Computer Systems Science & Engineering
DOI: 10.32604/csse.2023.034449
Article

Tech Science Press

Quantum Inspired Differential Evolution with Explainable Artificial Intelligence-Based COVID-19 Detection

Abdullah M. Basahel and Mohammad Yamin*

Department of Management Information Systems, Faculty of Economics and Administration, King Abdulaziz University, Jeddah, 21589, Saudi Arabia
*Corresponding Author: Mohammad Yamin. Email: myamin@kau.edu.sa

Received: 17 July 2022; Accepted: 11 October 2022

Abstract: Recent advancements in the Internet of Things (Io), 5G networks, and cloud computing (CC) have led to the development of Human-centric IoT (HIoT) applications that transform human physical monitoring based on machine monitoring. The HIoT systems find use in several applications such as smart cities, healthcare, transportation, etc. Besides, the HIoT system and explainable artificial intelligence (XAI) tools can be deployed in the healthcare sector for effective decision-making. The COVID-19 pandemic has become a global health issue that necessitates automated and effective diagnostic tools to detect the disease at the initial stage. This article presents a new quantum-inspired differential evolution with explainable artificial intelligence based COVID-19 Detection and Classification (QIDEXAI-CDC) model for HIoT systems. The QIDEXAI-CDC model aims to identify the occurrence of COVID-19 using the XAI tools on HIoT systems. The QIDEXAI-CDC model primarily uses bilateral filtering (BF) as a preprocessing tool to eradicate the noise. In addition, RetinaNet is applied for the generation of useful feature vectors from radiological images. For COVID-19 detection and classification, quantum-inspired differential evolution (QIDE) with kernel extreme learning machine (KELM) model is utilized. The utilization of the QIDE algorithm helps to appropriately choose the weight and bias values of the KELM model. In order to report the enhanced COVID-19 detection outcomes of the QIDEXAI-CDC model, a wide range of simulations was carried out. Extensive comparative studies reported the supremacy of the QIDEXAI-CDC model over the recent approaches.

Keywords: Human-centric IoT; quantum computing; explainable artificial intelligence; healthcare; COVID-19 diagnosis

1 Introduction

Several techniques have recently been developed in distinct fields, like 5G which allows ubiquitous cloud services and wireless transmission anywhere. This technology motivates Human-centric IoT Application (HIoT) that converts human physical monitoring depending on machine monitoring [1]. This HIoT system is distributed geographically and has multiple concerns, including anomaly avoidance, Industrial IoT (IIoT) data security, delay, automation, mobility, and services cost [2]. But IIoT Application or method is static or fixed. This complicated dynamic problem could not be resolved using the current approaches. Thus, space complexity is ubiquitous and dynamic uncertainty with time poses several study problems [3]. Quantum computing has the possibility of improving the study of IIoT Applications. The use case also considerably assists advanced Human-centric IoT Application quadruple purpose. It assists a product's lifecycle, from manufacturing and design to remanufacturing, utilization, and service [4]. The IoT is no longer a trend that has been demonstrated to be a very effective technique that current digital-based businesses use for rapidly expanding operations. The IoT offers real-time data, i.e., crucial to making sudden decisions according to distinct effective AI approaches [5]. Fig. 1 illustrates the opinion of Human-Centered Computing.

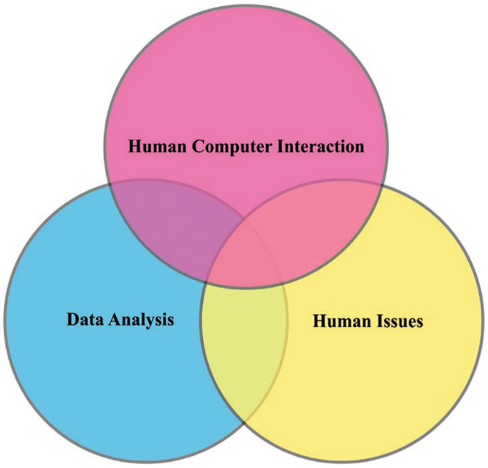

Figure 1: View of human-centered computing

In the current pandemic scenario, each country, together with India, is fighting against COVID-19 and yet seeking cost-effective and practical solutions to confront the problem that arises in various modes [6]. In the challenging present epidemic scenario, the several diseased persons are gradually increasing worldwide, and there is a massive requirement to utilize the well organized and adequate services provided by the IoT technique. Moreover, IoT has been previously applied for serving the purpose in distinct fields where the Internet of Medical Things (IoMT) or Internet of Healthcare Things (IoHT) are related to the current problems [7]. Currently, the more popular testing method for COVID-19 diagnoses is a real-time reverse transcription-polymerase chain reaction (RT-PCR).

Chest radiological imaging, namely X-ray and computed tomography (CT), have significant roles in earlier diagnoses and treatment [8]. Chest X-ray image has been exposed to be beneficial in monitoring the effect that COVID-19 causes on lung tissue. Accordingly, chest X-ray images might be utilized in the diagnosis of COVID-19 [9,10]. In the study, several deep-learning (DL)-based models use chest X-ray images to detect disease. Fig. 2 depicts the process of existing XAI model.

This article presents a new quantum-inspired differential evolution with explainable artificial intelligence based COVID-19 Detection and Classification (QIDEXAI-CDC) model for HIoT systems. The QIDEXAI-CDC model mainly uses bilateral filtering (BF) as a preprocessing tool to eradicate the noise. Moreover, RetinaNet is applied to generate useful feature vectors from radiological images. For COVID-19 detection and classification, a quantum-inspired differential evolution (QIDE) with kernel extreme learning machine (KELM) model is utilized. The QIDE algorithm helps appropriately choose the weight and bias values of the KELM model. In order to report the enhanced COVID-19 detection outcomes of the QIDEXAI-CDC model.

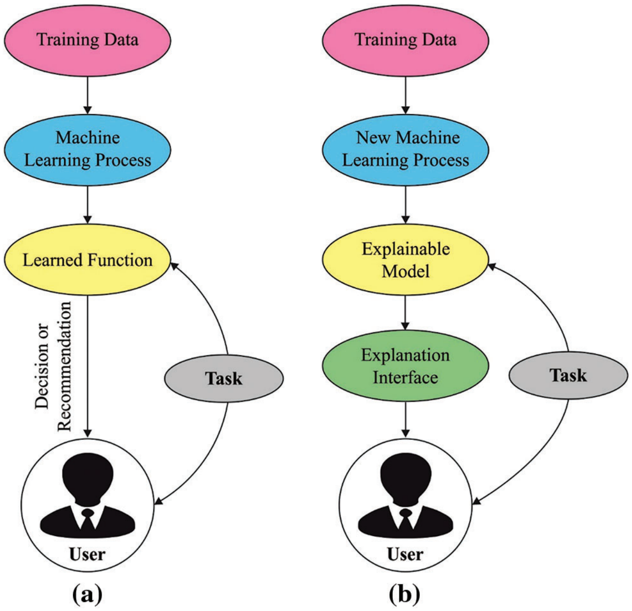

Figure 2: Difference between ai and xai: (a) existing model (b) xai model

2 Related Works

Ozturk et al. [11] presented an approach for automated COVID-19 diagnosis with raw chest X-ray images. The suggested method is technologically advanced for providing a precise diagnosis for the binary classifier (No-Findings *vs*. COVID) and multiclass classifier (Pneumonia *vs*. COVID *vs*. No-Findings.). The method presented a classification performance of 87.02% for multiclass cases and 98.08% for binary classes. The DarkNet method has been used to classify the YOLO real-time object recognition system. In [12], COVID-19 was identified by a DL, a class of artificial intelligence. The data class is reorganized with the Fuzzy Color method as a preprocessing phase, and the image that is organized as the original image is stacked. Next, the stacked data set was trained by the DL approaches (SqueezeNet, MobileNetV2), and the feature set attained by the algorithms has been treated by utilizing the Social Mimic optimization approach. Then, the useful feature is classified and combined with SVM.

