Neural Computing and Applications (2024) 36:18277–18295
https://doi.org/10.1007/s00521-024-10150-0

ORIGINAL ARTICLE

ETSVF-COVID19: efficient two-stage voting framework for COVID-19 detection

Kemal Akyol[1]

Received: 2 August 2023 / Accepted: 27 June 2024 / Published online: 24 July 2024

Abstract

COVID-19 disease, an outbreak in the spring of 2020, reached very alarming dimensions for humankind due to many infected patients during the pandemic and the heavy workload of healthcare workers. Even though we have been saved from the darkness of COVID-19 after about three years, the importance of computer-aided automated systems that support field experts in the fight against with global threat has emerged once again. This study proposes a two-stage voting framework called ETSVF-COVID19 that includes transformer-based deep features and a machine learning approach for detecting COVID-19 disease. ETSVF-COVID19, which offers 99.2% and 98.56% accuracies on computed tomography scan and X-radiation images, respectively, could compete with the related works in the literature. The findings demonstrate that this framework could assist field experts in making informed decisions while diagnosing COVID-19 with its fast and accurate classification role. Moreover, ETSVF-COVID19 could screen for chest infections and help physicians, particularly in areas where test kits and specialist doctors are inadequate.

Keywords COVID-19 · Transformer architectures · Deep features · Two-stage voting framework

1 Introduction

COVID-19 is a contagious disease with a high death rate [1]. The World Health Organization first informed on Dec 31, 2019, that it became aware of this virus with a series of ‘viral pneumonia’ cases in Wuhan, People’s Republic of China [2]. Scientists and policymakers have been concerned about the quick contagion of this disease [3]. Because of countless infected patients during the pandemic and the heavy workload of the healthcare worker, artificial intelligence-based computer-aided systems can assist field experts by speeding up the diagnosis process. In cases with few radiologists, these systems contribute to more accurate disease diagnosis in less time [1]. Medical imaging has a role in the diagnosis of this disease. A chest X-radiation (X-ray) is a type of radiography imaging that is commonly used for COVID-19 detection. X-radiation was discovered by German scientist W. C. Röntgen in 1895 [4]. Computed tomography (CT) scan is a chest radiography image that utilizes X-ray to create three-dimensional images of the lungs [5]. Radiological examinations, particularly thin-section lung CT, are crucial to deal with this infectious disease [6]. A chest CT scan image provides a detailed view of the patient’s blood vessels, organs, soft tissues, and bones, as well as more details about the patient’s health [7]. Clinicians could make quick decisions using the COVID-19 detection system [8]. In another respect, manually examining chest CT scan images for this disease is tedious, takes too much time, and tends to make human mistakes [9]. Since the COVID-19 outbreak, convolutional neural networks (CNN) that have become attractive with performances in disease diagnosing have been used [1]. In this context, many deep learning approaches have been proposed since the emergence of the disease. These strategies show promise for automating COVID-19 diagnosis and reducing the strain on medical systems [10]. Several researchers analyzed both chest X-ray and CT scan images using computer vision approaches that presented good results. For example, Acar et al. [1] presented a study that includes normalization, data augmentation, pre-trained CNN models, and classification. Babukarthik et al. [11]

✉ Kemal Akyol
kakyol@kastamonu.edu.tr

[1] Faculty of Engineering and Architecture, Computer Engineering, Kastamonu University, Kastamonu, Turkey

proposed a genetic deep learning architecture to identify this disease. Balaha et al. [12] introduced hybrid learning with optimized hyperparameters. Gupta and Bajaj used two different pre-trained models and a deep learning model they designed from scratch and reported that DarkNet19 achieved the highest classification accuracy of 98.91% [9]. Hosseinzadeh presented the multi-view feature learning framework. They reported that their model, which combines the features extracted through AlexNet, GoogleNet, ResNet50, SqueezeNet, and VGG19 pre-trained deep CNN models, outperformed pre-trained models [13]. Jalali et al. used the K-nearest neighbors classifier, which takes the neighborhood labeling agreement into account in the last layer of the CNN they proposed. In addition, they automatically tuned CNN hyperparameters with an evolutionary algorithm they developed and improved the classification accuracy [14]. Khan et al. developed a CNN architecture that extracts deep features from the chest CT scan images. Then, they used the correntropy technique to select the most distinguishing deep features and classified them using the one-class kernel extreme learning machines [15]. Khurana and Soni compared the performances of several CNN models and achieved promising results with ResNet50 [16]. Liu et al. [17] performed multidirectional data augmentation and extracted features. Then, they classified the fused features. Loey et al. first used conditional generative adversarial nets and classical data augmentation techniques. Then, they conducted state-of-the-art deep CNN and achieved the best test accuracy with ResNet50 [18]. Moghaddam and Gholamalinejad proposed a new deep learning architecture that includes the Haar wavelet on the pooling layer to perform simultaneous dimension reduction and feature extraction [19]. Nayak et al. examined the performances of some pre-trained deep learning models for COVID-19 classification efficiency. Then, they conducted comparative analyses by tuning the hyperparameters of the models to determine the best model in their study [20]. Nigam et al. [21] conducted experiments with state-of-the-art CNN models and achieved 93.48% accuracy with the EfficientNet-B7 model. Rahimzadeh et al. [22] applied the dropout technique to the CNN architectures and used the softmax activation function in the output layer. Roy and Kumar combined the predictions of the pre-trained CNN models and classified them with ensemble modeling [23]. Sahin introduced a scratch CNN model with 30 batch size, 1e-4 learning rate, and Adam optimization algorithm. The author reported that their model outperformed the pre-trained MobileNetv2 and ResNet50 models with an accuracy of 96.71% on the testing set [24]. Lastly, Zhang et al. [25] applied augmentation techniques on the original training dataset and achieved high classification accuracy with their proposed deep comparative mutual learning. In addition, three-class experiments were also conducted for COVID-19. For example, Shi et al. [10] first addressed the issue of insufficient data and then designed a four-layer network architecture based on the 'model capacity model complexity' balance concept to detect COVID-19 from chest X-ray images automatically. Kavya et al. employed the VGG16 and ResNet50 deep learning models to identify COVID-19 cases from chest X-ray images. They employed data augmentation and random sampling approaches to deal with unbalanced data [26]. Budak et al. [27] used transfer learning-based deep learning models to diagnose COVID-19 from chest X-ray images and achieved high diagnostic accuracies using DenseNet121 and DenseNet201 models. Karacı trained CNN models on the lung region images, which they detected with the YOLOv3 deep learning model and achieved very high success [28]. Nasser et al. proposed a smart healthcare system using IoT and cloud technology. The authors employed the ResNet50 deep learning model for COVID-19 detection in their suggested system [29]. Islam et al. detected COVID-19 using a hybrid architecture of CNN and recurrent neural networks (RNN). The authors employed CNN to extract complicated features from samples and RNN to classify them, and they reported that the VGG19-RNN architecture surpassed all other networks in terms of accuracy [30]. Li et al. used depth separable neural networks and extended and depth separable convolutional neural networks for detecting COVID-19. The first one employed LeNet-5, VGG16, and ResNet-18, while the second used VGG16 and ResNet-18 as its backbone [31].

The COVID-19 diagnosing process can be time-consuming and also error-prone for radiologists. On the other hand, the similarity between COVID-19 and pneumonia types makes it difficult for radiologists to diagnose the disease accurately. Deep learning techniques, which are faster than RT-PCR testing, could distinguish COVID-19 cases and non-COVID-19 cases [27]. Machine learning and deep neural networks are core components of current computer-assisted works for issues in medicine. In particular, deep learning presents good results in medical imaging due to its high feature extraction capability [24]. Generally, the researchers used the approaches summarized below to detect COVID-19 automatically.

- Extracting features with CNN models and then conducting machine learning experiments
- Applying data augmentation techniques on the datasets that have class-wise imbalanced
- Applying CNN hyperparameter tuning techniques
- Performing ensemble learning approaches such as stacking and boosting

This study addressed two distinct voting strategies that include hybrid models that presented promising results in the experiments. Furthermore, the contrast-limited adaptive histogram equalization (CLAHE) and median filter image pre-processing techniques are applied to the original images to extract meaningful feature maps and performance improvement. The models' performances are validated on the two-class CT scan images and three-class or X-ray images published in the Kaggle and Mendeley repositories, respectively. To the author's best knowledge, transformer architectures used in this study have not been conducted on these datasets so far. Overall, an efficient voting framework capable of performing high-accuracy classification called ETSVF-COVID19 is introduced. The letters 'E,' 'T,' 'S,' 'V,' and 'F' in the 'ETSVF-COVID19' refer to efficient, two, stage, voting, learning, and framework words, respectively. In brief, the primary contributions of this paper are summarized as follows.

- CLAHE and the median filter are conducted on the original images to improve the robustness of the proposed model.
- The deep feature maps of CT scan and X-ray images are extracted using cutting-edge transformer architectures.
- Machine learning experiments are conducted on the deep features.

The remainder of this paper is arranged as follows. Section 2 introduces a brief review of the methodology used in this study. Section 3 addresses the experiments, and Section 4 presents the results. Section 5 discusses the results with state-of-the-art related works in the literature. Finally, Section 6 summarizes the paper with concluding remarks and possible future works.

2 Methodology

2.1 Transformers

Transformers, proposed by Vaswani et al. [32], are widely used in spine models for natural language processing and computer vision tasks. An origin transformer has input and output sequences [33]. A transformer includes an encoder and decoder. The encoder's function is to encode the representation vector matrix, which is necessary for feature extraction. The decoder's function is to predict the output based on the encoded information's input matrix. The input sequence is converted into an output sequence by combining the two parts [34]. With their computational efficiency and scalability, transformers have played a role in many computer vision tasks [35]. A query and a set of key–value pairs are mapped to an output where the query, key, value, and output vectors are in the attention mechanism, an essential feature in a transformer. The output matrix is calculated as Equation 1 [32]:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

where Q, K, and V refer to the query, key, and value, respectively, and $1/\sqrt{d_k}$ is the scaling factor.

Transformer-based works, which have become prominent recently, are carried out, albeit limited. For example, Li et al. used a transformer fusing CNN architectures to segment malignant thyroid nodules automatically. The authors composed a multiscale fusion module to incorporate multiscale feature maps into their proposed model [36]. Hu et al. [37] used a transformer-based model to predict early Alzheimer's disease and obtained 77.2% classification accuracy. Leng et al. [34] employed a sensing transformer-based model developed for peripheral blood leukocytes and achieved a sensitivity performance of 94.3%.

