3.2.1 ResNet-50

ResNet-50 network architecture has 4 stages. Each ResNet [41] architecture performs initial convolution and maximum pooling using 7 × 7 and 3 × 3 core sizes, respectively. Each layer of a ResNet consists of several blocks. In our study, 1000 features were extracted by using the "fc1000" layer in this model for feature extraction.

3.2.2 MobileNetv2

MobileNetv2 [42] offers a new CNN layer with the inverted residual and linear bottleneck layer that provides high accuracy and performance in mobile and embedded video applications. Especially developed for devices with low computing power, this model reduces the complexity cost of the network. In addition, the model size decreases. In our study, again 1000 features were extracted by using the "logits" layer in this model.

Structures of the models are summarized in Table 1.

3.3 Machine learning algorithms

3.3.1 Support vector machine

SVM is one of the basic approaches for supervised learning. Additionally, it is widely used in classification and regression applications, and also frequently in clustering [43, 44], feature selection [45–47], feature extraction [48, 49], etc. SVM, based on the statistical learning theory [50, 51], is a distribution independent learning algorithm since it does not require joint distribution function information. The basic working principle of the algorithm is to determine a hyperplane that can optimally separate the pixels belonging to two classes from each other [52]. SVM applies the principle of minimizing the structure risk to minimize empirical error and learner complexity [50]. In this study, C, degree, and kernel parameters in SVM were obtained by the HP tuning process. SVM is demonstrated in Fig. 1.

Equations (5) and (6) represent formulas for a line or hyper plane, respectively. SVM should find weights so that the data points are separated according to a decision rule.

Table 1 Structure of models

Models	Model description	Image size	Optimization	Mini Batch Size
MobileNetv2	32 conv + 19 residual bottleneck layers	224×224	Stochastic Gradient Descent (SGD)	128
ResNet-50	49 conv + 1 fc layers	224×224	Stochastic Gradient Descent (SGD)	128



Fig. 1 Support vector machine
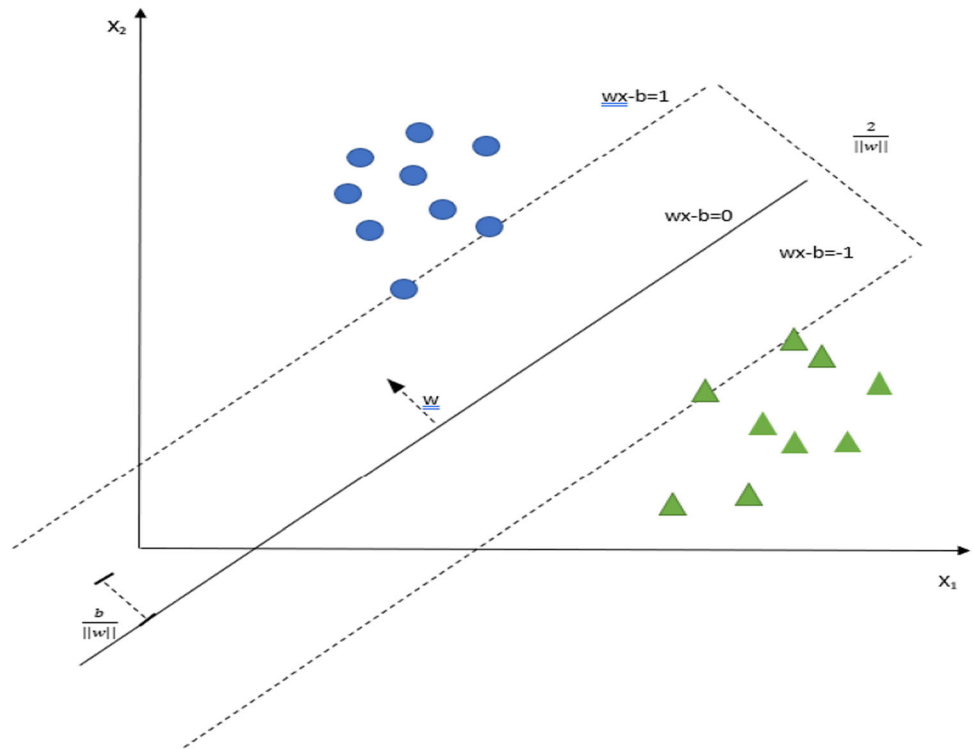
$$wx + b = 0 \tag{5}$$

$$y = mx + b \tag{6}$$

Here w is a weight vector, x is input vector, b is bias. C is a parameter that changes depending on the optimization. The higher the C value, the tighter the margin and care is taken to minimize the number of misclassifications. As the value of C decreases, it is allowed to overflow the classes because it becomes the goal of SVM to keep the margin between the two classes maximum [53]. The degree parameter determines the flexibility of the decision boundary. The lowest order polynomial is the linear kernel, which is not sufficient when there is a nonlinear relationship between the features. Also, increasing these parameters leads to higher training times. Kernel parameters have a very important influence on the decision-making boundaries. Kernel parameters select the type of hyperplane. The linear one uses a linear hyperplane. rbf, sigmoid and poly use a nonlinear hyperplane.
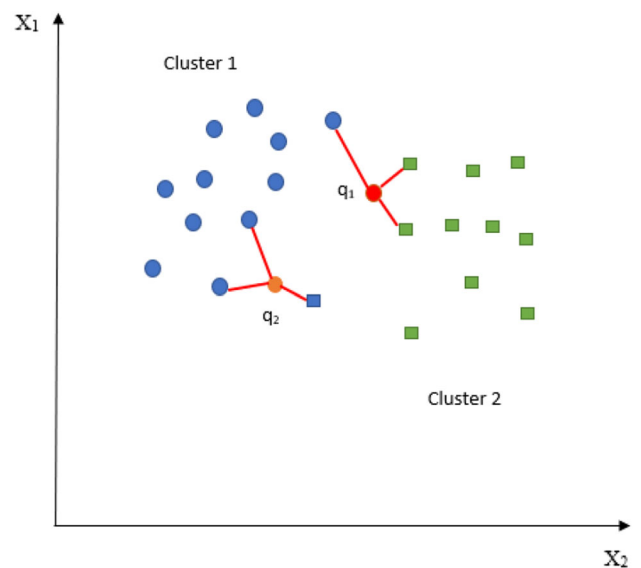
Fig. 2 A simple example of 3-nearest neighbor classification