In [13], a CNN method named CoroDet for automated diagnosis of COVID-19 by utilizing a CT scan and raw chest X-ray images was presented. Corot is designed to serve as a precise diagnosis method. In [14], a CNN method is presented by using end-to-end training. The experiments utilised a data set containing 180 COVID-19 and 200 normal (healthy) chest X-ray images. The deep feature extraction from the ResNet50 method and SVM classification using the linear kernel function is employed. In [15], an alternate diagnosis method for detecting COVID-19 cases with accessible resources and an innovative DL method is developed. The presented approach is performed in four stages: preprocessing, data augmentation, stage-I and -II deep network models. Narin et al. [16] proposed a five pretrained convolution neural network-based method (ResNet101, ResNet50, InceptionV3, Inception-ResNetV2 and ResNet152) for detecting coronavirus pneumonia-diseased person with chest X-ray radiographs. Through five-fold cross-validation, they have performed three distinct binary classifiers with 4 groups (normal (healthy), COVID-19, bacterial pneumonia, and viral pneumonia).

3 The Proposed QIDEXAI-CDC Model

In this study, a new QIDEXAI-CDC technique has been designed to identify the occurrence of COVID-19 using the XAI tools on HIoT systems. The proposed QIDEXAI-CDC model incorporates several stages of subprocesses, namely BF-based noise removal, RetinaNet-based feature extraction, KELM-based classification, and QIDE-based parameter tuning. The QIDE algorithm is applied to appropriately choose the weight and bias values of the KELM model, thereby enhancing the classifier efficiency. Fig. 3 demonstrates the block diagram of the QIDEXAI-CDC technique.

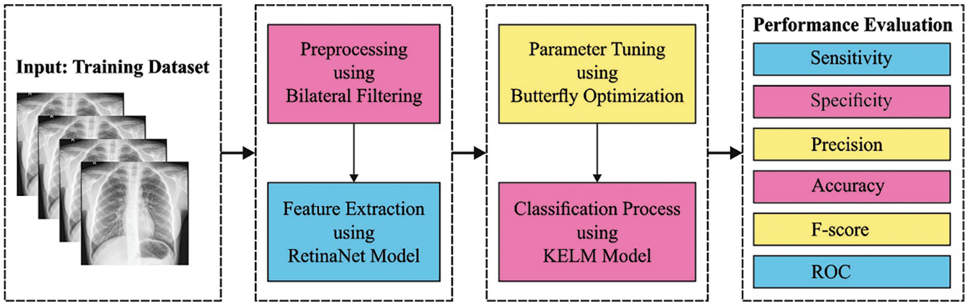
Input: Training Dataset
Preprocessing using Bilateral Filtering
Feature Extraction using RetinaNet Model
Parameter Tuning using Butterfly Optimization
Classification Process using KELM Model
Performance Evaluation
Sensitivity
Specificity
Precision
Accuracy
F-score
ROC

Figure 3: Block diagram of QIDEXAI-CDC technique

3.1 Data Pre-Processing

At the primary stage, the noise in the radiological images can be discarded by using the BF technique. Consider F denotes a multi-channel image, and W denotes a sliding window of finite size $n \times n$. Let the pixel in W be the Cartesian Coordinate and denoted as $\mathrm{i} = (i_1, \ i_2) \in Y^2$ the location of pixel $\mathrm{F_i}$ in W whereas $Y = \{0, \ 1, \ . \ . \ . \ , \ n-1\}$ is endowed with the common order. Based on [17], the BF replace the centralized pixel of filtering window through a weighting average of neighboring color pixel. The weight $\mathcal{W}(\mathrm{F_i}, \ \mathrm{F_j})$ equivalent to pixel $\mathrm{F_j}$ respect to $\mathrm{F_i}$ is the product of photometrical and spatial elements

$$\mathcal{W}(\mathrm{F_i}, \ \mathrm{F_j}) = \mathcal{W}_s(\mathrm{F_i} \ , \ \mathrm{F_j})\mathcal{W}_p(\mathrm{F_i}, \ \mathrm{F_j}) \tag{1}$$

Whereas the spatial element $\mathcal{W}_s \ (\mathrm{F_i} \ , \ \mathrm{F_j})$ is represented as

$$\mathcal{W}_s(F_\mathrm{i}, F_\mathrm{j}) = e^{-\frac{\|\mathrm{i}-\mathrm{j}\|_2^2}{2\sigma_S^2}} \tag{2}$$

and the photometrical element $\mathcal{W}_p(\mathrm{F_i}, \ \mathrm{F_j})$ is represented as

$$\mathcal{W}_p(\mathrm{F_i}, \ \mathrm{F_j}) = e^{-\frac{\Delta E_{Lab}(\mathrm{F_i},\mathrm{F_j})^2}{2\sigma_p^2}} \tag{3}$$

In which $\Delta E_{Lab} = \left[(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2\right]^{\frac{1}{2}}$ indicates the perceptual color error from the $L^*a^*b^*$ color space, and $\sigma_s, \ \ \sigma_p > 0$. The color vector output $\widetilde{Fi}$ of the filter is calculated by the standardized weight as given below.

$$\overline{\mathrm{F_i}} = \frac{\sum_{F_J \in w} \mathcal{W}(\mathrm{F_i}, \mathrm{F_j})\mathrm{F_j}}{\sum_{F_J \in w} \mathcal{W}(\mathrm{F_i}, \mathrm{F_j})} \tag{4}$$

The $\mathcal{W}_p$ weighted function reduced as perceptual color variance amongst the color vector increases, the $\mathcal{W}_s$ weighted function reduces as spatial distance in the images among I and j increases. The photometric component decreases the impact of pixel perceptually distinct concerning the one under processing while the spatial element reduces the impact of pixel-reducing blurring. In this method, the sharpness of edges is preserved, and a perceptually similar pixel area is averaged together. The σ_s and σ_p parameters are utilized for adjusting the impact of the photometric and the spatial elements, correspondingly. They are assumed as rough thresholds to identify pixels similar or close to the centralized one sufficiently. Noted that if $\sigma_p \rightarrow \infty$, the BF approach a Gaussian filter and if $\sigma_s \rightarrow \infty$, the filter approaches a range filter without spatial concept. In such cases, if $\sigma_p \rightarrow \infty$ and $\sigma_s \rightarrow \infty$, the BF act as AMF.