2.2 Classification and performance evaluation

Classification in machine learning refers to a modeling task in which a class label is predicted for input data. For this purpose, the performance of a model built on a certain part of the dataset is verified on the remaining part. The classification performance of a model is measured using several metrics, such as accuracy, precision, sensitivity, and specificity in machine learning experiments. The mathematical expressions of the metrics for datasets having two classes are given in between Eqs. 2 and 5.

$$\text{Accuracy}(\text{Acc}) = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{TN} + \text{FP}} \tag{2}$$

$$\text{Precision (Pre)} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3}$$

$$\text{Sensitivity (Sen)} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

$$\text{Specificity}(\text{Spe}) = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{5}$$

where true negative TN, false positive FP, true positive TP, and false negative FN are the parameters that measure the performance of any classifier. Precision (Pre) refers to the ratio of the number of correctly classified positive samples to the number of positively classified samples. Accuracy (Acc) indicates the model's overall correct classification ratio. Sensitivity (Sen) denotes the rate of correctly classifying positive cases, whereas specificity (Spe) denotes the rate of negative cases correctly classified. In multi-class problems, the handled class is regarded as positive, while others are negative. Accordingly, class-wise measurements are obtained for each class by using the metrics in Eqs. 2, 3, 4, and 5. Then, the average measurement values are calculated using the expressions given in Eqs. 6 and 9.

Table 1 Information of the classifiers with default parameter settings

Classifier	Parameter	Explanation
MLP	Hidden_layer_sizes=100	Neuron number in the ith hidden layer
	Activation='relu	Activation function
	Solver='adam'	Weight optimization
	Learning_rate_init=0.001	Weight updates
	Early_stopping=False	Terminate model training
SVM	C=1.0	Regularization parameter
	Gamma='scale'	Kernel coefficient
	Kernel='rbf'	Radial basis function
RF	n_estimators=100	Number of trees
	Criterion='gini'	A quality measure of a split
	Max_depth=None	Maximum depth of a tree
	Max_features='sqrt'	Number of features for the best split
XGBoost	Gamma	Degree of conservative
	Learning_rate	Weight updates
	Max_depth	Maximum depth of a tree

Table 2 Libraries and frameworks used in this study

Task	Library
Image processing	Opencv, skimage, numpy
Feature extracting	PyTorch
Classifying	Scikit-learn, XGBoost
Plotting	Matplotlib

For a class, c

$$\text{AverageAcc} = \frac{1}{\#\text{classes}} \sum_{k=1}^{\#\text{classes}} \text{Acc}(c) \tag{6}$$

$$\text{Average Pre} = \frac{1}{\#\text{classes}} \sum_{k=1}^{\#\text{classes}} \text{Pre}(c) \tag{7}$$

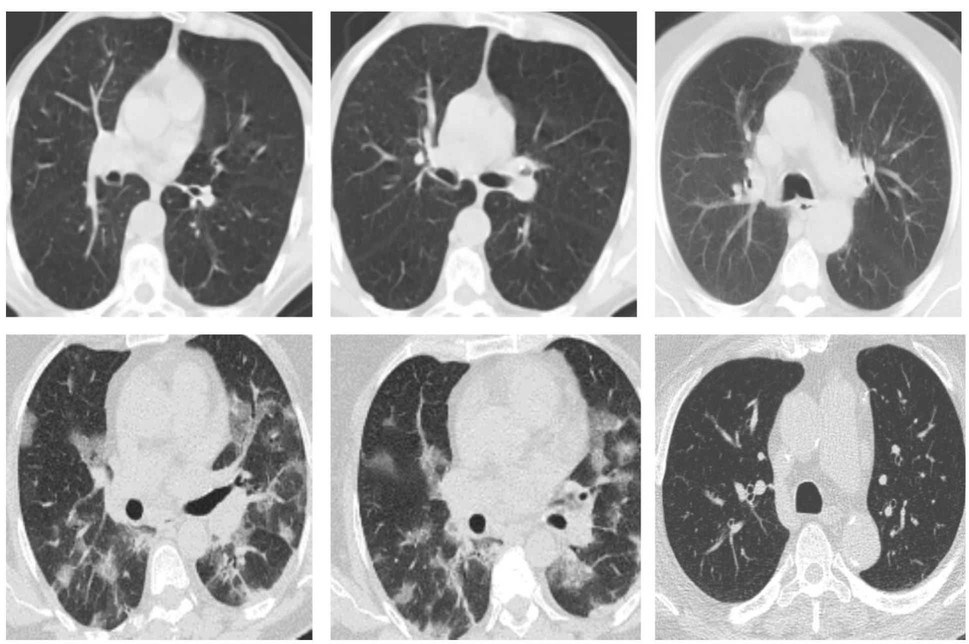

Figure 1 Some CT scan samples from Dataset #1 (from up to bottom rows; normal, COVID-19)

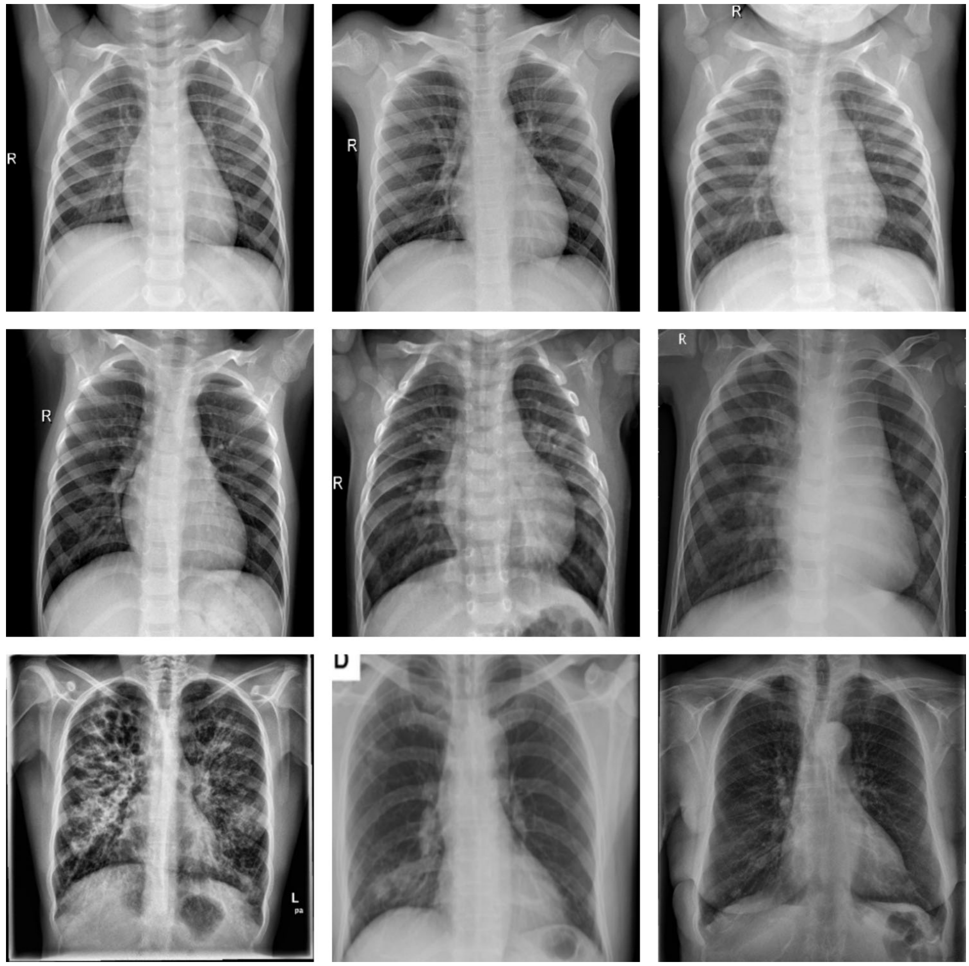

Figure 2 Some X-ray samples from Dataset #2 (from up to bottom rows; normal, pneumonia, COVID-19)

Table 3 Dataset information

Dataset	Original images	CLAHE-enhanced images	CLAHE-enhanced and median-filtered images
Dataset #1	Dataset #1.a	Dataset #1.b	Dataset #1.c
Dataset #2	Dataset #2.a	Dataset #2.b	Dataset #2.c

Table 4 Transformer-based deep learning architectures with parameter values

Parameter	Transformer architecture				
	LeViT	PoolFormer	Swin	SwinV2	ViT
Hidden_size(s)	[128, 256, 384]	[64, 128, 320, 512]	–	–	768
Patch_size(s)	16	[3, 7]	4	4	16
Hidden_act	–	'gelu'	'gelu'	'gelu'	'gelu'
Drop_path_rate	0	0.0	0.1	0.1	–

$$\text{AverageSen} = \frac{1}{\#\text{classes}} \sum_{k=1}^{\#\text{classes}} \text{Sen}(c) \quad (8)$$

$$\text{AverageSpe} = \frac{1}{\#\text{classes}} \sum_{k=1}^{\#\text{classes}} \text{Spe}\ (c) \quad (9)$$

Another metric used to verify the classification performance of a model is the receiver operating characteristic

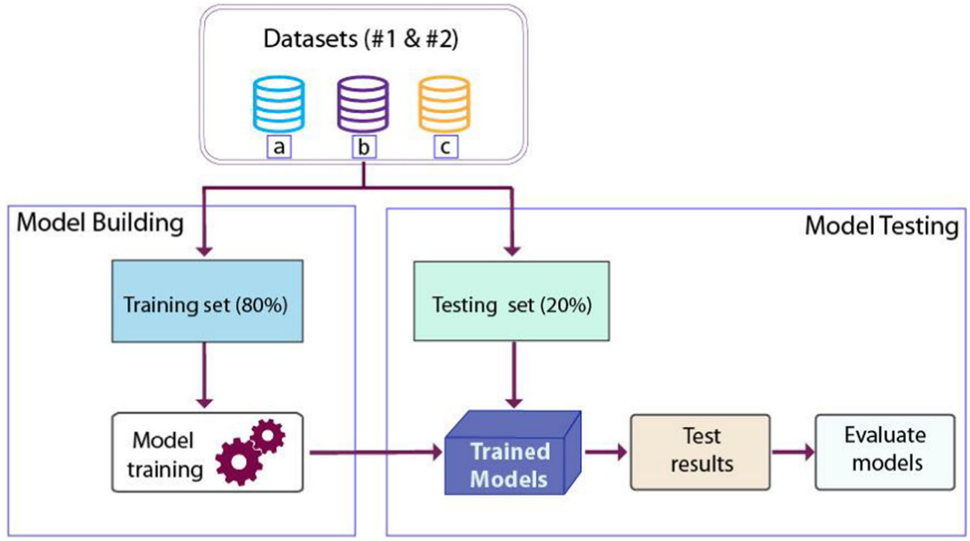

Figure 3 Workflow for evaluating hybrid models' performances

(ROC) curve. The ROC curve represents the relationship between true positive rates and false positive rates. The true positive rate refers to the ratio of accurately predicted positives to total positive data. The false positive rate refers to the ratio of incorrectly predicted negatives to total negatives. While the area under curve (AUC) score near one suggests that the model accurately identified the data, this score near zero indicates that a model misclassified samples [38]. Overall, the great area in the ROC curve and, thus, an AUC value close to one indicate that a model has high classification performance.