Fig. 3 Flowchart of the BO-based proposed approach
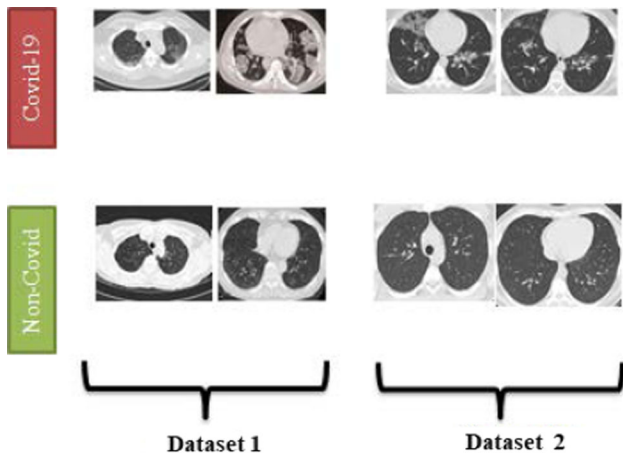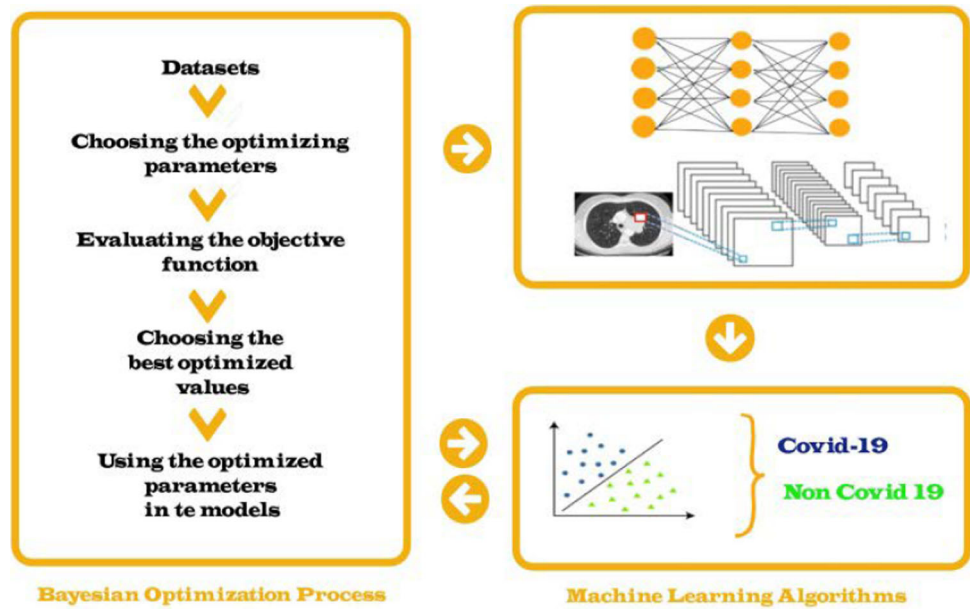
Fig. 4 CT images that COVID-19 is infected and is non-infected in the used dataset (Dataset1 and Dataset2)

Table 2 Sample sizes used for Training and Testing in Dataset1 and Dataset2

Datasets	Training Samples	Test Samples	Total
Dataset 1	558	140	698
Dataset2	1968	492	2460
Mixed Dataset	2527	631	3158

3.3.2 k-Nearest neighbor algorithm

kNN algorithm is a nonparametric classification method. It is a method with simple structure but is effective [54]. The kNN classifier tries to classify the data by assigning observation data of unknown classes to the class with the most similar examples [55]. The first value to be determined in the kNN algorithm is the distance between data. The distance measurement methods generally used for this are: Euclidean, Manhattan and Minkowski methods.

The most used Euclidean distance method in practice, between sample X_i and X_j is defined as [56]:

$$(X_i, X_j) = \sqrt{(X_{i1} - X_{j1})^2 + (X_{i2} - X_{j2})^2 + \ldots + (X_{in} - X_{jn})^2} \tag{7}$$

The second value to be determined is the parameter k. It is effective in determining the number of neighborhoods. Choosing an appropriate k value for kNN significantly affects the success of the classification. There are many ways to choose the K value. However, the simplest is to run the algorithm with different k values to select the one with the best performance [57]. Choosing a small value of k will increase the number of classes and create classes that do not exist. If the value of k chosen is too large, the classes will be fewer than they should be and thus the error values of the classes are increased. In general, larger k values are more resistant to potential noise in the data and make the

Fig. 5 Confusion matrix for 2-class

boundaries between classes smoother [58]. In this study, n_neighboor (k), weights, and metric parameters were obtained by tuning.

Figure 2 represents the neighborhood for 2 sample points at k = 2.

If k = 2, the q_1 point is largely labeled cluster 2 and the q_2 point is labeled cluster 1.

3.4 Proposed approach

This study consists of three steps. In the first step, DNN was trained separately with all datasets. The learning rate, momentum, and L2 regularization parameters needed for the SGD optimization algorithm used to update the weights in the training process were found through BO. The "bayesopt" function in MATLAB was used to find these parameters. The following value ranges were used to find suitable values for these parameters; Initial Learn Rate: [1e-2, 1], Momentum: [0.8, 0.98], and L2 Regularization: [1e-10, 1e-2]. These parameters were run at the given ranges using BO. The parameter values that provide the best value for the trained networks at the end of the operation are recorded for the network. The network created with these values was used in the second stage feature extraction.

The second step is the feature extraction step from DNN models trained with datasets. DNN feature extraction is obtained from the activation of the desired layer. Network activations are used for feature extraction. In the study, 1000 properties of each image in the dataset were extracted by using the "Logits" layer for MobilNetv2 and the

(a) MobileNetv2
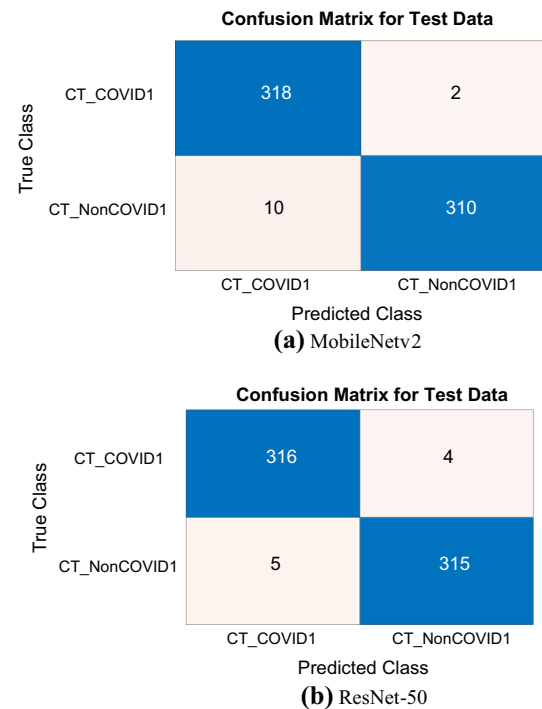
(b) ResNet-50

Fig. 6 Confusion Matrix for Mixed Dataset at the MobileNetv2 and the ResNet-50

"fc1000" layer for ResNet. Feature extraction was done separately for each dataset and saved as a *.mat file.