3.2 RetinaNet Based Feature Extraction

Next to data preprocessing, the RetinaNet is applied to produce a collection of feature vectors. It comprises three subnetworks, namely residual network (ResNet), feature pyramid network (FPN), and two fully convolution networks (FCN) [18]. The detailed work of each component is given in the following.

ResNet: The major intention of the ResNet involves the concept of residual learning that enables the actual input data to be straightway fed into the next layer. It utilizes several network layers. A few common versions of ResNet are 50 layers, 101 layers, and 152 layers. Here, ResNet 101 model is applied and the features derived are forwarded to the succeeding network.

FPN: It is used for effectually deriving the features of all dimensions in the image using the traditional CNN approach. Initially, a 1-D image is fed as input to the ResNet. Followed by the succeeding layer of the convolution network, the features of every layer are chosen by the FPN and then integrated to produce the end feature output combination.

FCN: The class subnet in the FCN performs the classifier process. It can identify the class of radiological images. It aims to identify the presence of COVID-19 in the radiological images and record the positions.

Focal loss: It is an extended kind of cross-entropy loss, and the binary cross-entropy is mathematically formulated as follows:

$$CE(p,\ y) = \begin{cases} -\log(p), & \mathit{if}\ y = 1, \\ -\log(1-p), & \mathit{otherwise}, \end{cases} \tag{5}$$

where $y \in [\pm,\ 1]$ indicates the ground truth class and $p \in [0, 1]$ indicates the estimated possibility of the class label
$y = 1$.

$$p_t = \begin{cases} p, & \mathit{if}\ y = 1, \\ 1-p, & \mathit{otherwise}, \end{cases} \tag{6}$$

$$CE\ (p,\ y) = CE(p_t) = -\log\ (p_t) \tag{7}$$

To resolve the data imbalance issue among the COVID-19 and Non-COVID 19 instances, the actual form can be transformed.

$$CE(p_t) = -\alpha_t \log\ (p_t) \tag{8}$$

Among them,

$$\alpha_t = \begin{cases} \alpha, & \mathit{if}\ y = 1 \\ l-\alpha, & \mathit{otherwise} \end{cases} \tag{9}$$

where $\alpha \in [0, 1]$ indicates the weight factor. For resolving the issue of complicated samples, the focus variable C is applied to attain the end form of the focal loss:

$$FL\ (p_t) = -\alpha_t(1-p_t)^{\gamma}\ \log\ (p_t) \tag{10}$$

3.3 Image Classification Using the KELM model

The generated features from the previous step are fed into the KELM model to detect and classify COVID-19. In recent times, the ELM learning approach using better generalization efficiency and fast learning speed was receiving considerable interest from a researcher [19]. In ELM, the first parameter of hidden state does not have to be tuned and most of the non-linear piecewise continuous function is utilized as the hidden neuron. Thus, for N arbitrary sample $\{(x_i,\ t_i)|x_i \in R^n, t_i \in R^m, i = 1, \ldots, \text{N}\}$, the output function in ELM using L hidden neuron is

$$f_L(x) = \sum\nolimits_{i=1}^{L} \beta_i h_i(x) = h(x)\beta \tag{11}$$

Whereas $\beta = [\beta_1,\ \beta_2,\ \beta_L]$ denotes the vector of output weights among the hidden state of L neuron and output neurons and $h(x) = [h_1(\text{x}),\ h_2(x),\ \cdots, h_L(\text{x})]$ represent the output vector of hidden state regarding the input x, that maps data from input space to the ELM feature space [19]. For improving the generalization performance and minimizing the training error of neural network, the output weights and training error must be minimalized simultaneously, as follows

$$Minimalize : \|H\beta - T\|, \|\beta\| \tag{12}$$

The least-square solution of Eq. (12) depends on KKT condition as follows

$$\beta = H^T\left(\frac{1}{C} + HH^T\right)^{-1} T \tag{13}$$

In which H denotes the hidden state output matrix, C indicates the regulation coefficient, and T denotes the expected output matrix of instance. Next, the output function of ELM can be formulated as follow

$$f(x) = h(\text{x})H^T\left(\frac{1}{C} + HH^T\right)^{-1} T \tag{14}$$

Once the feature mapping $h(?)$ is not known and the kernel matrix of ELM depends on Mercer condition is described by:

$$M = HH^T : m_{ij} = h(x_i)h(x_j) = k(x_i,\ x_j) \tag{15}$$

Therefore, the output function $f(x)$ of the KELM is formulated by

$$f(x) = [k(\text{x}, \text{x}_1),\ \cdots, k(x, \text{x}_N)]\left(\frac{1}{C} + M\right)^{-1} T \tag{16}$$

Whereas $M = HH^T$ and $k(x,y)$ indicates the kernel function of hidden neuron of SLFN. There are several kernel functions fulfilling the Mercer condition presented from the current study including polynomial kernel, linear kernel, exponential kernel, and Gaussian kernel. Here, we employ three standard kernel functions for performance and simulation study and the selected kernel function is given by.

(1) Gaussian kernel:

$$k(x,y) = \exp\ (-a\|x-y\|) \tag{17}$$

(2) hyperbolic tangent (sigmoid) kernel:

$$k(x, y) = \tanh\left(b\mathrm{x}^T y + c\right) \tag{18}$$

(3) wavelet kernel:

$$k(x, y) = \cos\left(d\frac{\|x - y\|}{e}\right)\exp\left(-\frac{\|x - y\|^2}{f}\right) \tag{19}$$

Whereas tangent kernel function is a standard global nuclear function and Gaussian kernel function is a standard local kernel function, correspondingly [8]. Additionally, the complicated wavelet kernel function is utilized to test the efficiency of the algorithm.

3.4 Parameter Tuning Using QIDE Algorithm

At the final stage, the weight and bias values of the KELM model can be effectually adjusted by the use of QIDE algorithm. The QIDE algorithm is derived from the concepts of quantum computing into the DE. It makes use of quantum features for enhancing the outcomes of the DE. Here, the *Q*-bits can be employed as fundamental units. It performs encoding and quantum chromosomes will produce quantum population. Based on the modification of the quantum population, the optimum solution of the objective function can be defined [20]. The development of quantum population involves the succeeding features.

At QC, the data will be stored and analyzed in the *Q*-bit way. It can be defined in Eq. (20):

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{20}$$

where $|\alpha|^2 + |\beta|^2 = 1$, α and β represents the likelihood amplitude of qubit as 0 and 1 correspondingly, $|\alpha|^2$ and $|\beta|^2$ represents the likelihood of qubit as 0 and 1, correspondingly.

