Computers, Materials & Continua
DOI:10.32604/cmc.2021.015541
Article

Tech Science Press

Classification of COVID-19 CT Scans via Extreme Learning Machine

Muhammad Attique Khan[1], Abdul Majid[1], Tallha Akram[2], Nazar Hussain[1], Yunyoung Nam[3,*], Seifedine Kadry[4], Shui-Hua Wang[5] and Majed Alhaisoni[6]

[1]Department of Computer Science, HITEC University, Taxila, 47040, Pakistan
[2]Department of Electrical Engineering, COMSATS University Islamabad, Wah Campus, Wah Cantt, Pakistan
[3]Department of Computer Science and Engineering, Soonchunhyang University, Asan, Korea
[4]Department of Mathematics and Computer Science, Faculty of Science, Beirut Arab University, Lebanon
[5]Department of Mathematics, University of Leicester, Leicester, UK
[6]College of Computer Science and Engineering, University of Ha'il, Ha'il, Saudi Arabia
[*]Corresponding Author: Yunyoung Nam. Email: ynam@sch.ac.kr
Received: 26 November 2020; Accepted: 23 January 2021

Abstract: Here, we use multi-type feature fusion and selection to predict COVID-19 infections on chest computed tomography (CT) scans. The scheme operates in four steps. Initially, we prepared a database containing COVID-19 pneumonia and normal CT scans. These images were retrieved from the Radiopaedia COVID-19 website. The images were divided into training and test sets in a ratio of 70:30. Then, multiple features were extracted from the training data. We used canonical correlation analysis to fuse the features into single vectors; this enhanced the predictive capacity. We next implemented a genetic algorithm (GA) in which an Extreme Learning Machine (ELM) served to assess GA fitness. Based on the ELM losses, the most discriminatory features were selected and saved as an ELM Model. Test images were sent to the model, and the best-selected features compared to those of the trained model to allow final predictions. Validation employed the collected chest CT scans. The best predictive accuracy of the ELM classifier was 93.9%; the scheme was effective.

Keywords: Coronavirus; classical features; feature fusion; feature optimization; prediction

1 Introduction

The novel coronavirus pandemic disease that appeared in China has rapidly spread worldwide [1]. The World Health Organization (WHO) termed the disease caused by the virus COVID-19 on February 1, 2020 [2]. COVID-19 spread from Wuhan, China, to become a major global health problem [3]. The WHO has recorded 86,806 confirmed cases of COVID-19 in China, and 4634 deaths to date (16th December 2020) [4]. Many members of the coronavirus family cause disease. The virus that causes COVID-19 is termed the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) [5]. The common symptoms of COVID-19 are fever and cough, and sometimes headache and fatigue [6]. The COVID-19 virus was first discovered in humans in 2019 and spread rapidly in respiratory droplets [7].

To date (19th December 2020), there are 76,186,444 confirmed COVID-19 cases worldwide, with 1,684,864 deaths, according to WHO. It shows that the global mortality rate is 6.9%. The USA's confirmed COVID-19 cases are 17,899,267 and 321,025 deaths, which is top of the list in the world. Confirmed COVID-19 cases in India are 10,013,478, Brazil are 7,163,912, Russia are 2,819,429, France are 2,442,990, Turkey are 1,982,090, and UK are 1,977,167, respectively. In these countries, the number of deaths are 145,298, 185,687, 50,347, 60,229, 17,610, and 66,541, respectively to date (19th December 2020). And these cases are increasing day by day. Italy is another highly affecting country by this virus, and the positive reported cases are 1,921,778, and total deaths are 67,894. The Asian countries such as India is on the top, which is highly affected by this virus. In Pakistan, this rate is much slower as compared to other Asian Countries.

COVID-19 poses a major healthcare problem worldwide [8,9]. Early detection of infected patients would be helpful [10]. Many computer-aided diagnostic (CAD) systems allow physicians to recognize the stomach [11], lungs [12], and brain cancer [13], and COVID [14] infections. The existing COVID detection methods are slow and expensive. It is possible to use machine-learning algorithms to identify diseased lungs on computed tomography (CT) images. Most such algorithms feature supervised machine-learning. CAD systems automatically extract textural [15], geometric [16], and deep [17] features of chest CT images. However, not all extracted features are useful; higher-dimensionality data render diagnoses slow and compromise the results. Many medical researchers seek to develop feature selection methods [18] that transform high-dimensional features into simpler components. The algorithms select robust features and remove irrelevant information [19]. A robust feature set enhances CAD system performance in terms of both accuracy and time. The many methods of optimal feature selection include entropy-based approaches [20], Particle Swarm Optimization (PSO), and the so-called Grasshopper, and genetic algorithms (GAs) [21], to name a few [22]. In this work, we presented an automated technique for COVID19 classification using CT images. Our major contributions in this work are in the following steps: (i) Multi properties features are extracted in different directions from the CT images; (ii) Employed a parallel canonical correlation analysis approach for features fusion; (iii) GA is implemented for the best features selection, and (iv) ELM based classify the selected features.