3 Experiments

3.1 Experimental environment

All experiments were performed with Python 3 Google Compute Engine backend (GPU) having 12.7 GB RAM and 15 GB GPU RAM configurations. The LeViT [39], ViT [40], PoolFormer [41], Swin [42], and SwinV2 [43] transformer architectures were utilized for deep feature maps. In addition, support vector machines (SVM) and multi-layer perceptrons (MLP) traditional classifiers and random forest (RF) and extreme gradient boosting (XGBoost) ensemble classifiers with default parameter settings were used for classification in machine learning experiments. Table 1 introduces the classifiers information used in this paper. Furthermore, the PyTorch [44], Scikit-image [45], Opencv [46], Matplotlib [47], Numpy [48], Scikit-learn [49], and XGBoost [50] libraries summarized in Table 2 were utilized in experiments.

3.2 Datasets

Two public datasets were used to validate the success of the proposed voting framework in this study. First, the publicly available SARS-CoV-2 CT scan dataset [51] was comprised by Soares et al. [51] from real patients in hospitals in Sao Paulo, Brazil, to stimulate the research and methods to detect whether a person is infected with SARS-CoV-2 by CT scan image. This dataset includes 1252 CT scans of COVID-19 positive patients and 1230 CT scans of uninfected patients. Second, the public chest X-ray PA dataset contains 6939 lung X-ray images collected from different sources [52]. The dataset has three classes: 2313 with COVID-19, 2313 with pneumonia, and 2313 with no diagnosis, or those categorized as normal. It noted that 'Dataset #1' and 'Dataset #2' indicate the first and second datasets used in this study. Figures 1 and 2 depict some chest CT scan and X-ray images from Dataset #1 and Dataset #2, respectively.

3.3 Pre-processing

CLAHE and median filter pre-processing techniques were also applied to the images in Dataset #1 and Dataset #2. The CLAHE corrects the difference in small areas of the image so that the histogram of these areas can conform to the specified histogram shape. Bilinear interpolation is used to connect adjacent tiles [53]. The median filter is applied to an image using a floating window. The median value of the corresponding window is used to replace the center pixel of an N*N neighborhood [54]. This study also obtained two datasets, including CLAHE-enhanced images and CLAHE-enhanced and median-filtered images, through the image pre-processing methods mentioned above. In this context, the CLAHE was applied to the

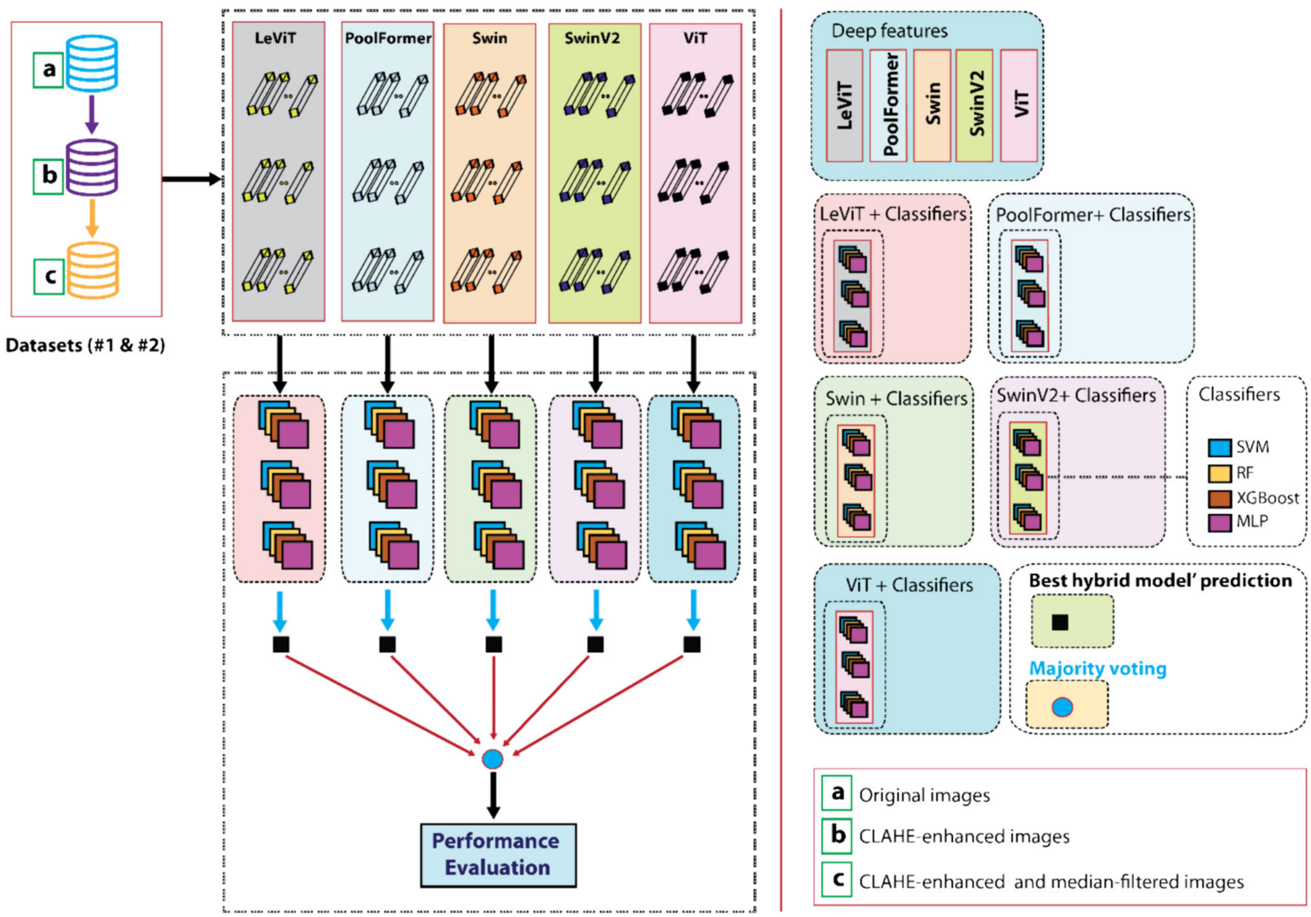

Figure 4 General block diagram of Scenario #1

original images, and then, the median filter was applied to the CLAHE-enhanced images. Accordingly, two datasets were included in experimental studies in addition to the original dataset. All images were resized to 224 × 224 for transformer architectures. Throughout the paper, the denotations for original images and pre-processed images are summarized in Table 3. Accordingly, 'Dataset #1.a' and 'Dataset #2.a' include the original images. 'Dataset #1.b' and 'Dataset #2.b' include the CLAHE-enhanced images, and 'Dataset #1.c' and 'Dataset #2.c' include the CLAHE-enhanced and median-filtered images, respectively.

3.4 Machine learning and voting strategies

Transformer-based architectures are commonly used for feature extraction and modeling tasks in machine learning. This study used transformer-based deep learning architectures to extract deep features from images. Transformer architectures were used with default parameter values that are summarized in Table 4. A hybrid model refers to the combines of a feature extractor and a classifier in this study. The feature extractor is located in the backbone of the hybrid model. For example, the feature extractor is ViT, and the classifier is MLP in the ViT+MLP hybrid model. This study employs several classifiers to determine the best classifier for deep features extracted from the medical images using transformer architectures. The classifiers were trained using the features extracted by transformer models from Dataset #1 and Dataset #2, and the best-performing classifiers for each transformer were detected. Classifiers' training and validation processes were conducted on the same training and testing sets with a fair approach. 80% of the datasets were reserved for training classifiers, and the remaining parts were reserved for evaluating the performances of these classifiers. Figure 3 demonstrates the workflow for model training and testing processes. Then, Scenario #1 and Scenario #2 were conducted on the classifiers' predictions to provide a robust framework for COVID-19 detection. The majority voting technique was carried out in both scenarios. Briefly, majority voting determines the class label that received the most votes from the predictions of all models as the final

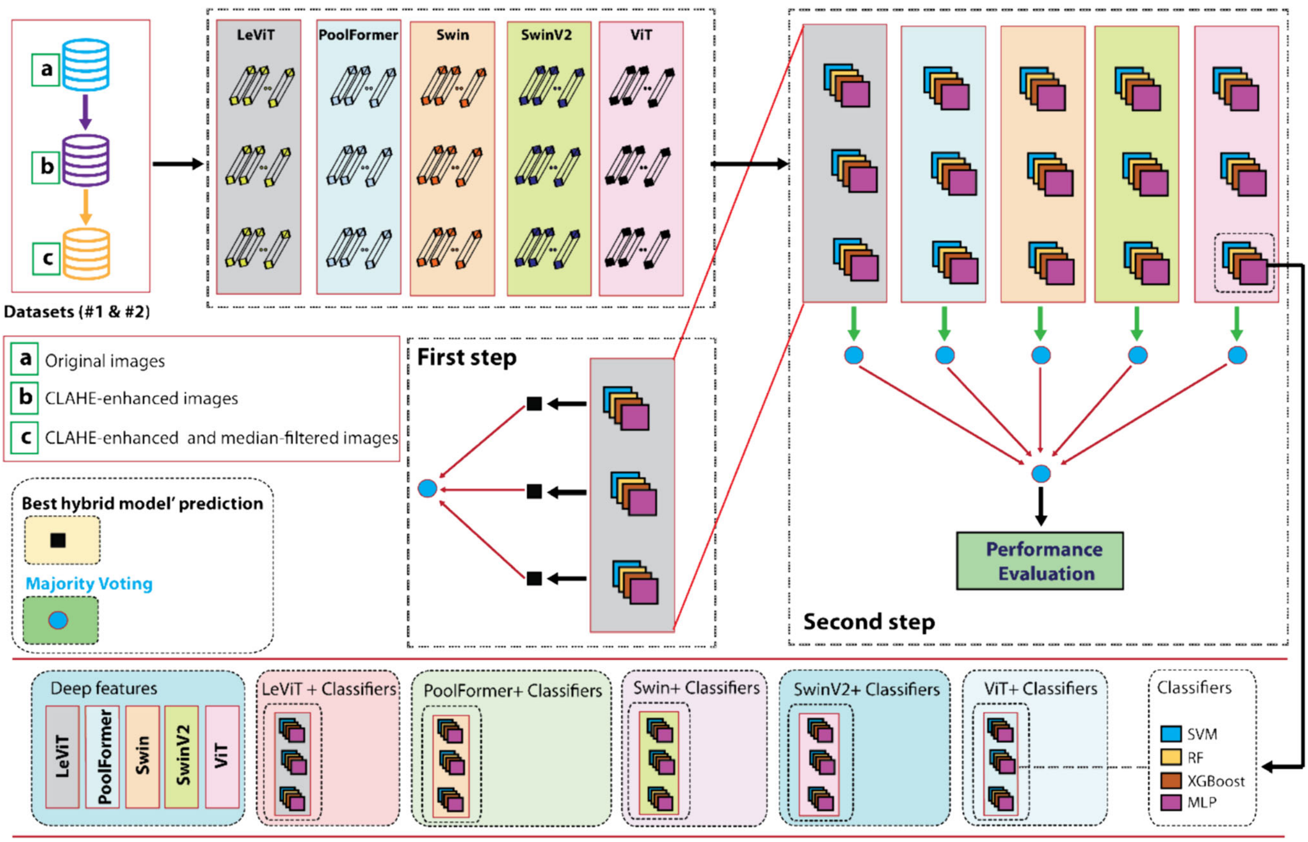

Figure 5 General block diagram of Scenario #2

result. To avoid repeatedly using the 'majority' word, only the word 'voting' was preferred throughout the paper.