In the last step, the extracted features were classified by ML algorithms using Python language. BO was used to find the HPs for the ML algorithms. In addition to the BO method, the aim was to compare the results by using the GS method in finding the parameters. Five-fold cross

Table 3 Optimal DNN results obtained by using BO at the experiment1

Models	Classes	Precision	recall	f1-score	Accuracy	Parameters		
						Learning Rate	Momentum	L2 Regularization
Dataset1								
MobileNetv2	COVID	0.9855	0.9714	0.9783	0.9786	0.01	0.8786	0.000361
	Non-COVID	0.9718	0.9857	0.9787				
ResNet-50	COVID	0.9459	1	0.9722	0.9714	0.072	0.8807	0.0065
	Non-COVID	1	0.9428	0.9705				
Dataset2								
MobileNetv2	COVID	0.9959	1	0.9979	0.998	0.01	0.8552	0.0004
	Non-COVID	1	0.9959	0.9979				
ResNet-50	COVID	0.9919	1	0.9959	0.9959	0.0040	0.8996	0.79201
	Non-COVID	1	0.9918	0.9958				
Mixed Dataset								
MobileNetv2	COVID	0.9695	0.9937	0.9814	0.9812	0.00099	0.8003	0.33411
	Non-COVID	0.99359	0.9687	0.9810				
ResNet-50	COVID	0.9844	0.9875	0.9859	0.9859	0.0065	0.8914	0.0035
	Non-COVID	0.9874	0.9843	0.9858				

Table 4 Findings HPs based on the BO for SVM and kNN algorithms

ML algorithms	Datasets	Models of features	Findings methods	Parameters		
				C	Degree	Kernel
SVM	Dataset1	MobileNetv2	GridSearchCV	0.1	1	rbf
			Bayesian Optimization	0.2444	5	rbf
		ResNet-50	GridSearchCV	10	1	rbf
			Bayesian Optimization	0.01	5	Llinear
	Dataset2	MobileNetv2	GridSearchCV	0.1	1	Poly
			Bayesian Optimization	0.05	1	Poly
		ResNet-50	GridSearchCV	10	1	Poly
			Bayesian Optimization	38.12	5	rbf
	Mixed Dataset	MobileNetv2	GridSearchCV	10	1	rbf
			Bayesian Optimization	100	5	rbf
		ResNet-50	GridSearchCV	10	1	Poly
			Bayesian Optimization	0.02	5	Linear
				n_neighbors	Metric	Weights
kNN	Dataset1	MobileNetv2	GridSearchCV	11	Euclidean	Uniform
			Bayesian optimization	10	Minkowski	Distance
		ResNet-50	GridSearchCV	3	Manhattan	Uniform
			Bayesian optimization	4	Manhattan	Uniform
	Dataset2	MobileNetv2	GridSearchCV	11	Euclidean	Uniform
			Bayesian optimization	11	Minkowski	Distance
		ResNet-50	GridSearchCV	5	Euclidean	uniform
			Bayesian optimization	5	Euclidean	Distance
	Mixed Dataset	MobileNetv2	GridSearchCV	5	Euclidean	Uniform
			Bayesian optimization	6	Euclidean	Distance
		ResNet-50	GridSearchCV	3	Manhattan	Uniform
			Bayesian optimization	4	Manhattan	Distance

validation was used to ensure the reliability of these methods. In the method proposed in the study, SVM and kNN ML algorithms were preferred. The parameter values used in these algorithms are C, kernel, and degree values for SVM. For kNN, n_neighbors (k) are metric and weights.

The recommended approach is shown in Fig. 3.

4 Results

4.1 Dataset

4.1.1 Dataset 1

The dataset used in the study was taken from the [59] study. The dataset includes two classes, COVID and Non-COVID. The COVID-CT-dataset has 349 CT images containing clinical signs of COVID-19 in 216 patients. The non-COVID dataset includes 396 CT images. According to the study, the images in the dataset were confirmed by a senior radiologist at Tongji Hospital, Wuhan, China, who diagnosed and treated a large number of COVID-19 patients during the outbreak of this disease between January and April. They also state that the dataset in this study was collected from articles on COVID-19 taken from medRxiv, bioRxiv, NEJM, JAMA, Lancet, etc. In our study, a total of 698 images, 349 from each class, were used for the classes to contain an equal number of images. These images were used in DNN as 20% test data.

4.1.2 Dataset 2

The second dataset we used is from the Kaggle. This dataset includes 1252 CT images of SARS-CoV-2 infection (COVID-19) and 1230 CT images without COVID-19 (Non-COVID). The dataset was collected from real patients in hospitals in Sao Paulo, Brazil [60, 61].