The individual of length *m* is defined in Eq. (10).

$$x_i = \left[\begin{array}{c|c|c|c|} \alpha_1 & \alpha_2 & \ldots & \alpha_m \\ \beta_1 & \beta_2 & \ldots & \beta_{\mathrm{m}} \end{array}\right] = \left(\left|\begin{array}{c} \cos\left(\theta_{i,1}\right) \\ \sin\left(\theta_{i,1}\right) \end{array}\right|\left|\begin{array}{c} \cos\left(\theta_{i,2}\right) \\ \sin\left(\theta_{i,2}\right) \end{array}\right| \begin{array}{c} \wedge \\ \wedge \end{array} \left|\begin{array}{c} \cos\left(\theta_{i,\mathrm{m}}\right) \\ \sin\left(\theta_{i,\mathrm{m}}\right) \end{array}\right|\right) \tag{21}$$

$$\theta_{ij} = \pi \text{ rand} \tag{22}$$

For computing the objective function of the QIDE algorithm, it is essential to undergo mapping of the quantum chromosome to solution space $\Omega_{QDE} = [a,\ b]$. The solution space conversion can take place using Eq. (23).

$$q_{i,j} = \frac{(b - a)q_{i,j} + (b + a)}{2} = \frac{\begin{bmatrix} (b - a)\cos\left(\theta_{i,j}\right) + (b + a) \\ (b - a)\sin\left(\theta_{i,j}\right) + (b + a) \end{bmatrix}}{2} \tag{23}$$

Quantum chromosome indicates a superposition of a pair of chromosomes. If the quantum string gets noticed, then the chromosome gets collapsed from quantum to unique states. The end state of the chromosome can be produced by the linear superposition of the likelihood amplitude of every qubit. The quantum observation can be mathematically defined using Eq. (24):

$$q_{i,j} = \begin{cases} \cos\left(\theta_{i,j}\right), & if\ \ \cos\left(\theta_{i,j}\right)^2 < rand \\ \sin\left(\theta_{i,j}\right), & else \end{cases} \tag{24}$$

A major principle of QC is to create the ground state which comprises the superposition state interferes via the quantum gate and convert the related phase amongst quanta. It enables the model to balance among global as well as local searching processes, and thereby convergence rate gets improved. At the QIDE, with respect to Q-bit, the evolution procedure comprises quantum mutation, crossover, and selection operations [20].

Step 1. Quantum mutation operation

At the QIDE, the mutation mechanism of the DE is appropriate. With respect to Q-bit, the $DE/rand/1$ scheme is considered as an instance, as given below.

$$\theta_i^t = \theta_{r1}^t + F\ rand\left(\theta_{r2}^t - \theta_{r3}^t\right) \tag{25}$$

Step 2. Quantum crossover operation

For ensuring population diversity, the new individuals are produced with a specific possibility among the test vectors and parent individuals.

$$\theta_{i,j}^{t'} = \begin{cases} \theta_{i,j}^{t'}, & if\ rand\ < CR\ or\ j = rand \\ \theta_{i,jnew}^{t}, & else \end{cases} \tag{26}$$

Step 3. Quantum selection operation

For choosing the optimum individual for the succeeding round, the fitness value of the objective function for every test vector is passed via greedy strategy. The quantum selection function can be defined using Eq. (27).

$$u_i^k = \begin{cases} \theta_i^t, & if\ f\left(x_i^t\right) < f\left(x_i^{t'}\right) \\ \theta_{i,j}^{t'}, & else \end{cases} \tag{27}$$

The QIDE approach derives a fitness function to obtain enhanced classification efficiency. It describes a positive integer to characterize the good performance of the candidate solution. Here, the minimization of the classification error rate is regarded as the fitness function. The worse solution achieves an increased error rate and the optimum solution has a minimal error rate.

$$fitness(x_i) = Classifier\ Error\ Rate(x_i) = \frac{number\ of\ misclassified\ instances}{Total\ number\ of\ instances} * 100 \tag{28}$$

4 Experimental Validation

The performance validation of the QIDEXAI-CDC model is performed using the benchmark COVID-19 database from Kaggle repository (available at https://www.kaggle.com/tawsifurrahman/covid19-radiography-database). The dataset holds 3616 COVID images, 6012 Lung Opacity images, 10192 Normal images, and 1312 Viral Pneumonia images. In this work, 3500 images under each class are taken. A few sample images are illustrated in Fig. 4.

Fig. 5 reports the confusion matrices produced by the QIDEXAI-CDC model on the classification of distinct training/testing data. The figure depicted that the QIDEXAI-CDC model has accomplished maximum COVID-19 detection outcomes. For instance, with training/testing data of 90:10%, the QIDEXAI-CDC model has recognized 346 instances into COVID and 347 instances into Non-COVID. Besides, with training/testing data of 60:40%, the QIDEXAI-CDC method has recognized 1335 instances into COVID and 1357 instances into Non-COVID.