Scenario #1 individually detects the classifiers that perform high performance on the deep feature sets for each of the three datasets and then applies the voting approach based on the predictions of these classifiers. For example, the classifier that presents high accuracy on the SwinV2 deep feature sets of Dataset #1.a, Dataset #1.b, and Dataset #1.c was determined. Likewise, other hybrid models that include best classifiers on other four feature sets were detected. Then, the voting of the predictions of these five models was conducted. Figure 4 demonstrates the workflow of Scenario #1.

Scenario #2 is a two-stage voting strategy called ETSVF-COVID19. The first step of this strategy detects the classifiers that present high accuracy on each of the deep feature sets extracted from each dataset separately and votes on the predictions of these models. Accordingly, the results of four classifiers on the deep feature sets of Dataset #1 and Dataset #2 were evaluated in the first step of this scenario. For example, the MLP classifier performed better than others on each of the ViT feature sets extracted from Dataset #1.a, Dataset #1.b, and Dataset #1.c. Therefore, predictions of ViT+MLP hybrid models were voted on in this scenario. The same process was carried out on the other deep feature sets. Thus, the first step was conducted. Then, the five prediction results of the first step were voted in the second step of this scenario. Figure 5 demonstrates the workflow of Scenario #2.

4 Results

In this study, the models' performances were verified on two public datasets. One of the datasets contains two-class CT scan images, while the other contains three-class X-ray images. In addition to the original images, experiments were conducted on the CLAHE-enhanced images and CLAHE-enhanced and median-filtered images, as well. Accuracy, sensitivity, specificity, and precision measurements of the hybrid models that were used in two scenarios are presented in Tables 5 and 7, respectively. It should be mentioned that the performance measurements summarized in the related tables and figures were obtained on the testing images for all experiments that the hybrid models had never seen before. Accordingly, Table 5 presents detailed performance measures for all classifiers on Datasets #1.a and #2.a that include original images. When evaluated using the accuracy measure, MLP outperformed others on all deep feature sets extracted from Dataset #1.a.

Table 5 Performance results of the hybrid models on original images

Transformer architectures	Classifier	Performance results (%)							
		Dataset #1.a				Dataset #2.a			
		Acc	Sen	Spe	Pre	Acc*	Sen*	Spe*	Pre*
LeViT	SVM	90.14	89.24	91.06	91.06	95.96	93.95	96.98	94.22
	RF	85.11	86.45	83.74	84.44	94.38	91.57	95.78	91.57
	XGBoost	84.91	84.46	85.37	85.48	95.72	93.59	96.79	93.66
	MLP	95.57	94.82	96.34	96.36	96.44	94.67	97.33	94.76
PoolFormer	SVM	93.56	91.24	95.93	95.82	96.78	95.17	97.59	95.32
	RF	87.93	86.06	89.84	89.63	94.62	91.93	95.97	91.95
	XGBoost	86.12	84.06	88.21	87.92	96.25	94.38	97.19	94.42
	MLP	95.57	96.02	95.12	95.26	96.93	95.39	97.69	95.55
Swin	SVM	92.35	89.24	95.53	95.32	96.64	94.96	97.48	95.25
	RF	89.34	88.05	90.65	90.57	94.77	92.14	96.08	92.23
	XGBoost	90.54	90.44	90.65	90.80	96.16	94.24	97.12	94.41
	MLP	96.18	95.62	96.75	96.77	96.45	94.67	97.33	94.80
SwinV2	SVM	92.96	91.24	94.72	94.63	96.73	95.10	97.55	95.26
	RF	90.54	90.84	90.24	90.48	95.15	92.72	96.36	92.83
	XGBoost	90.74	89.64	91.87	91.84	96.35	94.52	97.26	94.65
	MLP	97.59	97.61	97.56	97.61	96.54	94.81	97.40	94.93
ViT	SVM	90.14	86.85	93.50	93.16	96.16	94.24	97.12	94.70
	RF	88.13	88.84	87.40	87.80	95.77	93.66	96.83	93.77
	XGBoost	89.34	88.45	90.24	90.24	96.49	94.74	97.37	95.0
	MLP	94.57	95.62	93.50	93.75	96.83	95.24	97.62	95.32

*Indicates average measures

As seen in this table, the SwinV2+MLP hybrid model showed the best performance with 97.59% accuracy, 97.61% sensitivity, 97.56% specificity, and 97.61% precision. On the contrary, LeViT+XGBoost offered relatively poor performance with 84.91% accuracy, 84.46% sensitivity, 85.37% specificity, and 85.48% precision. On the other hand, the PoolFormer+MLP appears to offer the highest accuracy on Dataset 2.a. This model presented 96.93% average accuracy, 95.39% average sensitivity, 97.69% average specificity, and 95.55% average precision. In addition, LeViT+RF offered relatively low performance with 94.38% average accuracy, 91.57% average sensitivity, 95.78% average specificity, and 91.57% average precision on this dataset.

Table 6 presents detailed performance measures for all classifiers on Datasets #1.b and #2.b that include CLAHE-enhanced images. As seen in this table, the MLP classifier performed better than others on the deep feature sets of Dataset #1.b. The MLP classifier achieved the highest accuracy of 97.59% on the SwinV2 features set. Also, this classifier presented 94.77%, 95.98%, 94.97%, and 95.37% accuracies on the LeViT, PoolFormer, Swin, and ViT sets, respectively. In addition, LeViT+SVM, PoolFormer+MLP, Swin+MLP, SwinV2+MLP, and ViT+XGBoost presented best average accuracies with 96.06%, 96.01%, 96.49%, 95.96%, and 95.96%, respectively, for Dataset #2.b.

Table 7 presents detailed performance measures for all classifiers on Datasets #1.c and #2.c that include CLAHE-enhanced and median-filtered images. As seen in this table, the MLP classifier outperformed others for Dataset #1.c. This classifier offered the highest classification accuracies of 93.76%, 95.37%, 95.57%, 97.38%, and 94.37% on the LeViT, PoolFormer, Swin, SwinV2, and ViT sets, respectively. Furthermore, it appears that LeViT+MLP, PoolFormer+MLP, Swin+MLP, SwinV2+MLP, and ViT+MLP performed best average accuracies with 95.63%, 96.74%, 96.45%, 96.40%, and 96.06%, respectively, for Dataset #2.c.

Figure 6 presents the classification accuracies of hybrid models for Dataset #1.a, Dataset #1.b, and Dataset #1.c. Also, Fig. 7 presents the classification accuracies of each hybrid model for Dataset #2.a, Dataset #2.b, and Dataset #2.c. For example, the ViT+SVM hybrid model with a 90.14% accuracy value is shown by the first green color bar in Fig. 6a. The bold ones on the bars in these Fig. s represent the highest accuracies for related datasets in the experiments. Accordingly, the hybrid models that include the MLP classifier presented the highest accuracies for Dataset #1.a, Dataset #1.b, and Dataset #1.c. In addition, it

Table 6 Performance results of the hybrid models on CLAHE-enhanced images

Transformer architectures	Classifier	Performance results (%)							
		Dataset #1.b				Dataset #2.b			
		Acc	Sen	Spe	Pre	Acc*	Sen*	Spe*	Pre*
LeViT	SVM	89.74	87.65	91.87	91.67	96.06	94.09	97.05	94.26
	RF	85.71	85.26	86.18	86.29	94.04	91.06	95.53	91.05
	XGBoost	87.32	86.85	87.80	87.90	95.34	93.01	96.51	93.11
	MLP	94.77	94.82	94.72	94.82	95.53	93.30	96.65	93.39
PoolFormer	SVM	91.55	88.05	95.12	94.85	95.92	93.88	96.94	94.09
	RF	85.92	84.46	87.40	87.24	94.14	91.21	95.61	91.21
	XGBoost	88.93	86.45	91.46	91.18	95.39	93.08	96.54	93.19
	MLP	95.98	96.41	95.53	95.65	96.01	94.02	97.01	94.06
Swin	SVM	90.14	87.65	92.68	92.44	95.68	93.52	96.76	93.85
	RF	86.12	85.26	86.99	86.99	94.38	91.57	95.79	91.74
	XGBoost	88.33	87.25	89.43	89.39	95.25	92.87	96.44	93.04
	MLP	94.97	94.42	95.53	95.56	96.49	94.74	97.37	94.86
SwinV2	SVM	92.76	92.03	93.50	93.52	95.82	93.73	96.87	93.96
	RF	88.33	89.24	87.40	87.84	94.0	90.99	95.50	91.16
	XGBoost	90.95	89.64	92.28	92.21	95.49	93.23	96.61	93.46
	MLP	97.59	98.41	96.75	96.86	95.96	93.95	96.98	94.13
ViT	SVM	90.95	89.24	92.68	92.56	95.68	93.52	96.76	93.85
	RF	89.13	90.04	88.21	88.63	95.29	92.94	96.47	93.06
	XGBoost	88.73	89.64	87.80	88.24	95.97	93.95	96.97	94.10
	MLP	95.37	95.62	95.12	95.24	95.96	93.95	96.97	94.04

*Indicates average measures

can be stated that the models that mainly include MLP and SVM classifiers offer the highest accuracies for Dataset #2.