Table 5 Results obtained by the SVM and the kNN algorithms for Dataset1

	Classes	Precision	Recall	f1-score	Accuracy
MobileNetv2					
SVM					
Normal	COVID	0.9589	1	0.9790	0.9785
	Non-COVID	1	0.9571	0.9780	
GridSearchCV	COVID	0.9589	1	0.9790	0.9785
	Non-COVID	1	0.9571	0.9780	
Bayesian	COVID	0.9589	1	0.9790	0.9785
	Non-COVID	1	0.9571	0.9780	
kNN					
Normal	COVID	0.9589	1	0.9790	0.9785
	Non-COVID	1	0.9571	0.9781	
GridSearchCV	COVID	0.9583	0.9857	0.9718	0.9714
	Non-COVID	0.9852	0.9571	0.9710	
Bayesian	COVID	0.9583	0.9857	0.9718	0.9714
	Non-COVID	0.9852	0.9571	0.9710	
ResNet-50					
SVM					
Normal	COVID	0.9718	0.9857	0.9787	0.9785
	Non-COVID	0.9855	0.9714	0.9784	
GridSearchCV	COVID	0.9583	0.9857	0.9718	0.9714
	Non-COVID	0.9852	0.9571	0.9710	
Bayesian	COVID	0.9583	0.9857	0.9718	0.9714
	Non-COVID	0.9852	0.9571	0.9710	
kNN					
Normal	COVID	0.9583	0.9857	0.9718	0.9714
	Non-COVID	0.9852	0.9571	0.9710	
GridSearchCV	COVID	0.9718	0.9857	0.9787	0.9785
	Non-COVID	0.9855	0.9714	0.9784	
Bayesian	COVID	0.9452	0.9857	0.9650	0.9642
	Non-COVID	0.9850	0.9428	0.9635	

An example of the images used for datasets is shown in Fig. 4. In addition, the sample numbers for these datasets used in the training and testing phase are given in Table 2.

4.2 Evaluation metrics

The application we developed for our study was written using MATLAB and Python programs. The computer where the applications ran had 16 GB of RAM and an I7 processor. In addition, models were run on GeForce 1070 graphics card with GPU. The performance metrics [62] of our study were obtained using a confusion matrix. The confusion matrix is the matrix N X N where N is the predicted number of classes. Since there are 2 classes in our study, a 2×2 matrix is obtained. These metrics are; accuracy: the ratio of the total number of predictions that are correct, positive predictive value or precision: proportion of correctly identified positive cases, negative predictive value: proportion of correctly identified negative events and sensitivity or recall: proportion of true positive cases correctly identified. For a 2-class structure, these values are shown in Fig. 5 for the confusion matrix. Metrics are calculated according to Eqs. 8–12.

$$S_e = \frac{TP}{TP + FN} \tag{8}$$

$$S_p = \frac{TN}{TN + FP} \tag{9}$$

$$Pre = \frac{TP}{TP + FP} \tag{10}$$

$$F - score = \frac{2TP}{2TP + FP + FN} \tag{11}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

	Classes	Precision	Recall	f1-score	Accuracy
MobileNetv2					
SVM					
Normal	COVID	0.9959	1	0.9979	0.9979
	Non-COVID	1	0.9959	0.9979	
GridSearchCV	COVID	0.9959	1	0.9979	0.9979
	Non-COVID	1	0.9959	0.9979	
Bayesian	COVID	0.9959	1	0.9979	0.9979
	Non-COVID	1	0.9959	0.9979	
kNN					
Normal	COVID	0.9959	1	0.9979	0.9979
	Non-COVID	1	0.9959	0.9979	
GridSearchCV	COVID	0.9959	1	0.9979	0.9979
	Non-COVID	1	0.9959	0.9979	
Bayesian	COVID	0.9959	1	0.9979	0.9979
	Non-COVID	1	0.9959	0.9979	
ResNet-50					
SVM					
Normal	COVID	0.9959	0.9959	0.9959	0.9959
	Non-COVID	0.9959	0.9959	0.9959	
GridSearchCV	COVID	0.9959	1	0.9979	0.9979
	Non-COVID	1	0.9959	0.9979	
Bayesian	COVID	0.9959	1	0.9979	0.9979
	Non-COVID	1	0.9959	0.9979	
kNN					
Normal	COVID	0.9959	1	0.9979	0.9979
	Non-COVID	1	0.9959	0.9979	
GridSearchCV	COVID	0.9959	1	0.9979	0.9979
	Non-COVID	1	0.9959	0.9979	
Bayesian	COVID	0.9959	1	0.9979	0.9979
	Non-COVID	1	0.9959	0.9979	

Table 6 Results obtained by the SVM and the kNN algorithms for Dataset2

4.3 Experiments

The experimental studies performed consist of three steps.

4.3.1 Experiment 1

In the first experiment, attempts were made to find HPs of DNN models by using BO.

In these experimental studies, the optimum values were found for the Learning Rate, Momentum and L2 regularization parameters for the SGD optimization algorithm using BO in MobileNet and ResNet models. The results obtained from these optimum values and models are presented in Table 3. Bayesopt tool in MATLAB program is used for BO. This tool uses cross-validation loss as an objective function for BO.

When Table 3 is examined, an accuracy rate of 97.86% is achieved in the MobileNetv2 model for Dataset 1. For

Dataset 2, both models provided an accuracy rate of over 99%. Again, precision, recall and f1-score values for this dataset are over 99%. For the Mixed dataset created by mixing both datasets, the ResNet-50 model achieved 98.50% success. These ratios are derived from confusion matrices. The confusion matrix obtained from the models for the Mixed Dataset is given in Fig. 6.

4.3.2 Experiment 2

In the second experiment, the features obtained from Dataset 1 and Dataset 2 were classified using SVM and kNN ML algorithm models. Here again, BO was used to find HP of these models. In addition, GS was also used to verify the reliability of the hyperparameter optimization. The HP found is given in Table 4 separately for each model and each ML algorithm. Results obtained by HP optimization are given for all datasets in order. Table 5 shows

Table 7 Results obtained by the SVM and the kNN algorithms for Mixed Dataset

	Classes	Precision	Recall	f1-score	Accuracy
MobileNetv2					
SVM					
Normal	COVID	0.9674	0.9939	0.9805	0.97972
	Non-COVID	0.9934	0.9647	0.9788	
GridSearchCV	COVID	0.9703	0.9939	0.9819	0.98128
	Non-COVID	0.9934	0.9679	0.9805	
Bayesian	COVID	0.9703	0.9939	0.9819	0.98128
	Non-COVID	0.9934	0.9679	0.9805	
kNN					
Normal	COVID	0.9484	0.9513	0.9499	0.9485
	Non-COVID	0.9485	0.9455	0.9470	
GridSearchCV	COVID	0.9484	0.9513	0.9499	0.9485
	Non-COVID	0.9485	0.9455	0.9470	
Bayesian	COVID	0.9494	0.9696	0.9593	0.9579
	Non-COVID	0.9672	0.9455	0.9562	
ResNet-50					
SVM					
Normal	COVID	0.9939	0.9908	0.9923	0.9922
	Non-COVID	0.9904	0.9935	0.9920	
GridSearchCV	COVID	0.9938	0.9878	0.9908	0.99064
	Non-COVID	0.9872	0.9935	0.9904	
Bayesian	COVID	0.9938	0.9878	0.9908	0.99064
	Non-COVID	0.9872	0.9935	0.9904	
kNN					
Normal	COVID	0.9909	0.9939	0.9924	0.9922
	Non-COVID	0.9935	0.9903	0.9919	
GridSearchCV	COVID	0.9909	0.9969	0.9939	0.99376
	Non-COVID	0.9967	0.9903	0.9935	
Bayesian	COVID	0.9909	0.9969	0.9939	0.99376
	Non-COVID	0.9967	0.9903	0.9935	