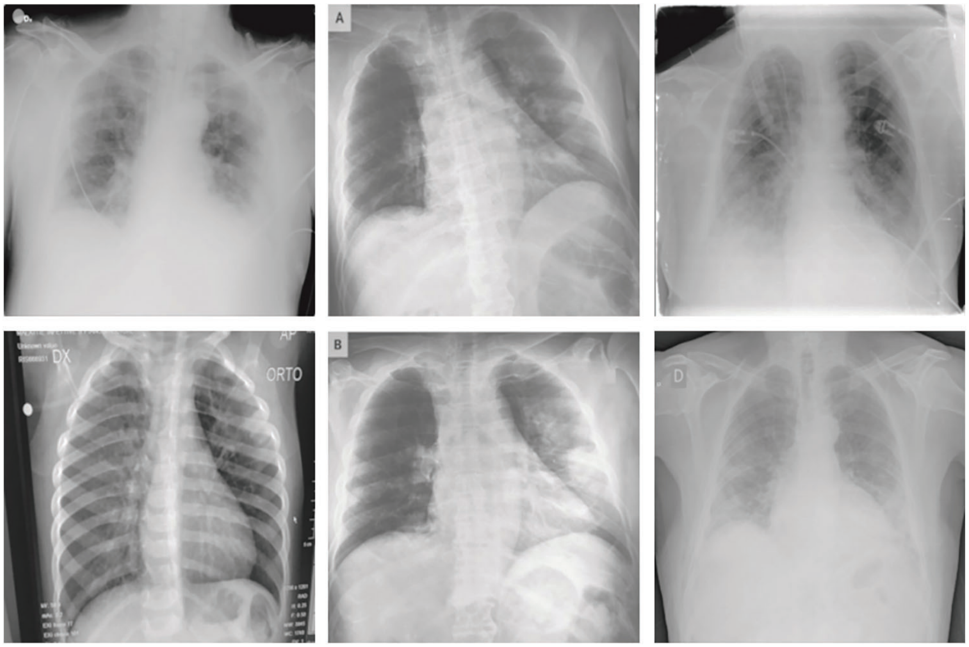

Figure 4: Sample images

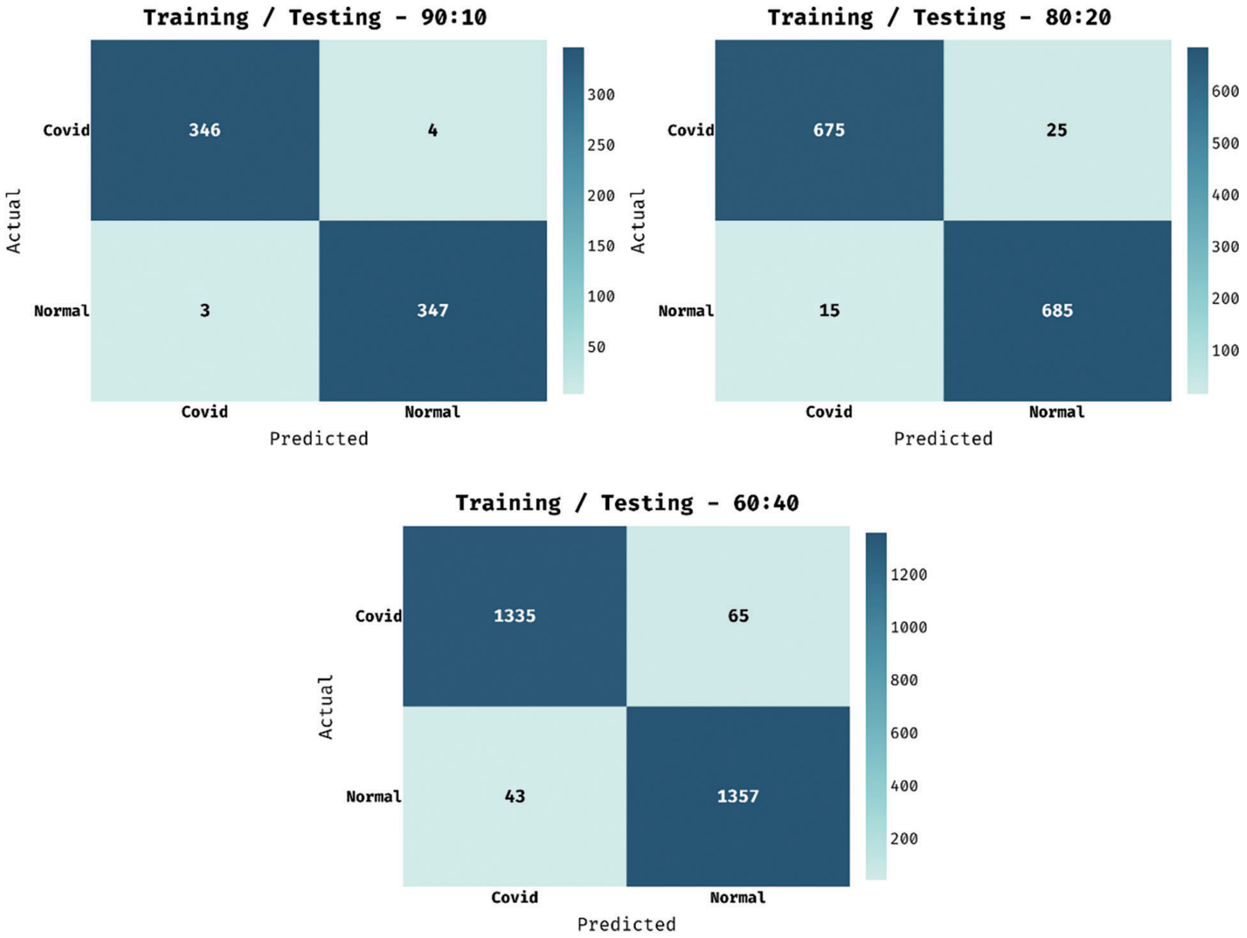

Figure 5: Confusion matrix of QIDEXAI-CDC technique distinct training/testing data

Table 1 exhibits the overall classifier results of the QIDEXAI-CDC model on COVID-19 detection and classification under different training/testing data. Fig. 6 demonstrates the classification outcome of the QIDEXAI-CDC model on various sizes of training/testing data. The figure reported that the QIDEXAI-CDC model has obtained effectual outcomes on all training/testing data. For instance, with training/testing data of 90:10%, the QIDEXAI-CDC model has resulted to $sens_y$ of 98.86%, $spec_y$ of 99.14%, and $prec_n$ of 99.14%. Moreover, with training/testing data of 80:20%, the QIDEXAI-CDC technique has resulted to $sens_y$ of 96.43%, $spec_y$ of 97.86%, and $prec_n$ of 97.83%. Furthermore, with training/testing data of 60:40%, the QIDEXAI-CDC approach has resulted to $sens_y$ of 95.36%, $spec_y$ of 96.93%, and $prec_n$ of 96.88%.

Table 1: Result analysis of QIDEXAI-CDC technique under different training/testing data

Measures	Sensitivity	Specificity	Precision	Accuracy	F-score
Training/Testing (90:10)	98.86	99.14	99.14	99.00	99.00
Training/Testing (80:20)	96.43	97.86	97.83	97.14	97.12
Training/Testing (60:40)	95.36	96.93	96.88	96.14	96.11
Average	96.88	97.98	97.95	97.43	97.41

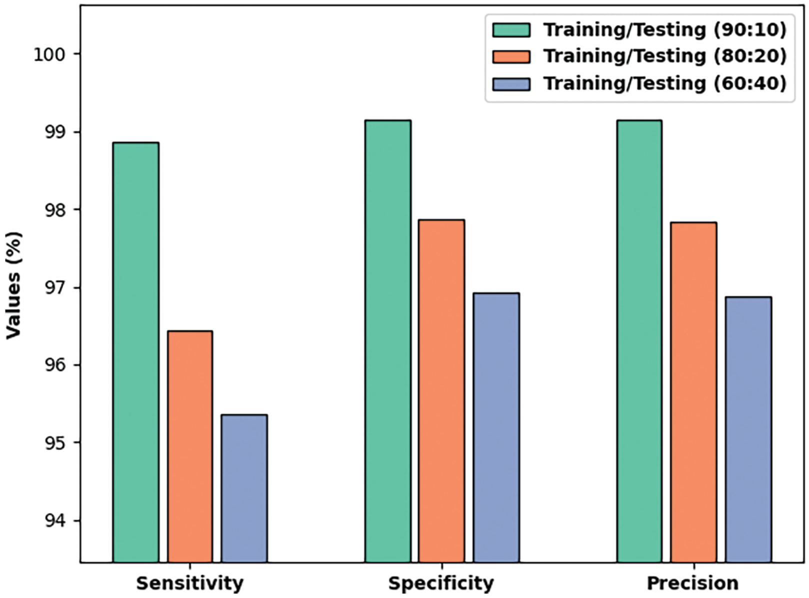

Figure 6: Result analysis of QIDEXAI-CDC technique under different training/testing data

Fig. 7 portrays the classification outcome of the QIDEXAI-CDC technique on various sizes of training/testing data. The figure exposed that the QIDEXAI-CDC method has reached effectual outcome on all training/testing data. For instance, with training/testing data of 90:10%, the QIDEXAI-CDC methodology has resulted to acc_y of 99% and F_{score} of 99%. Also, with training/testing data of 80:20%, the QIDEXAI-CDC approach has resulted to acc_y of 97.14% and F_{score} of 97.12%. At last, with training/testing data of 60:40%, the QIDEXAI-CDC method has resulted to acc_y of 96.14% and F_{score} of 96.11%.