The accuracy results demonstrate that hybrid models, which integrate transformer-based deep features and classifiers, are satisfactory and reliable diagnostic models. In this context, the accuracy metric was utilized to identify hybrid models that should take part in Scenarios #1 and #2. It should be noted that when the hybrid models present the same accuracies, the model with the lower false negative value was preferred in two scenarios. As Kathamuthu et al. [55] stated, false positives and patient quarantine are preferable to mistaking a positive case for a negative one and allowing disease transmission. Fig. s 8 and 9 show the confusion matrix results of both scenarios for two-class and three-class datasets, respectively. Accordingly Fig. 8a presents Scenario #1's confusion matrix on Dataset #1. As seen here, three COVID-19 cases and five No COVID-19 cases were misclassified in this scenario. Similarly, Fig. 8b presents Scenario #2's confusion matrix on this dataset. Scenario #2 misclassified three COVID-19 cases and only one no COVID-19 case. In addition, it is seen that 6 COVID-19 images were classified as normal in Scenario #1 for Dataset #2. There are no COVID-19 images classified as pneumonia. 32 pneumonia images were classified as normal, and one was classified as COVID-19. Lastly, only five images in the normal class were classified as pneumonia in this scenario. Furthermore, it is seen that only one COVID-19 image was classified as normal, and two were classified as pneumonia in Scenario 2. Also, only one pneumonia image was classified as COVID-19, and 25 images in the pneumonia class were classified as Normal. Lastly, only one image in the normal class was classified as pneumonia. There are no images classified as COVID-19 for this class. Overall, the EFV4COVID-19 framework has fewer false positives and false negatives compared to Scenario #1 in experiments on both Dataset #1 and Dataset #2.

Table 8 shows the class-wise results and average measures of two scenarios, including the most successful hybrid models for Dataset #1 and Dataset #2. The accuracy, precision, sensitivity, and specificity measures in this table present the performances of two scenarios in this study. Overall, ETSVF-COVID19 has satisfactory classification performance in terms of these metrics. This framework correctly classified COVID-19 images at a ratio of 99.6% and correctly classified images with no findings at a rate of 98.78% in Dataset #1. In addition, COVID-19 images were classified correctly at a ratio of 98.8%, while images with no findings were classified correctly at a ratio of 97.97% in Scenario #1. This is also available for Dataset

Table 7 Performance results of the hybrid models on CLAHE-enhanced and median-filtered images

Transformer architectures	Classifier	Performance results (%)							
		Dataset #1.c				Dataset #2.c			
		Acc	Sen	Spe	Pre	Acc*	Sen*	Spe*	Pre*
LeViT	SVM	91.15	90.04	92.28	92.24	95.29	92.94	96.47	93.16
	RF	87.53	85.66	89.43	89.21	93.90	90.85	95.43	90.89
	XGBoost	86.32	86.85	85.77	86.17	95.0	92.51	96.26	92.65
	MLP	93.76	94.42	93.09	93.31	95.63	93.44	96.72	93.48
PoolFormer	SVM	91.55	90.44	92.68	92.65	96.49	94.74	97.37	94.93
	RF	86.72	86.85	86.59	86.85	94.57	91.86	95.93	91.89
	XGBoost	87.32	89.64	84.96	85.88	95.92	93.88	96.94	93.96
	MLP	95.37	97.21	93.50	93.85	96.74	95.10	97.55	95.22
Swin	SVM	92.96	90.44	95.53	95.38	95.96	93.95	96.98	94.29
	RF	87.53	87.25	87.80	87.95	94.57	91.86	95.93	91.96
	XGBoost	89.54	88.45	90.65	90.61	95.24	92.87	96.43	93.07
	MLP	95.57	95.22	95.93	95.98	96.45	94.67	97.33	94.80
SwinV2	SVM	93.36	92.03	94.72	94.67	95.73	93.59	96.79	93.82
	RF	89.34	88.45	90.24	90.24	93.95	90.92	95.46	91.06
	XGBoost	90.14	88.45	91.87	91.74	95.58	93.37	96.69	93.65
	MLP	97.38	98.41	96.34	96.48	96.40	94.60	97.30	94.76
ViT	SVM	89.74	88.45	91.06	90.98	95.05	92.58	96.29	92.84
	RF	84.51	86.06	82.93	83.72	94.72	92.07	96.04	92.10
	XGBoost	85.11	86.45	83.74	84.44	95.68	93.52	96.76	93.67
	MLP	94.37	96.41	92.28	92.72	96.06	94.09	97.04	94.12

*Indicates average measures

#2, where multi-class classification is conducted. EFV4-COVID-19 more correctly identified COVID-19 images than images in pneumonia and normal classes for Dataset #2. EFV4COVID-19 accurately identified COVID-19 images by 99.35%, pneumonia images by 94.38%, and normal images by 99.78%. As seen from experimental results, Scenario #1 and Scenario #2 improved the hybrid models' achievements for Dataset #1 and Dataset #2. It is noteworthy that EFV4COVID-19 correctly classified both true positives and true negatives at such a high rate in two-class classification and offered better performance than Scenario #1. Also, the average elapsed times of Scenario #1 and Scenario #2 for images in the testing sets of Dataset #1 and Dataset #2 were calculated. Accordingly, Scenario #1 and Scenario #2 performed the average diagnosis process on the 497 images that belong to Dataset #1's testing set with 2.388 and 7.008 s, respectively. In addition, Scenario #1 and Scenario #2 performed the average diagnosis process on the 1388 images that belong to Dataset #2's testing set with 2.394 and 6.756 s, respectively. The average diagnosis times of the proposed model are considered acceptable. In addition, it should not be overlooked that the system configuration has an effect on the diagnosing time. Overall, the absence of false negatives and false positives is especially desirable in the analysis of medical images. In this respect, it is very valuable that ETSVF-COVID19's diagnosis time is about 7 s.

Figure 10 depicts the ROC curves of Scenario #1 and Scenario #2 for Dataset #1. Also, Fig. 11 shows the ROC curves of both scenarios on the Dataset #2. The ROC curves especially show that Scenario #2 offered higher class-wise AUC values than Scenario #1 on Dataset #1 and Dataset #2. Overall, the ETSVF-COVID19 framework accurately classified testing images on both two-class and three-class datasets. This indicates that the framework can effectively distinguish between COVID-19 and non-COVID cases.

5 Discussion

By this time, many studies have been conducted on this disease. Table 9 summarizes the related works and this study. It is remarkable that many CNN-based works have been performed for COVID-19 disease and offered high classification accuracies. Balaha et al. [12] obtained 99.33% classification accuracy with stacked deep learning modeling. Khan et al. [15] selected the most distinctive ones from the features and reached 95.1% classification accuracy. Khurana and Soni achieved 98.9% accuracy

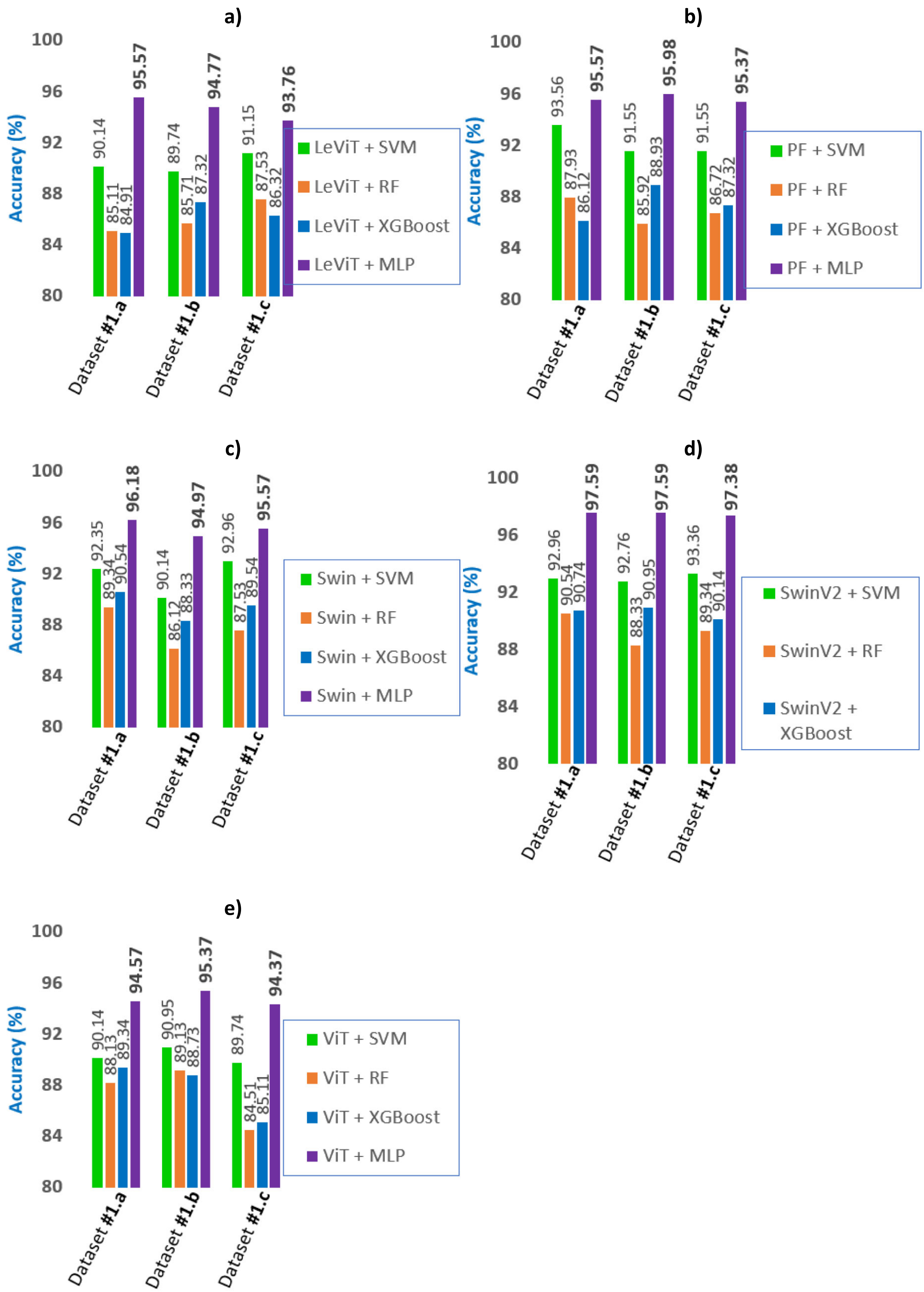

Fig. 6 Results of hybrid models on Dataset #1; **a** LeViT+Classifiers, **b** PoolFormer+Classifiers, **c** Swin+Classifiers, **d** SwinV2+Classifiers, **e** ViT+Classifiers