(a) ResNet Features with kNN   **(b)** ResNet Features with kNN(Bayesian)
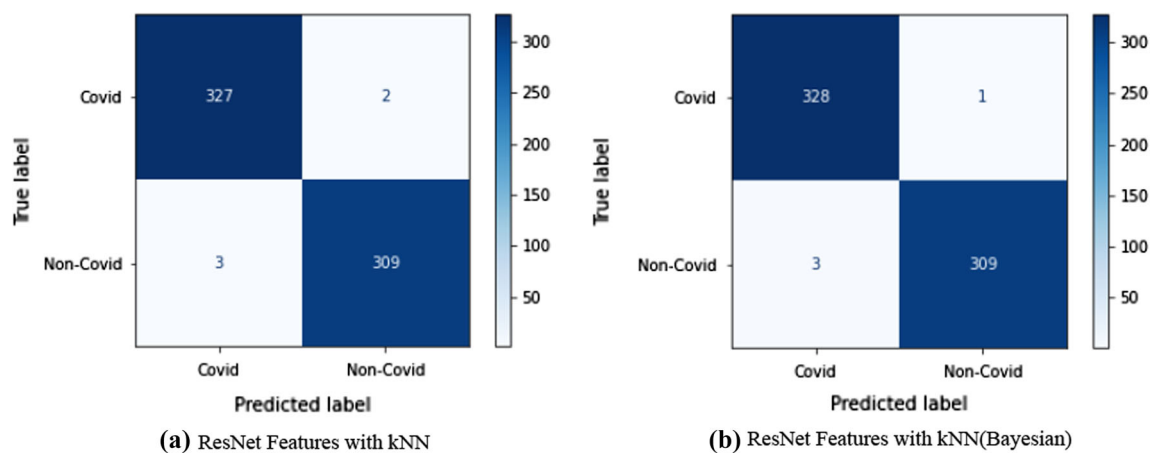
Fig. 7 Confusion Matrix for ResNet features with the kNN algorithm
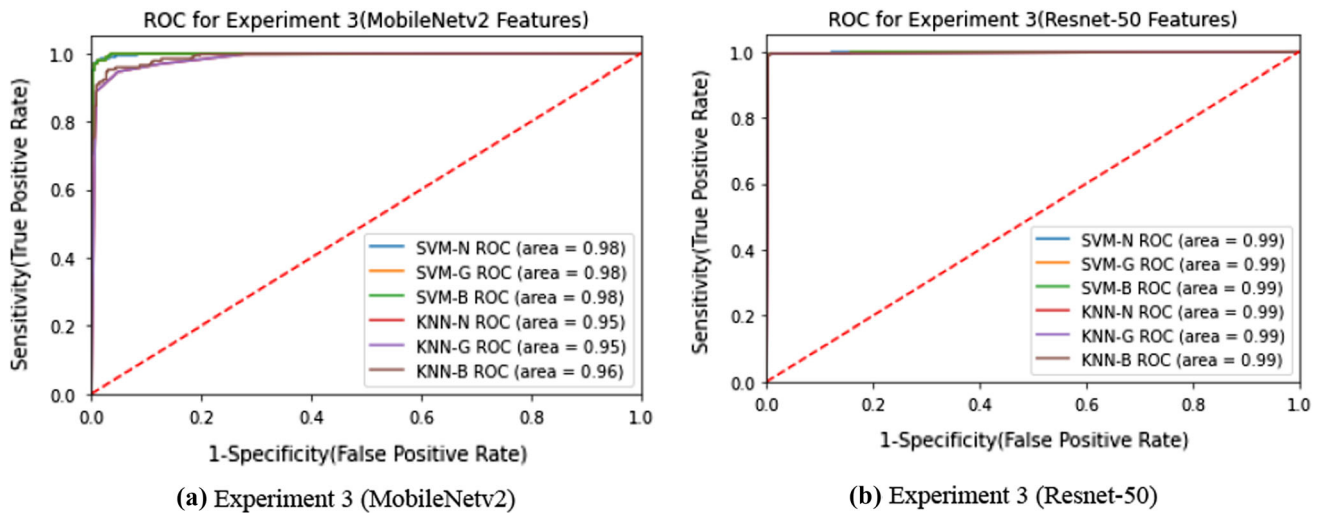
Fig. 8 Graphical of the ROC-curve obtained at the experiment3 by the MobileNetv2 and the Resnet-50

Table 8 Accuracy values obtained using the SVM and kNN algorithms by the MobileNetv2 and the ResNet-50

Overall Accuracy	Models	Network	SVM			kNN		
			Normal	GridSearchCV	Bayesian Optimization	Normal	GridSearchCV	Bayesian Optimization
Dataset1	MobileNetv2	0.9786	0.97857	0.97857	0.97857	0.97857	0.97143	0.97143
	ResNet-50	0.9714	0.97857	0.97143	0.97143	0.97143	0.97857	0.9629
Dataset2	MobileNetv2	0.998	0.99797	0.99797	0.99797	0.99797	0.99797	0.99797
	ResNet-50	0.9959	0.99593	0.99797	0.99797	0.99797	0.99797	0.99797
Mixed Dataset	MobileNetv2	0.9812	0.97972	0.98128	0.98128	0.94852	0.94852	0.95788
	ResNet-50	0.9859	0.9922	0.99064	0.99064	0.9922	0.99376	0.99376

the results for SVM and kNN for Dataset 1. Table 6 shows the results for SVM and kNN for Dataset 2.

For Dataset 1, when we look at the results obtained by using the features in the models with ML algorithms, an accuracy rate of 97.85% was obtained by using Mobile-Netv2 features and SVM. The kNN performance result for the features extracted from this model is 97.14%. Again for this dataset, the performance rate for the features obtained from the ResNet-50 model is 97.85%, while the performance rate for Bayesian kNN is 96.42%. For the features of this model, the highest performance was achieved with GS with 97.85%.