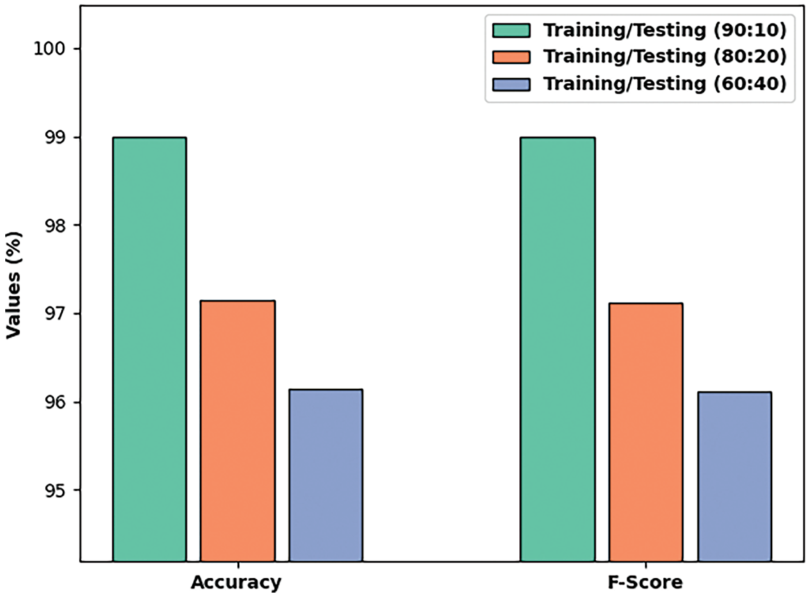

Figure 7: Acc_y and F_{score} analysis of QIDEXAI-CDC technique under different training/testing data

Fig. 8 illustrates the ROC analysis of the QIDEXAI-CDC methodology on the test dataset. The figure exposed that the QIDEXAI-CDC approach has reached increased outcomes with a higher ROC of 97.9295.

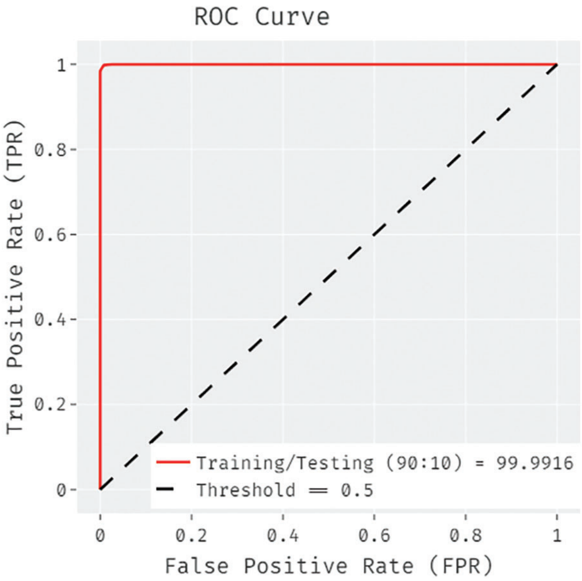

Figure 8: ROC analysis of QIDEXAI-CDC technique

Fig. 9 offers the accuracy and loss graph analysis of the QIDEXAI-CDC technique under three training/testing datasets. The results outperformed that the accuracy value tends to increase and loss value tends to decrease with an increase in epoch count. It can be also observed that the training loss is low and validation accuracy is high under three training/testing datasets.

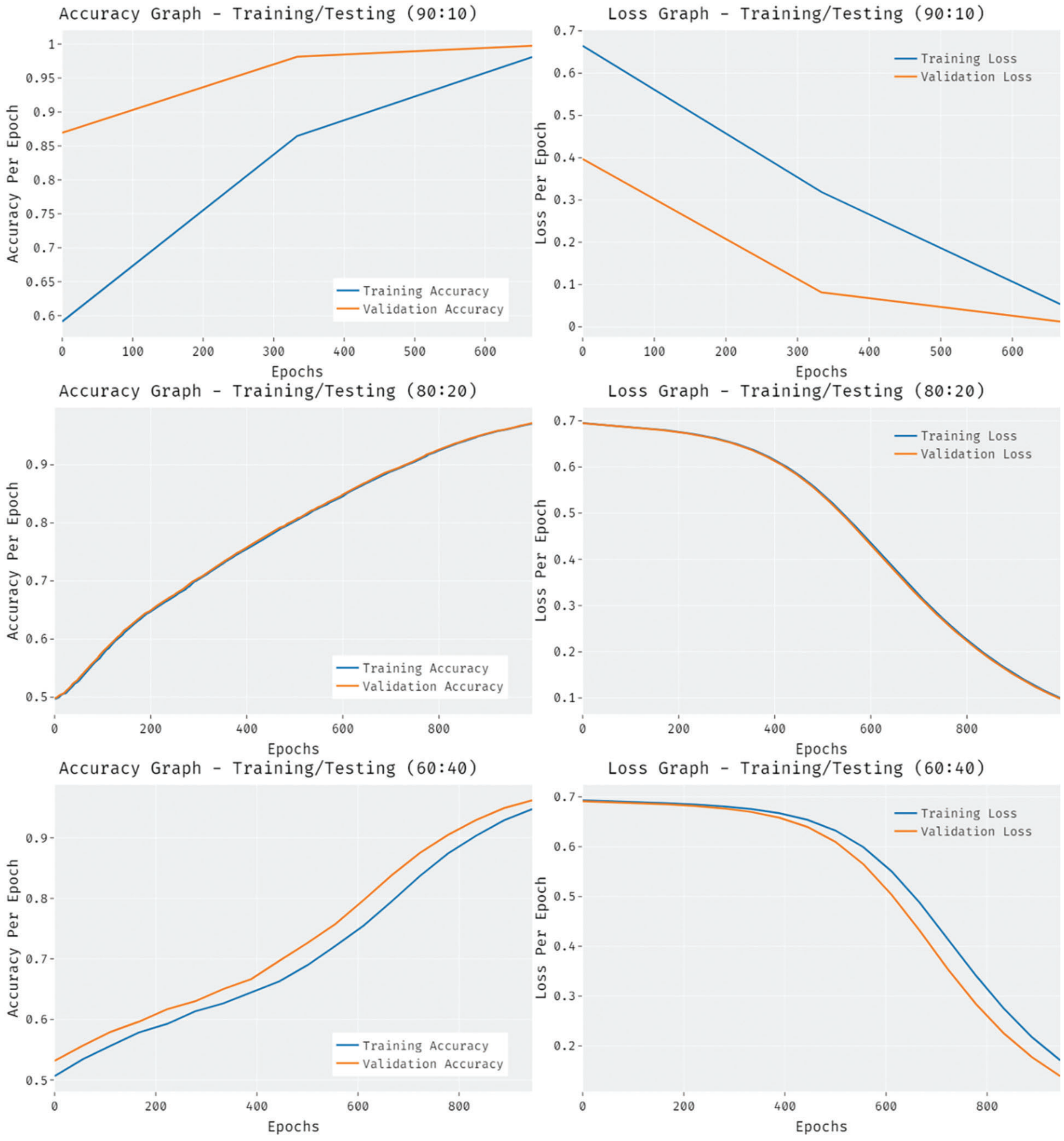

Figure 9: Accuracy and loss analysis of QIDEXAI-CDC technique under different training/testing data

Table 2 demonstrates the comparative analysis of QIDEXAI-CDC technique interms of various measures with recent methods. Fig. 10 demonstrates a comparative $sens_y$ and $spec_y$ analysis of the QIDEXAI-CDC technique with existing approaches [21]. The figure portrayed that the Inception v3 and NFNET models have obtained least performance with the minimal values of $sens_y$ and $spec_y$. Likewise, the VGG16 and ResNet101 techniques have reached slightly enhanced values of $sens_y$ and $spec_y$. Next, the Modified MobileNetv2 and MobileNetv2 systems have showcased reasonable values of $sens_y$ and *spec*$_y$. Eventually, the QIDEXAI-CDC algorithm has outperformed the other methods with the higher *sens*$_y$ and *spec*$_y$ of 98.86% and 99.14%.