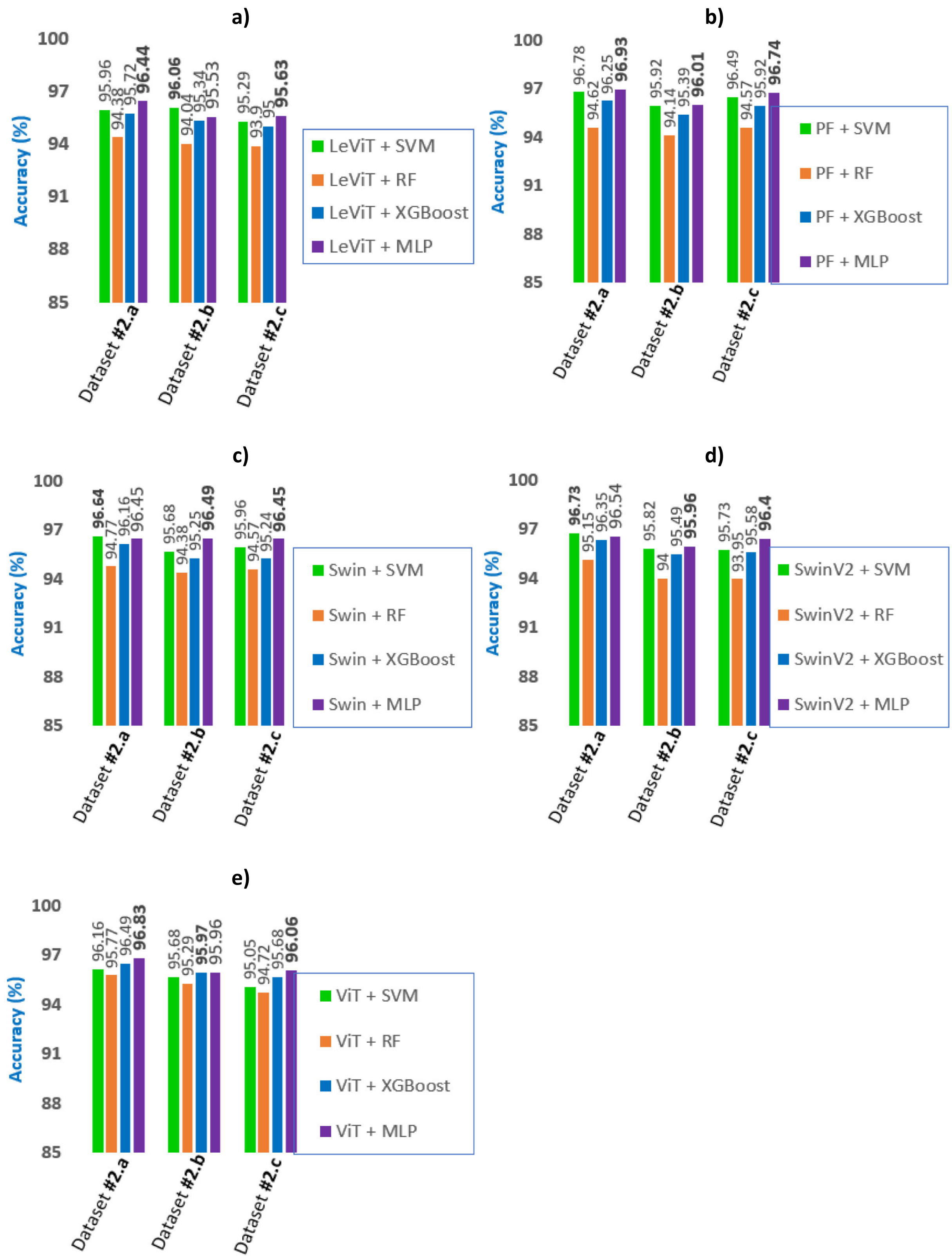

Fig. 7 Results of hybrid models on Dataset #2; **a** LeViT+Classifiers, **b** PoolFormer+Classifiers, **c** Swin+Classifiers, **d** SwinV2+Classifiers, **e** ViT+Classifiers

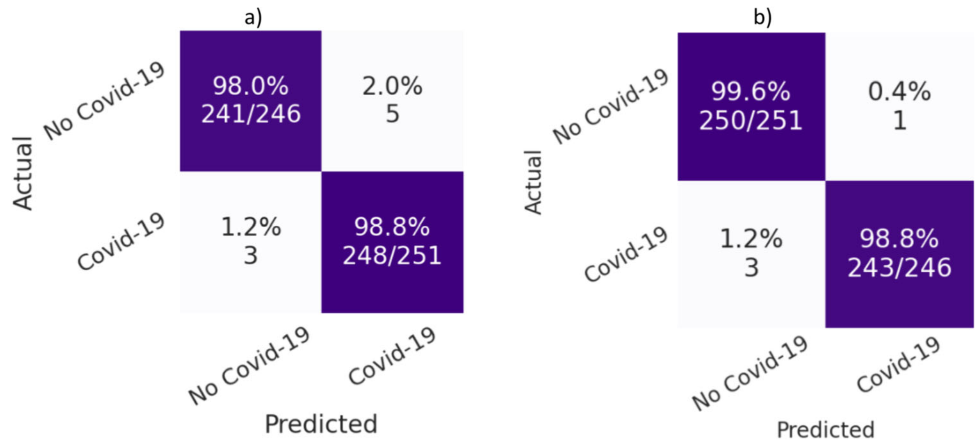

Figure 8 Confusion matrices of the both scenarios on Dataset #1; **a** Scenario #1 **b** Scenario #2

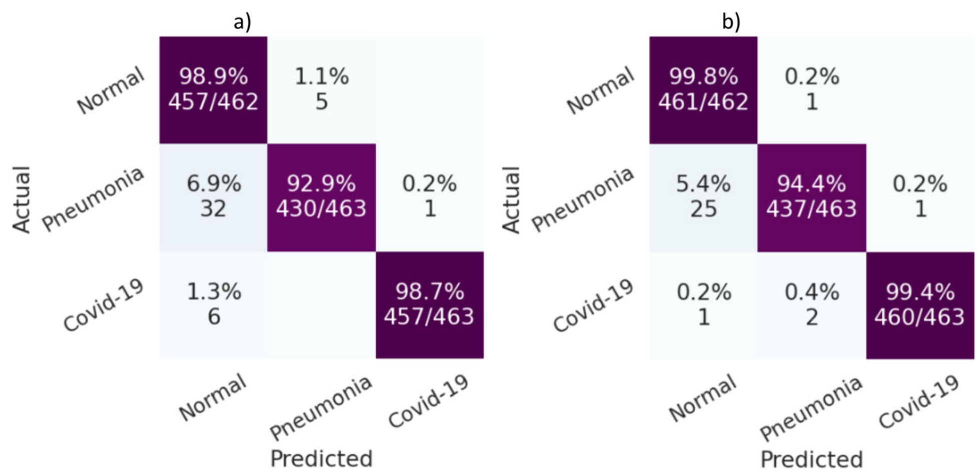

Figure 9 Confusion matrices of the both scenarios on Dataset #2; **a** Scenario #1 **b** Scenario #2

Table 8 Experimental results of the Scenario #1 and Scenario #2

Dataset	Scenario	Class	Sen (%)	Spe (%)	Pre (%)	Acc (%)	Elapsed time (sec)
Dataset #1	Scenario #1	No COVID-19	97.97	98.80	98.77		
		COVID-19	98.8	97.97	98.02		
		Average	98.39	98.39	98.40	98.39	2.388
	Scenario #2	No COVID-19	98.78	99.6	99.59		
		COVID-19	99.6	98.78	98.81		
		Average	99.19	99.19	99.20	99.20	7.008
Dataset #2	Scenario #1	Normal	98.92	95.90	92.32	96.90	
		Pneumonia	92.87	99.46	98.85	97.26	
		COVID-19	98.70	99.89	99.78	99.50	
		Average	96.83	98.42	96.98	97.89	2.394
	Scenario #2	Normal	99.78	97.19	94.66	98.05	
		Pneumonia	94.38	99.68	99.32	97.91	
		COVID-19	99.35	99.89	99.78	99.71	
		Average	97.84	98.92	97.92	98.56	6.756

with the ResNet50 deep learning network [16]. Loey et al. [18] achieved 82.91% classification accuracy with ResNet50 on the augmented images using conditional generative adversarial nets. Rahimzadeh et al. [22] achieved 98.49% classification accuracy by the deep learning approach they proposed on automatically selected fertile regions of CT scan images to detect COVID-19 disease. Kogilavani et al. [56] obtained a 97.68% value of accuracy using the VGG16 deep learning network. Ardakani et al. [57] achieved 99.51% classification accuracy

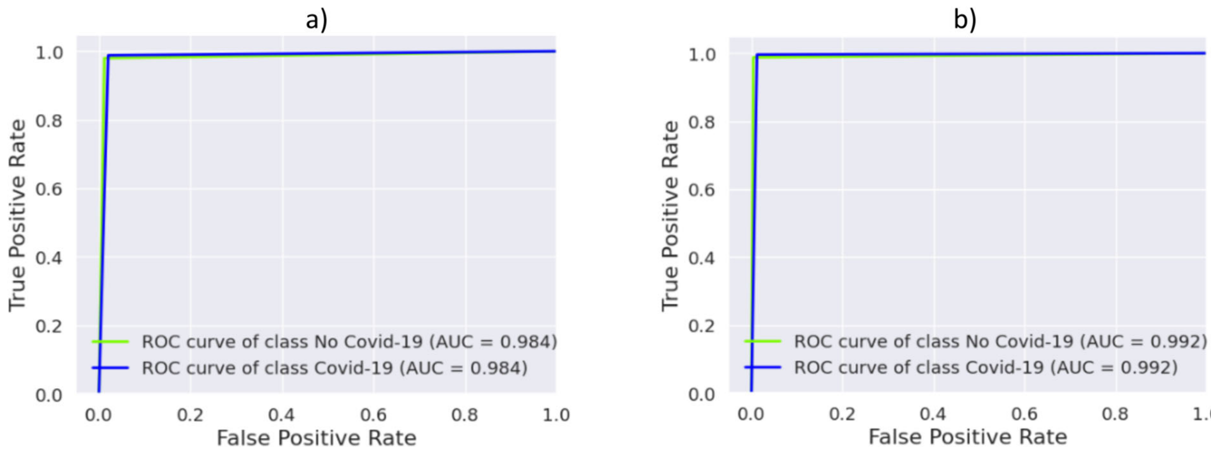

Figure 10 ROC curves with AUC values of both scenarios on Dataset #1; **a** Scenario #1 **b** Scenario #2