In the experimental studies for Dataset 2, the success is over 99%. This performance value is realized as a result of model training for both models.

4.3.3 Experiment 3

These experimental results were carried out on the Mixed Dataset obtained by mixing Dataset 1 and Dataset 2. The results obtained using this dataset are shown in Table 7.

When we examine the table for the Mixed Dataset, the features obtained from MobilNetv2 and SVM provided a performance of 98.12%, while kNN achieved a performance of 95.78%.

For ResNet-50, the performance achieved with SVM is 99.064%. The performance obtained from kNN is the highest success rate for the mixed dataset at 99.376%. The complexity matrix for this model, which provides the highest accuracy rate, is given in Fig. 7. In addition, the ROC curves obtained from this experimental result are given in Fig. 8.

In Table 8, all steps in our study are summarized in terms of accuracy values.

According to this table, while MobilNetv2 has the highest accuracy rate obtained as a result of DNN training, the highest performance dataset is Dataset 2. The SVM model obtained by using GridSearchCV and BO for Dataset 1 obtained higher accuracy than kNN. For Dataset 2, the performance is over 99% in all models. As a result of the training of DNN models, for the Mixed Dataset a performance value of 98.59% was obtained with the ResNet-50 model. It is possible to see that for the

Table 9 COVID-19 classification results in the literature using different methods

Class	Subjects	Method	Prec (%)	Sens. (%) or recall	f1-score	Acc (%)	Ref
COVID-19/ normal	73 cases	CNN	N/A	∼ 90	∼ 90	∼ 90	Sing et al. [3]
COVID-19/ normal	178 pneumonia 247 normal	DL IRRCNN	N/A	N/A	98.85	98.78	Alom et al. [8]
COVID-19/ normal	1601 pneumonia 1693 normal	Efficient DL	N/A	N/A	86.19	87.68	Silva et al. [9]
COVID-19/ normal	449 pneumonia 425 normal	DL Multi-tasking Learning	N/A	96.00	N/A	94.67	Amyar et al. [10]
COVID-19/ normal	630 pneumonia (499 CT training, 131 CT testing)	DeCoVNet	N/A	90.7	N/A	N/A	Wang et al. [11]
COVID-19/ normal	1,262 COVID-19 1,230 normal	DenseNet201	96.29	96.29	96.29	96.25	Jaiswal et al. [12]
COVID-19/ normal	Dataset1:698 349 COVID,349 Non-COVID Dataset2:2260 1230 COVID,1230 Non-COVID	MobileNetv2, ResNet50 and Deep Features	98.18	98.09	98.12	98.12	Proposed Approach (MobilNetv2 + SVM)
			95.83	95.75	95.77	95.79	Proposed Approach (MobilNetv2 + kNN)
			99.05	99.06	99.06	99.06	Proposed Approach (ResNet-50 + SVM)
			99.38	99.36	99.37	99.37	Proposed Approach (ResNet-50 + kNN)

Class, classification; *Sens*, sensitivity/recall; *Spec*, specificity; *Prec*, precision; *Acc*, accuracy; *Ref*, reference

properties obtained from MobilNetv2, the performance rates obtained with SVM are higher than the performance rates obtained with kNN. For this dataset, a high performance rate of 99.37% was obtained for both GridSearchCV and Bayesian kNN in the classification of the features obtained from the ResNet-50 model.

5 Discussion

When we look at the studies in this field, it is possible to see that many studies have been carried out recently. These studies were conducted on different datasets with different methods. It is possible to see that all the studies carried out today, where the disease is defined as a pandemic, make a certain contribution to the field. Our first aim was to

contribute to these studies in this study. We hope that this approach, which we have obtained with the use of BO with DNN and ML algorithms, will be among the studies that contribute to the field. The results of the studies conducted on the CT images for COVID-19 and the comparison table of our approach are given in Table 9.

Table 10 shows the performance of previous publications and our study with these datasets. Our study is about COVID-19 diagnosis on CT images. Studies on this subject continue in the current period. The advantage of our study is that we use a model that does not require high computation, such as MobilNetv2. In this way, we think very fast results can be obtained. As is known, individuals infected with this disease must be isolated from others very quickly. We aimed to achieve this with our approach. When other studies completed in this field are examined, given in

Table 10 Performance comparison of the COVID-19/normal classification in this study according to the literature

Class	Subjects	Method	Prec (%)	Sens. (%) or recall	f1-score	Acc (%)	Ref
COVID-19/ normal	Dataset 2	DenseNet-201	96.29	96.29	96.29	96.25	Jaiswal et al. [12]
COVID-19/ normal	Dataset 2	VGG19	94.86	94.04	--	95.0	Panwar et al. [24]
COVID-19/ normal	Dataset 1	CNN	93.0	95.0	--	78.5	Wu et al. [26]
COVID-19/ normal	Dataset 1	DenseNet-169	--	--	0.85	0.86	He et al. [25]
COVID-19/ normal	Dataset1:698 349 COVID,349 Non-COVID Dataset2:2260 1230 COVID,1230 Non-COVID	MobileNetv2, ResNet50 and Deep Features	98.18	98.09	98.12	98.12	Proposed Approach (MobilNetv2 + SVM)
			95.83	95.75	95.77	95.79	Proposed Approach (MobilNetv2 + kNN)
			99.05	99.06	99.06	99.06	Proposed Approach (ResNet-50 + SVM)
			99.38	99.36	99.37	99.37	Proposed Approach (ResNet-50 + kNN)

Table 10, high success was obtained in this study. As mentioned in the experimental results section, we attempted to show that the performances of the datasets can give different results by taking separate results for more than one dataset. Another advantage of our study is that BO was tested on two models and the results were shown, and the results were given with Bayes-based ML algorithms using deep features. In other words, these methods, which are presented separately in other studies, are presented together in our study.