Table 2: Comparative analysis of QIDEXAI-CDC technique with recent approaches

Methods	Sensitivity	Specificity	Precision	Accuracy	F-score
QIDEXAI-CDC	98.86	99.14	99.14	99.00	99.00
Modified MobileNetV2	98.54	97.14	97.25	98.95	97.33
MobileNetV2	97.76	96.24	96.91	97.53	96.45
VGG16	95.42	93.68	94.05	95.41	94.19
ResNet101	96.07	93.33	93.78	95.73	94.94
InceptionV3	95.37	92.80	92.66	94.02	93.51
NFNET	94.58	93.05	94.47	94.74	94.36

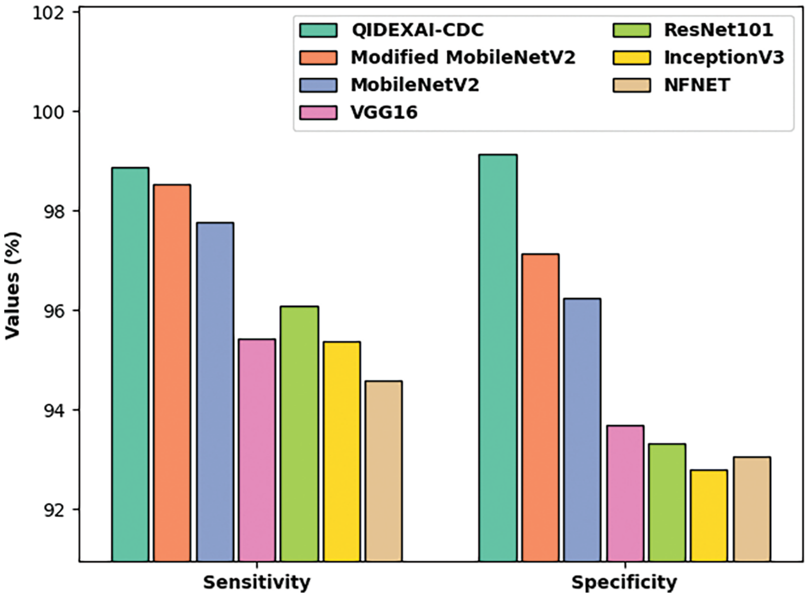

Figure 10: *Sens*$_y$ and *Spec*$_y$ analysis of QIDEXAI-CDC technique with recent algorithms

Fig. 11 provides a comparative *prec*$_n$ and F_{score} analysis of the QIDEXAI-CDC algorithm with existing techniques [21]. The figure revealed that the Inception v3 and NFNET techniques have reached least performance with the lower values of *prec*$_n$ and F_{score}. In addition, the VGG16 and ResNet101 techniques have gained slightly enhanced values of *prec*$_n$ and F_{score}. Similarly, the Modified MobileNetv2 and MobileNetv2 techniques have illustrated reasonable values of *prec*$_n$ and F_{score}. Finally, the QIDEXAI-CDC model has outperformed the other methods with the increased *prec*$_n$ and F_{score} of 99.14% and 99%.

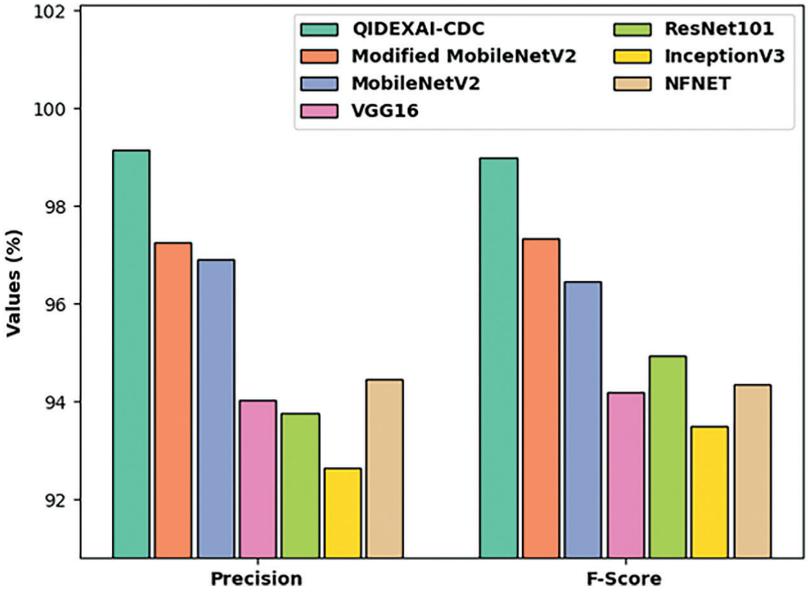

Figure 11: *$Prec_n$* and *F_{score}* analysis of QIDEXAI-CDC technique with recent algorithms

Fig. 12 offers a comparative $accu_y$ analysis of the QIDEXAI-CDC model with existing techniques [21]. The figure shows that the Inception v3 and NFNET models have obtained least performance with the minimal values of $accu_y$. In line with, the VGG16 and ResNet101 models have reached slightly enhanced values of $accu_y$. Followed by, the Modified MobileNetv2 and MobileNetv2 models have showcased reasonable values of $accu_y$. However, the QIDEXAI-CDC model has outperformed the other methods with the maximum $accu_y$ of 94.71%.

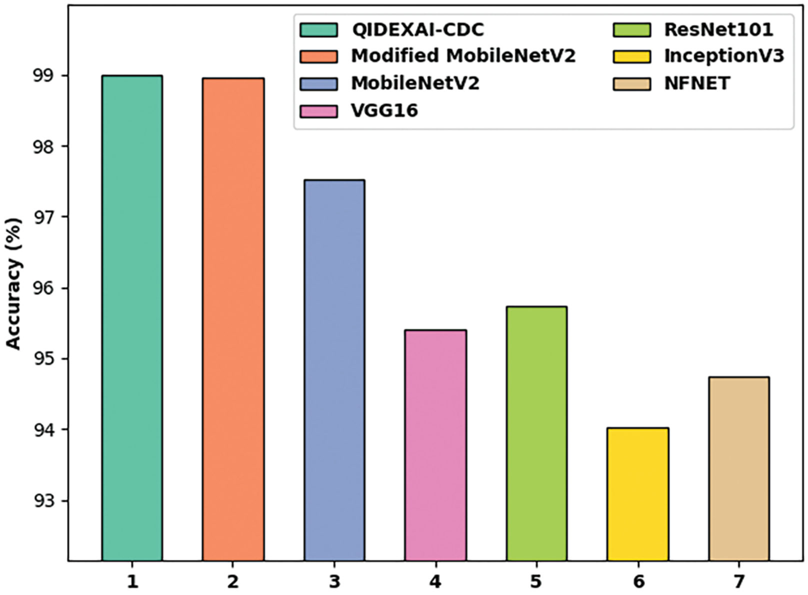

Figure 12: *Acc_y* analysis of QIDEXAI-CDC technique with recent algorithms

From the above results and discussion, it is evident that the QIDEXAI-CDC model has the ability to outperform the other methods on COVID-19 detection and classification.