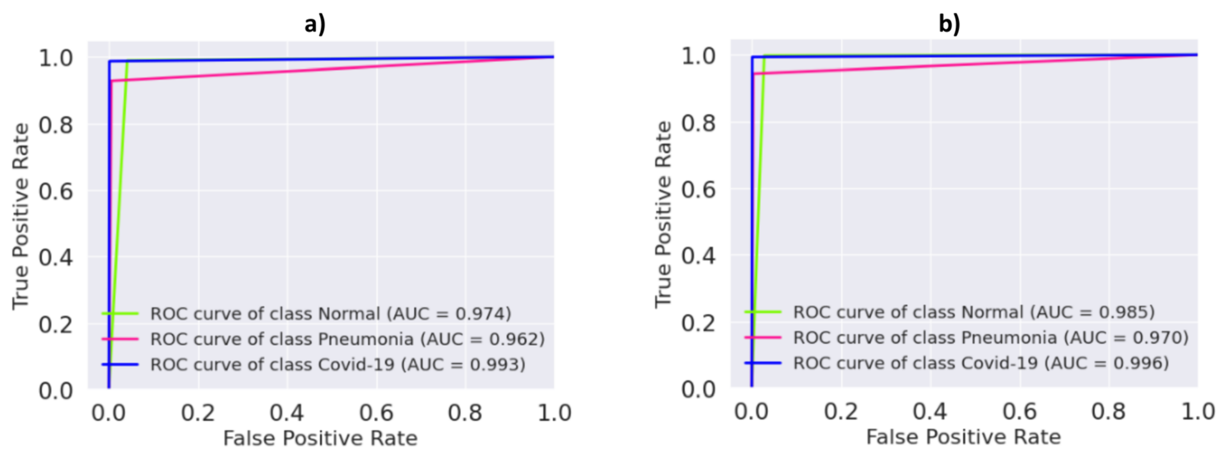

Figure 11 ROC curves with AUC values of both scenarios on Dataset #2; a) Scenario #1 b) Scenario #2

through ResNet-101 and Xception pre-trained networks in their studies via various pre-trained models. The following two works in the two-class section of this table used the same dataset as this study. Ahmad et al. [5] achieved 98.56% classification accuracy. Gupta and Bajaj reached 98.91% classification accuracy with DarkNet19 deep learning model [9]. Zhang et al. [25] introduced a deep comparative mutual learning name model and got 97.42% classification accuracy on the dataset containing enriched SARS-CoV-2 CT scan images. Furthermore, it is available in three-class works. Shi et al. [10] achieved 96.25% accuracy with their proposed lightweight networks model. Kavya et al. [26] achieved an average accuracy of 91.39% with the ResNet50 model in their study for COVID-19 detection. Budak et al. [27] achieved 98% diagnosis accuracy with the DenseNet121 and DenseNet201 pre-trained models. Nasser et al. [29] achieved 98.6% accuracy with the ResNet50 deep learning model in the IoT-based system they proposed for COVID-19 detection. The last two works in the three-class section of this table used the same dataset as this study. Islam et al. [30] achieved 99.86% accuracy with their proposed VGG19-RNN architecture. Li et al. [31] achieved 93.63 $\pm$ 0.76% success with dilated and depthwise separable convolutional neural networks for COVID-19 detection. Lastly, Karacı proposed a cascade deep learning model and achieved 99.84% and 97.16% accuracies in the two-class and three-class datasets, respectively [28].

To the author's knowledge, the previous works include feature engineering and classification approaches. Those studies revealed that various CNN-based experiments are successful in detecting COVID-19 disease. This study aimed to create a robust framework for COVID-19 detection. With this aim, the classifiers were trained on deep feature maps extracted with transformer architectures. Then, the results of the classifiers that reached the best performance on the datasets were voted in two scenarios. The findings revealed the effectiveness and potential of ETSVF-COVID19 in contributing to the detection of COVID-19 by field experts. Particularly, achieving high success on both datasets is considered to be the success of the proposed artificial intelligence model. Experiments

Table 9 Comparing the ETSVF-COVID19 with the existing studies.

Number of classes	Study	Model	Dataset	Acc (%)
Two-class	Balaha et al. [12]	Stacked deep learning	Other	99.33
	Khan et al. [15]	Deep features and classifier	Other	95.1
	Khurana and Soni [16]	ResNet50	Other	98.9
	Loey et al. [18]	Conditional generative adversarial nets + ResNet50	Other	82.91
	Rahimzadeh et al. [22]	ResNet50V2, feature pyramid network, and classification	Other	98.49%
	Karacı [28]	VGGCOV19-NET	Other	99.84
	Kogilavani et al. [56]	VGG16	Other	97.68
	Ardakani et al. [57]	ResNet-101	Other	99.51
	Ahmad et al. [5]	Lightweight ResGRU	Same	98.56
	Gupta and Bajaj [9]	DarkNet19	Same	98.91
	Zhang et al. [25]	Deep contrastive mutual learning	Same	97.42
	This study	**ETSVF-COVID19**	**Same**	**99.2**
Three-class	Shi et al. [10]	Lightweight networks model	Other	96.25
	Kavya et al. [26]	ResNet50	Other	91.39*
	Budak et al. [27]	DenseNet121 and DenseNet201	Other	98
	Karacı [28]	VGGCOV19-NET	Other	97.16
	Nasser et al. [29]	ResNet50	Other	98.6*
	Islam et al. [30]	VGG19-RNN	Same	99.86
	Li et al. [31]	Dilated and depthwise separable convolutional neural networks	Same	93.63± 0.76
	This study	**ETSVF-COVID19**	**Same**	**98.56**

*Average accuracy value

The result of ETSVF-COVID19 is in bold font

showed that the ETSVF-COVID19 is a promising approach with achieved competitive accuracy for detecting COVID-19 on both CT scan and X-ray images. As a result, the ETSVF-COVID19 demonstrated a strong performance in COVID-19 detection, comparable to other state-of-the-art methods in the literature. Overall, this study provides valuable information on using transformer architectures as feature extractors for diagnosing COVID-19 disease; it should be noted that this study also has some limitations addressed below.

Experiments with various paradigms highlight the superior performance of ETSVF-COVID19, supporting its broader applicability and generalizability for COVID-19 detection. However, building a more robust model based on domain adaptation by working with datasets from different sources is inevitable for an artificial intelligence system that provides more reliable decision support for field experts. So, field experts can reach for a clear distinction between classes, allowing for more accurate predictions. In addition, applying feature engineering methods such as feature fusion and feature selection on feature maps obtained from different dataset sources will be the right operations to demonstrate the robustness of the classifier model. The second limitation is that transformer-based features were sent directly to machine learning algorithms in this study. Although the best of the four algorithms was detected in experimental studies, hyperparameter tuning was not addressed in experiments. More successful models could be built with hyperparameter tuning for classifiers needed for expert systems that will assist field experts. The third limitation is that the datasets used in this study contain two classes and three classes. Modeling studies on datasets, including other lung diseases, will be an even more appropriate approach.

6 Conclusion

This paper introduces ETSVF-COVID19, which includes deep learning-based state-of-the-art models and a two-stage voting strategy, which differs from other voting approaches in the literature for COVID-19 diagnosis detection. Here, transformer architectures acted as the backbone of the hybrid models. Detailed experiments were carried out on two publicly available datasets containing 2482 and 6939 images, using accuracy, specificity, sensitivity, and precision performance metrics to assess the performances of the proposed approach. ETSVF-COVID19 presented 99.20% accuracy, 99.19% sensitivity, 99.19% specificity, and 99.20% precision on Dataset #1 and 98.56% average accuracy, 97.84% average sensitivity, 98.92% average specificity, and 97.92% average precision on Dataset #2. Consequently, ETSVF-COVID19 presented promising results on the CT scans and X-ray images for COVID-19 detection. Thus, this framework could be helpful for radiologists and favorable for use in real-world scanning environments. Furthermore, this study has the importance of providing decision support to field experts in taking rapid measures for the isolation of patients infected with COVID-19 and its variants in order to slow the spread of the virus. The suggestions are listed below:

- New transformer models could be included to design more efficient hybrid frameworks.
- Integration of transformer and fusion techniques could lead to more good results.
- The ETSVF-COVID19 could be used in other computer vision experiments.

It is planned to investigate more effective fusion strategies to enhance the detection of COVID-19-infected patients. Moreover, working with other diseases to improve the model's robustness is among the target works.

Acknowledgments The author would like to thank Soares et al. [51] and Asraf and Islam [52] for providing the public datasets.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK). The author declares that no funds, grants, or other support was received during the preparation of this manuscript.

Data availability and access The datasets used in the current study are available at the below link: Dataset #1: https://www.kaggle.com/datasets/plameneduardo/sarscov2-ctscan-dataset. Dataset #2: https://data.mendeley.com/datasets/mxc6vb7svm/2

Declarations

Conflict of interest The author has no relevant financial or nonfinancial interests to disclose.

Ethical standards This research did not require ethics approval.