6 Conclusions

In this study, a BO-based approach that diagnoses COVID-19 on CT images is proposed. MobilNetv2 and ResNet-50 models, which are DNN models, were used in the first stage of the study to find optimum HPs. In the second stage of the study, feature extraction was achieved using these models. In our experimental results, two datasets from different countries were used. A mixed dataset was created by mixing these datasets and the performances of the

models were shown for this dataset. Among ML algorithms, SVM and kNN algorithms were preferred in our study because they are the most widely used in this field in literature reviews. Again, BO was used to select the optimum of some parameter values for these algorithms. In addition, by using GS, a methods used in HP detection, the results are given comparatively with BO. A 99.37% success rate for the Mixed Dataset was achieved with BO parameters and the kNN algorithm had high performance. The study is expected to act as a decision support mechanism that helps experts with diagnosis of this disease. In future studies, studies carried out with different models and methods will contribute to this field.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This research work does not involve chemicals, procedures, or equipment that have any unusual hazards inherent in their use.

References

1. I Ozsahin B Sekeroglu MS Musa MT Mustapha D Uzun Ozsahin 2020 Review on diagnosis of COVID-19 from Chest CT images using artificial ıntelligence Comput Math Methods Med https://doi.org/10.1155/2020/9756518

2. T Ai Z Yang H Hou 2020 Correlation of chest CT and RT-PCR testing for coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases Radiology 296 E32 E40 https://doi.org/10.1148/radiol.2020200642

3. D Singh V Kumar KM Vaishali 2020 Classification of COVID-19 patients from chest CT images using multi-objective differential evolution–based convolutional neural networks Eur J Clin Microbiol Infect Dis 39 1379 1389 https://doi.org/10.1007/s10096-020-03901-z

4. HX Bai B Hsieh Z Xiong 2020 Performance of radiologists in differentiating COVID-19 from non-COVID-19 viral Pneumonia at chest CT Radiology 296 E46 E54 https://doi.org/10.1148/radiol.2020200823

5. SS Hare AN Tavare V Dattani 2020 Validation of the British society of thoracic imaging guidelines for COVID-19 chest radiograph reporting Clin Radiol 75 9 14 https://doi.org/10.1016/j.crad.2020.06.005

6. Y Tingting W Junqian W Lintai X Yong 2019 Three-stage network for age estimation CAAI Trans Intell Technol 4 122 126 https://doi.org/10.1049/trit.2019.0017

7. Y Fang H Zhang J Xie 2020 Sensitivity of chest CT for COVID-19: comparison to RT-PCR Radiology 296 E115 E117

8. Alom MZ, Rahman MMS, Nasrin MS, et al (2020) COV-ID_MTNet: COVID-19 detection with multi-task deep learning approaches. https://arxiv.org/abs/2004.03747

9. P Silva E Luz G Silva 2020 COVID-19 detection in CT images with deep learning: a voting-based scheme and cross-datasets analysis Inf Med Unlock https://doi.org/10.1016/j.imu.2020.100427

10. A Amyar R Modzelewski H Li S Ruan 2020 Multi-task deep learning based CT imaging analysis for COVID-19 pneumonia: Classification and segmentation medRxiv https://doi.org/10.1101/2020.04.16.20064709

11. X Wang X Deng Q Fu 2020 A weakly-supervised framework for COVID-19 classification and lesion localization from chest CT IEEE Trans Med Imag 39 2615 2625 https://doi.org/10.1109/TMI.2020.2995965

12. A Jaiswal N Gianchandani D Singh 2021 Classification of the COVID-19 infected patients using DenseNet201 based deep transfer learning J Biomol Struct Dyn 39 5682 5689 https://doi.org/10.1080/07391102.2020.1788642

13. Mayhew MB, Tran E, Choi K, et al (2020) Optimization of genomic classifiers for clinical deployment: evaluation of Bayesian optimization to select predictive models of acute ınfection and ın-hospital mortality

14. S Cabras 2021 A bayesian-deep learning model for estimating covid-19 evolution in spain Mathematics 9 22 2921

15. Ghoshal B, Tucker A (2020) Estimating Uncertainty and Interpretability in Deep Learning for Coronavirus (COVID-19) Detection. arXiv: https://arxiv.org/abs/2003.10769

16. S Dhamodharavadhani R Rathipriya JM Chatterjee 2020 COVID-19 mortality rate prediction for india using statistical neural network models Front Public Heal 8 441 https://doi.org/10.3389/fpubh.2020.00441

17. F Ucar D Korkmaz 2020 COVIDiagnosis-Net: Deep Bayes-SqueezeNet based diagnosis of the coronavirus disease 2019 (COVID-19) from X-ray images Med Hypoth 140 109761 https://doi.org/10.1016/j.mehy.2020.109761

18. SE Arman S Rahman SA Deowan 2022 COVIDXception-Net: a Bayesian optimization-based deep learning approach to diagnose COVID-19 from X-Ray images SN Comput Sci 3 1 22

19. M Nour Z Cömert K Polat 2020 A novel medical diagnosis model for COVID-19 infection detection based on deep features and Bayesian optimization Appl Soft Comput 97 106580 https://doi.org/10.1016/j.asoc.2020.106580

20. ST Radev F Graw S Chen 2021 OutbreakFlow: model-based Bayesian inference of disease outbreak dynamics with invertible neural networks and its application to the COVID-19 pandemics in Germany PLOS Comput Biol 17 1 26 https://doi.org/10.1371/journal.pcbi.1009472

21. R Pal AA Sekh S Kar DK Prasad 2020 Neural network based country wise risk prediction of COVID-19 Appl Sci https://doi.org/10.3390/app10186448

22. Mallick A, Dwivedi C, Kailkhura B, et al (2020) Probabilistic neighbourhood component analysis: sample efficient uncertainty estimation in deep learning

23. K Gao J Su Z Jiang 2021 Dual-branch combination network (DCN): towards accurate diagnosis and lesion segmentation of COVID-19 using CT images Med Image Anal https://doi.org/10.1016/j.media.2020.101836

24. H Panwar PK Gupta MK Siddiqui 2020 A deep learning and grad-CAM based color visualization approach for fast detection of COVID-19 cases using chest X-ray and CT-Scan images Chaos Solitons Fractals 140 110190 https://doi.org/10.1016/j.chaos.2020.110190

25. X He X Yang S Zhang 2020 Sample-efficient deep learning for COVID-19 diagnosis based on CT scans medRxiv https://doi.org/10.1101/2020.04.13.20063941

26. Y-H Wu S-H Gao J Mei 2021 JCS: an explainable COVID-19 diagnosis system by joint classification and segmentation IEEE Trans Image Process 30 3113 3126 https://doi.org/10.1109/TIP.2021.3058783

27. J Wu X-Y Chen H Zhang 2019 Hyperparameter optimization for machine learning models based on Bayesian optimization J Electron Sci Technol 17 26 40