5 Conclusion

In this study, a new QIDEXAI-CDC technique has been designed to identify the occurrence of COVID-19 using the XAI tools on HIoT systems. The proposed QIDEXAI-CDC model incorporates several stages of subprocesses namely BF based noise removal, RetinaNet based feature extraction, KELM based classification, and QIDE based parameter tuning. The QIDE algorithm is applied to appropriately choose the weight and bias values of the KELM model, thereby enhancing the classifier efficiency. Extensive comparative studies reported the supremacy of the QIDEXAI-CDC model over the recent approaches. Therefore, the QIDEXAI-CDC model has been utilized as a protocol tool to detect and classify COVID-19 in the HIoT system. In the future, the QIDEXAI-CDC model can be tested using large scale benchmark datasets.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] A. Nieto and R. Rios, “Cybersecurity profiles based on human-centric IoT devices,” *Human-centric Computing and Information Sciences*, vol. 9, no. 1, pp. 39, 2019.

[2] O. G. Carmona, D. C. Mansilla, F. A. Kraemer, D. L. de-Ipiña and J. G. Zubia, “Exploring the computational cost of machine learning at the edge for human-centric Internet of Things,” *Future Generation Computer Systems*, vol. 112, no. 6, pp. 670–683, 2020.

[3] I. G. Marino, R. Muttukrishnan and J. Lloret, “Human-centric ai for trustworthy IoT systems with explainable multilayer perceptrons,” *IEEE Access*, vol. 7, pp. 125562–125574, 2019.

[4] D. C. Mansilla, I. Moschos, O. K. Esteban, A. C. Tsolakis, C. E. Borges *et al.,* “A human-centric & context-aware IoT framework for enhancing energy efficiency in buildings of public use,” *IEEE Access*, vol. 6, pp. 31444–31456, 2018.

[5] D. Milovanovic, V. Pantovic, N. Bojkovic and Z. Bojkovic, “Advanced human-centric 5g-iot in a smart city: Requirements and challenges,” in *Int. Conf. on Human-Centred computing*, Čačak, Serbia, pp. 285–296, 2019.

[6] A. Tahamtan and A. Ardebili, “Real-time RT-PCR in COVID-19 detection: Issues affecting the results,” *Expert Review of Molecular Diagnostics*, vol. 20, no. 5, pp. 453–454, 2020.

[7] T. Ji, Z. Liu, G. Wang, X. Guo, C. Lai *et al.,* “Detection of COVID-19: A review of the current literature and future perspectives,” *Biosensors and Bioelectronics*, vol. 166, pp. 112455, 2020.

[8] B. Udugama, P. Kadhiresan, H. N. Kozlowski, A. Malekjahani, M. Osborne *et al.,* “Diagnosing COVID-19: The disease and tools for detection,” *ACS Nano*, vol. 14, no. 4, pp. 3822–3835, 2020.

[9] O. S. Albahri, A. A. Zaidan, A. S. Albahri, B. B. Zaidan, K. H. Abdulkareem *et al.,* “Systematic review of artificial intelligence techniques in the detection and classification of COVID-19 medical images in terms of evaluation and benchmarking: Taxonomy analysis, challenges, future solutions and methodological aspects,” *Journal of Infection and Public Health*, vol. 13, no. 10, pp. 1381–1396, 2020.

[10] T. S. Santosh, R. Parmar, H. Anand, K. Srikanth and M. Saritha, “A review of salivary diagnostics and its potential implication in detection of COVID-19,” *Cureus*, 2020. https://doi.org/10.7759/cureus.7708.

[11] T. Ozturk, M. Talo, E. A. Yildirim, U. B. Baloglu, O. Yildirim *et al.,* “Automated detection of COVID-19 cases using deep neural networks with X-ray images,” *Computers in Biology and Medicine*, vol. 121, pp. 103792, 2020.

[12] M. Toğaçar, B. Ergen and Z. Cömert, "COVID-19 detection using deep learning models to exploit Social Mimic Optimization and structured chest X-ray images using fuzzy color and stacking approaches," *Computers in Biology and Medicine*, vol. 121, pp. 103805, 2020.

[13] E. Hussain, M. Hasan, M. A. Rahman, I. Lee, T. Tamanna *et al.,* "CoroDet: A deep learning based classification for COVID-19 detection using chest X-ray images," *Chaos, Solitons & Fractals*, vol. 142, no. 7798, pp. 110495, 2021.

[14] A. M. Ismael and A. Şengür, "Deep learning approaches for COVID-19 detection based on chest X-ray images," *Expert Systems with Applications*, vol. 164, no. 4, pp. 114054, 2021.

[15] G. Jain, D. Mittal, D. Thakur and M. K. Mittal, "A deep learning approach to detect COVID-19 coronavirus with X-Ray images," *Biocybernetics and Biomedical Engineering*, vol. 40, no. 4, pp. 1391–1405, 2020.

[16] A. Narin, C. Kaya and Z. Pamuk, "Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks," *Pattern Analysis and Applications*, vol. 24, no. 3, pp. 1207–1220, 2021.

[17] Y. He, Y. Zheng, Y. Zhao, Y. Ren, J. Lian *et al.,* "Retinal image denoising via bilateral filter with a spatial kernel of optimally oriented line spread function," *Computational and Mathematical Methods in Medicine*, vol. 2017, no. 3, article 67, pp. 1–13, 2017.

[18] Y. Wang, C. Wang, H. Zhang, Y. Dong and S. Wei, "Automatic ship detection based on retinanet using multi-resolution gaofen-3 imagery," *Remote Sensing*, vol. 11, no. 5, pp. 531, 2019.

[19] V. P. Vishwakarma and S. Dalal, "A novel approach for compensation of light variation effects with KELM classification for efficient face recognition," in *Advances in VLSI, Communication, and Signal Processing*, Lecture Notes in Electrical Engineering Book Series, vol. 587. Singapore: Springer, pp. 1003–1012, 2020.

[20] W. Deng, S. Shang, X. Cai, H. Zhao, Y. Zhou *et al.,* "Quantum differential evolution with cooperative coevolution framework and hybrid mutation strategy for large scale optimization," *Knowledge-Based Systems*, vol. 224, no. 1, pp. 107080, 2021.

[21] S. Akter, F. M. J. M. Shamrat, S. Chakraborty, A. Karim and S. Azam, "COVID-19 detection using deep learning algorithm on chest X-ray images," *Biology*, vol. 10, no. 11, pp. 1174, 2021.