References

1. Acar E, Şahin E, Yılmaz İ (2021) Improving effectiveness of different deep learning-based models for detecting COVID-19 from computed tomography (CT) images. Neural Comput Appl 33:17589–17609. https://doi.org/10.1007/S00521-021-06344-5
2. WHO (2023) Coronavirus disease (COVID-19). https://www.who.int/emergencies/diseases/novel-coronavirus-2019/question-and-answers-hub/q-a-detail/coronavirus-disease-covid-19. Accessed 19 Feb 2023
3. Basiri ME, Nemati S, Abdar M et al (2021) A novel fusion-based deep learning model for sentiment analysis of COVID-19 tweets. Knowl Based Syst 228:107242. https://doi.org/10.1016/J.KNOSYS.2021.107242
4. Nüsslin F (2020) Wilhelm Conrad Röntgen: the scientist and his discovery. Phys Med 79:65–68. https://doi.org/10.1016/J.EJMP.2020.10.010
5. Ahmad M, Bajwa UI, Mehmood Y, Anwar MW (2023) Lightweight ResGRU: a deep learning-based prediction of SARS-CoV-2 (COVID-19) and its severity classification using multimodal chest radiography images. Neural Comput Appl. https://doi.org/10.1007/S00521-023-08200-0
6. Zu ZY, Di Jiang M, Xu PP et al (2020) Coronavirus disease 2019 (COVID-19): a perspective from China. Radiology 296:E15–E25. https://doi.org/10.1148/RADIOL.2020200490
7. Fang Y, Zhang H, Xie J et al (2020) Sensitivity of chest CT for COVID-19: comparison to RT-PCR. Radiology 296:E115–E117. https://doi.org/10.1148/RADIOL.2020200432
8. He K, Zhao W, Xie X et al (2021) Synergistic learning of lung lobe segmentation and hierarchical multi-instance classification for automated severity assessment of COVID-19 in CT images. Pattern Recognit 113:107828. https://doi.org/10.1016/J.PATCOG.2021.107828
9. Gupta K, Bajaj V (2023) Deep learning models-based CT-scan image classification for automated screening of COVID-19. Biomed Signal Process Control 80:104268. https://doi.org/10.1016/J.BSPC.2022.104268
10. Shi Y, Tang A, Xiao Y, Niu L (2023) A lightweight network for COVID-19 detection in X-ray images. Methods 209:29–37. https://doi.org/10.1016/J.YMETH.2022.11.004
11. Babukarthik RG, Chandramohan D, Tripathi D et al (2022) COVID-19 identification in chest X-ray images using intelligent multi-level classification scenario. Comput Electr Eng 104:108405. https://doi.org/10.1016/J.COMPELECENG.2022.108405
12. Balaha HM, El-Gendy EM, Saafan MM (2021) CovH2SD: a COVID-19 detection approach based on Harris Hawks optimization and stacked deep learning. Expert Syst Appl 186:115805. https://doi.org/10.1016/J.ESWA.2021.115805
13. Hosseinzadeh H (2022) Deep multi-view feature learning for detecting COVID-19 based on chest X-ray images. Biomed Signal Process Control 75:103595. https://doi.org/10.1016/J.BSPC.2022.103595
14. Jalali SMJ, Ahmadian M, Ahmadian S et al (2022) X-ray image based COVID-19 detection using evolutionary deep learning approach. Expert Syst Appl 201:116942. https://doi.org/10.1016/J.ESWA.2022.116942
15. Khan MA, Kadry S, Zhang YD et al (2021) Prediction of COVID-19—pneumonia based on selected deep features and one class kernel extreme learning machine. Comput Electr Eng 90:106960. https://doi.org/10.1016/J.COMPELECENG.2020.106960
16. Khurana Y, Soni U (2022) Leveraging deep learning for COVID-19 diagnosis through chest imaging. Neural Comput Appl 34:14003–14012. https://doi.org/10.1007/S00521-022-07250-0

17. Liu J, Sun W, Zhao X et al (2022) Deep feature fusion classification network (DFFCNet): towards accurate diagnosis of COVID-19 using chest X-rays images. Biomed Signal Process Control 76:103677. https://doi.org/10.1016/J.BSPC.2022.103677
18. Loey M, Manogaran G, Khalifa NEM (2020) A deep transfer learning model with classical data augmentation and CGAN to detect COVID-19 from chest CT radiography digital images. Neural Comput Appl. https://doi.org/10.1007/S00521-020-05437-X
19. JavadiMoghaddam SM, Gholamalinejad H (2021) A novel deep learning based method for COVID-19 detection from CT image. Biomed Signal Process Control 70:102987. https://doi.org/10.1016/J.BSPC.2021.102987
20. Nayak SR, Nayak DR, Sinha U et al (2021) Application of deep learning techniques for detection of COVID-19 cases using chest X-ray images: a comprehensive study. Biomed Signal Process Control 64:102365. https://doi.org/10.1016/J.BSPC.2020.102365
21. Nigam B, Nigam A, Jain R et al (2021) COVID-19: automatic detection from X-ray images by utilizing deep learning methods. Expert Syst Appl 176:114883. https://doi.org/10.1016/J.ESWA.2021.114883
22. Rahimzadeh M, Attar A, Sakhaei SM (2021) A fully automated deep learning-based network for detecting COVID-19 from a new and large lung CT scan dataset. Biomed Signal Process Control 68:102588. https://doi.org/10.1016/J.BSPC.2021.102588
23. Roy PK, Kumar A (2022) Early prediction of COVID-19 using ensemble of transfer learning. Comput Electr Eng 101:108018. https://doi.org/10.1016/J.COMPELECENG.2022.108018
24. Emin Sahin M (2022) Deep learning-based approach for detecting COVID-19 in chest X-rays. Biomed Signal Process Control 78:103977. https://doi.org/10.1016/J.BSPC.2022.103977
25. Zhang H, Liang W, Li C et al (2022) DCML: Deep contrastive mutual learning for COVID-19 recognition. Biomed Signal Process Control 77:103770. https://doi.org/10.1016/J.BSPC.2022.103770
26. Sri Kavya N, Shilpa T, Veeranjaneyulu N, DivyaPriya D (2022) Detecting Covid19 and pneumonia from chest X-ray images using deep convolutional neural networks. Mater Today Proc 64:737–743. https://doi.org/10.1016/J.MATPR.2022.05.199
27. Budak C, Mençik V, Varışlı O (2023) Online diagnosis of COVID-19 from chest radiography images by using deep learning algorithms. Neural Comput Appl 35:20717–20734. https://doi.org/10.1007/S00521-023-08867-5
28. Karacı A (2022) VGGCOV19-NET: automatic detection of COVID-19 cases from X-ray images using modified VGG19 CNN architecture and YOLO algorithm. Neural Comput Appl 34:8253–8274. https://doi.org/10.1007/S00521-022-06918-X
29. Nasser N, Emad-ul-Haq Q, Imran M et al (2023) A smart healthcare framework for detection and monitoring of COVID-19 using IoT and cloud computing. Neural Comput Appl 35:13775–13789. https://doi.org/10.1007/S00521-021-06396-7
30. Islam MM, Islam MZ, Asraf A et al (2022) Diagnosis of COVID-19 from X-rays using combined CNN-RNN architecture with transfer learning. BenchCouncil Trans Benchmarks Stand Eval 2:100088. https://doi.org/10.1016/J.TBENCH.2023.100088
31. Li X, Zhai M, Sun J (2021) DDCNNC: dilated and depthwise separable convolutional neural Network for diagnosis COVID-19 via chest X-ray images. Int J Cognit Comput Eng 2:71–82. https://doi.org/10.1016/J.IJCCE.2021.04.001
32. Vaswani A, Shazeer N, Parmar N, et al (2017) Attention is all you need. Advances in Neural Information Processing Systems 2017-December, pp 5999–6009. https://doi.org/10.48550/arxiv.1706.03762
33. Wei Z, Ji X, Zhou L et al (2022) A novel deep learning model based on target transformer for fault diagnosis of chemical process. Process Saf Environ Prot 167:480–492. https://doi.org/10.1016/J.PSEP.2022.09.039
34. Leng B, Wang C, Leng M et al (2023) Deep learning detection network for peripheral blood leukocytes based on improved detection transformer. Biomed Signal Process Control 82:104518. https://doi.org/10.1016/J.BSPC.2022.104518
35. Yang M, Wu C, Guo Y et al (2023) Transformer-based deep learning model and video dataset for unsafe action identification in construction projects. Autom Constr 146:104703. https://doi.org/10.1016/J.AUTCON.2022.104703
36. Li G, Chen R, Zhang J et al (2023) Fusing enhanced transformer and large kernel CNN for malignant thyroid nodule segmentation. Biomed Signal Process Control 83:104636. https://doi.org/10.1016/J.BSPC.2023.104636
37. Hu Z, Wang Z, Jin Y, Hou W (2023) VGG-TSwinformer: transformer-based deep learning model for early Alzheimer's disease prediction. Comput Methods Programs Biomed 229:107291. https://doi.org/10.1016/J.CMPB.2022.107291
38. Fawcett T (2006) An introduction to ROC analysis. Pattern Recognit Lett 27:861–874. https://doi.org/10.1016/J.PATREC.2005.10.010
39. Graham B, El-Nouby A, Touvron H et al (2021) LeViT: a vision transformer in ConvNet's clothing for faster inference. In: Proceedings of the IEEE international conference on computer vision, pp 12239–12249. https://doi.org/10.48550/arxiv.2104.01136
40. Dosovitskiy A, Beyer L, Kolesnikov A et al (2020) An image is worth 16x16 words: transformers for image recognition at scale. https://doi.org/10.48550/arxiv.2010.11929
41. Yu W, Luo M, Zhou P et al (2021) MetaFormer Is actually what you need for vision. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition 2022-June, pp 10809–10819. https://doi.org/10.48550/arxiv.2111.11418
42. Liu Z, Lin Y, Cao Y et al (2021) Swin transformer: hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE international conference on computer vision, pp 9992–10002. https://doi.org/10.48550/arxiv.2103.14030
43. Liu Z, Hu H, Lin Y, et al (2021) Swin transformer V2: scaling up capacity and resolution. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition 2022-June, pp 11999–12009. https://doi.org/10.48550/arxiv.2111.09883
44. PyTorch. https://pytorch.org/tutorials/beginner/colab.html. Accessed 26 Apr 2024
45 Van Der Walt S, Schönberger JL, Nunez-Iglesias J et al (2014) Scikit-image: Image processing in python. PeerJ. https://doi.org/10.7717/PEERJ.453
46. OpenCV—Open Computer Vision Library. https://opencv.org/. Accessed 7 May 2024
47. Matplotlib—Visualization with Python. https://matplotlib.org/. Accessed 7 May 2024
48. NumPy. https://numpy.org/. Accessed 7 May 2024
49. Scikit-learn: machine learning in Python. https://scikit-learn.org/stable/. Accessed 7 May 2024
50. XGBoost. https://xgboost.readthedocs.io/en/stable/. Accessed 7 May 2024
51 Soares E, Angelov P, Biaso S et al (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

52. Asraf A, Islam Z (2021) COVID19, Pneumonia and normal chest X-ray PA dataset. 2. https://doi.org/10.17632/MXC6VB7SVM.2
53. Hayati M, Muchtar K, Roslidar et al (2023) Impact of CLAHE-based image enhancement for diabetic retinopathy classification through deep learning. Procedia Comput Sci 216:57–66. https://doi.org/10.1016/J.PROCS.2022.12.111
54. Sejuti ZA, Islam MS (2023) A hybrid CNN–KNN approach for identification of COVID-19 with 5-fold cross validation. Sens Int 4:100229. https://doi.org/10.1016/J.SINTL.2023.100229
55. Kathamuthu ND, Subramaniam S, Le QH et al (2023) A deep transfer learning-based convolution neural network model for COVID-19 detection using computed tomography scan images for medical applications. Adv Eng Softw 175:103317. https://doi.org/10.1016/J.ADVENGSOFT.2022.103317
56. Kogilavani SV, Prabhu J, Sandhiya R et al (2022) COVID-19 detection based on lung CT scan using deep learning techniques. Comput Math Methods Med. https://doi.org/10.1155/2022/7672196
57. Ardakani AA, Kanafi AR, Acharya UR et al (2020) Application of deep learning technique to manage COVID-19 in routine clinical practice using CT images: results of 10 convolutional neural networks. Comput Biol Med 121:103795. https://doi.org/10.1016/J.COMPBIOMED.2020.103795