28. P Doke D Shrivastava C Pan 2020 Using CNN with Bayesian optimization to identify cerebral micro-bleeds Mach Vis Appl 31 36 https://doi.org/10.1007/s00138-020-01087-0

29. MI Sameen B Pradhan S Lee 2020 Application of convolutional neural networks featuring Bayesian optimization for landslide susceptibility assessment CATENA 186 104249 https://doi.org/10.1016/j.catena.2019.104249

30. Frazier PI (2018) A Tutorial on Bayesian Optimization. arXiv: https://arxiv.org/abs/1807.02811

31. Ma X, Triki AR, Bermana M, et al (2019) A Bayesian optimization framework for neural network compression. In: IEEE/CVF International conference on computer vision (ICCV). Seoul, Korea (South), pp 10274–10283

32. J Wilson F Hutter M Deisenroth 2018 Maximizing acquisition functions for Bayesian optimization S Bengio H Wallach H Larochelle Eds Advances in neural information processing systems (NeurIPS 2018) Curran Associates Inc Montreal 905 9916

33. M Feurer F Hutter 2018 Hyperparameter optimization F Hutter L Kotthoff J Vanschoren Eds Automated machine learning (methods, systems, challenges) Springer International Publishing Cham 3 33

34. Garnett R (2015) Lecture 12: Bayesian optimization. In CSE 515T: Bayesian methods in machine learning

35. Congdon P (2001) The Bayesian method its benefits and ımplementation. In: Bayesian statistical modelling. Wiley, Chichester

36. J Gill 2002 Bayesian methods Bayesian methods: a social and behavioral sciences approach Chapman and Hall/CRC Boca Raton, Florida 1 30

37. Abdani SR, Zulkifley MA, Hani Zulkifley N (2020) A lightweight deep learning model for covıd-19 detection. In: 2020 IEEE Symposium on ındustrial electronics and applications (ISIEA). IEEE

38. Brochu E, Cora VM, de Freitas N (2010) A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. arXiv: https://arxiv.org/abs/1012.2599v1

39. B Shahriari K Swersky Z Wang 2016 Taking the human out of the loop: a review of Bayesian optimization Proc Inst Radio Eng 104 148 175 https://doi.org/10.1109/JPROC.2015.2494218

40. I Goodfellow Y Bengio A Courville 2016 Deep learning MIT Press Cambridge

41. He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for ımage recognition. arXiv: https://arxiv.org/abs/1512.03385

42. Sandler M, Howard A, Zhu M, et al (2018) MobileNetV2: Inverted residuals and linear bottlenecks. In: 2018 IEEE/CVF conference on computer vision and pattern recognition, pp 4510–4520

43. Y Yao Y Liu Y Yu 2013 K-SVM: an effective SVM algorithm based on K-means clustering J Comput 8 2632 2639 https://doi.org/10.4304/jcp.8.10.2632-2639

44. Rahamathunnisa U, Nallakaruppan MK, Anith A, KS SK (2020) Vegetable disease detection using k-means clustering and svm. In: 2020 6th International conference on advanced computing and communication systems (ICACCS)

45. Z Rustam SAA Kharis 2020 Comparison of support vector machine recursive feature elimination and kernel function as feature selection using support vector machine for lung cancer classification J Phys Conf Ser https://doi.org/10.1088/1742-6596/1442/1/012027

46. M-L Huang Y-H Hung WM Lee 2014 SVM-RFE based feature selection and taguchi parameters optimization for multiclass SVM classifier Sci World J 2014 795624 https://doi.org/10.1155/2014/795624

47. I Guyon J Weston S Barnhill V Vapnik 2002 Gene selection for cancer classification using support vector machines Mach Learn 46 389 422 https://doi.org/10.1023/A:1012487302797

48. Barstugan M, Ozkaya U, Ozturk S (2020) Coronavirus (COVID-19) Classification using CT Images by Machine Learning Methods. arXiv: https://arxiv.org/abs/2003.09424

49. I Guyon S Gunn M Nikravesh LA Zadeh 2008 Feature extraction: foundations and applications 207 Springer Cham

50. X Huang L Zhang B Wang 2017 Feature clustering based support vector machine recursive feature elimination for gene selection Appl Intell 48 594 607

51. V Vapnik SE Golowich A Smola 1997 Support vector method for function approximation, regression estimation, and signal processing Adv Neural Inf Process Syst 9 281 287

52. C Cortes V Vapnik 1995 Support-vector networks Mach Learn 20 273 297 https://doi.org/10.1007/BF00994018

53. M Awad R Khanna 2015 Machine learning and knowledge discovery Efficient learning machines Apress Berkeley 19 38

54. Hand D, Mannila H, Smyth P (2001) Principles of data mining. In: Encyclopaedia of environmentrics. Wiley, Cambridge

55. Z Zhang 2016 Introduction to machine learning: k-nearest neighbors Ann Transl Med 4 218

56. L Peterson 2009 K-nearest neighbour Scholarpedia 4 1883 https://doi.org/10.4249/scholarpedia.1883

57. G Guo H Wang D Bell 2003 KNN model-based approach in classification R Meersman Z Tari DC Schmidt Eds On the move to meaningful internet systems 2003: CoopIS, DOA, and ODBASE Springer Berlin, Heidelberg 986 996

58. SB Imandoust M Bolandraftar 2013 Application of K-nearest neighbor (kNN) approach for predicting economic events : theoretical background Int J Eng Res Appl 3 605 610

59. Zhao J, Zhang Y, He X, Xie P (2020) COVID-CT-Dataset: a CT scan dataset about COVID-19. ArXiv arXiv: https://arxiv.org/abs/2003.13865

60. Soares E, Angelov P, Biaso S, Froes MH and Abe DK (2020) Explainable by design approach for Covid-19 classification via CT scan. medRxiv https://doi.org/10.1101/2020.04.24.20078584

61. P Angelov E Almeida Soares 2020 SARS-CoV-2 CT-scan dataset: a large dataset of real patients CT scans for SARS-CoV-2 identification MedRxiv https://doi.org/10.1101/2020.04.24.20078584

62. M Canayaz 2021 MH-COVIDNet: diagnosis of COVID-19 using deep neural networks and meta-heuristic-based feature selection on X-ray images Biomed Signal Process Control https://doi.org/10.1016/j.bspc.2020.102257

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.