The rest of the manuscript is organized as follows: The existing relevant studies are discussed in Section 2 (related work). The proposed methodology is described in Section 3, which includes dataset preparation, features fusion, and selection. Section 4 represents the experimental results, and finally, analysis and conclusion are presented in Section 5.

2 Related Work

Recently, COVID-19 patients have been diagnosed by reference to their X-ray and CT images using computer vision (CV)-based machine-learning algorithms, principally, supervised learning and deep learning (DL) techniques. Apostolopoulos et al. [23] subjected X-ray image datasets to transfer learning using different convolutional neural network (CNN) models and evaluated their technique with the aid of two datasets that contained confirmed COVID-19, bacterial pneumonia, and normal cases. The maximum accuracies achieved by MobileNet V2 [24] were 96.78% (two-class) and 94.72% (three-class). Li et al. [25] developed a novel DL model (COVNet) to detect COVID-19 in chest CT images; the sensitivity was 90% and the specificity 96%. Shan et al. [26] implemented a neural network-based VB-Net model for segmentation of COVID-19-infected regions in CT images. The Dice similarity was 91.6 ± 10.0%. Tsiknakis et al. [27] explored the uncertainties of DL models used to detect COVID-19 in chest X-ray images. Uncertainty was estimated via transfer learning using a Bayesian DL classifier. Narin et al. [28] used three CNN models (ResNet50, Inception-ResNetV2, and InceptionV3) to identify COVID-19 patients employing chest X-ray images. ResNet50 performed best (98% classification accuracy; the figures for InceptionV3 and Inception-ResNetV2 were 97% and 87% respectively). In [29], a DL based COVIDX-Net was used for automatic detection of COVID-19 in X-ray images. The model employed seven CNN models including VGG19, DenseNet201, InceptionV3, ResNetV2, InceptionResNetV2, Xception, and MobileNetV2. The F1 scores of VGG19 and DenseNet201 were 0.91. A segmentation technique [30] has been used to identify lung regions infected by COVID-19. During pre-processing, CT images were enhanced using the Firefly Algorithm (FA) applying the guided Shannon Entropy (SE) thresholding method. The enhanced images were segmented employing the Markov Random Field (MRF) method. The segmentation accuracy exceeded 92%. Wang et al. [31] proposed a DL-based method to predict COVID-19 disease in CT images. The cited authors fine-tuned a modified Inception architecture and extracted CNN features for classification. The accuracy was 82.9% and the area under the curve (AUC) 0.90. Jaiswal et al. [32] extracted various textural features including the Local Directional Pattern (LDP), the Grey Level Co-occurrence Matrix (GLCM), the Gray Level Run Length Matrix (GLRLM), the Discrete Wavelet Transform (DWT), and the Grey Level Size Zone Matrix (GLSZM). The GLSZM afforded the highest accuracy of 99.68% when a support vector machine (SVM) classifier was employed. All of hybrid feature selection [33], a fuzzy approach with DL [34], a transfer learning-based method [35], and other approaches [36], have been described. In summary, these methods are tried to improve the classification accuracy but the main issue is availability of dataset. These methods are trained through classical techniques because of low dimensional datasets. In this work, we employed 58 patients' data and train a deep learning model instead of classical techniques.

3 Proposed Methodology

We develop automated prediction of positive COVID-19 pneumonia cases using CT scans. The positive cases are labeled via RT-PCR testing. The scheme features four steps. First, a database was prepared by collecting COVID-19 pneumonia-positive chest CT scans and normal scans from the Radiopaedia COVID-19 website (https://radiopaedia.org/cases). Next, the images were divided into training and test sets. Several features (the Dynamic Average LBP, the SFTA, the DWT, and the Reyni Entropy) were extracted from the training data. We used canonical correlation analysis (CCA) to fuse the features into single vectors; this enhanced the predictive power. We next implemented a GA using the fused feature vectors. The ELM classifier served as the GA fitness function. We used ELM loss to select the most discriminatory features and saved them as an ELM Model. Next, the test images were evaluated and the best-selected features compared to those of the trained model in terms of final predictions. A flow chart is shown in Fig. 1; 60% of the images were used to train the ELM Model and the remainder for testing.

3.1 Database Assembly

All images (58 patients) were collected from the Radiopaedia COVID-19 website. We downloaded data on 30 patients with COVID-19 pneumonia confirmed via RT-PCR. We gathered 3,000 COVID pneumonia-positive images and 2,500 control images (Fig. 2). All images were resized to 512 × 512 pixels. Case 2 was contributed by Dr. Chung et al. (https://radiopaedia.org/cases/covid-19-pneumonia-2), Case 3 by Dr. Bahman (https://radiopaedia.org/cases/covid-19-pneumonia-3), Case 4 by Dr. Bahman (https://radiopaedia.org/cases/covid-19-pneumonia-4), Case 7 by Dr. Domineco (https://radiopaedia.org/cases/covid-19-pneumonia-7), and Case 12 by Dr. Fabio (https: //radiopaedia.org/cases/covid-19-pneumonia-12), among others.

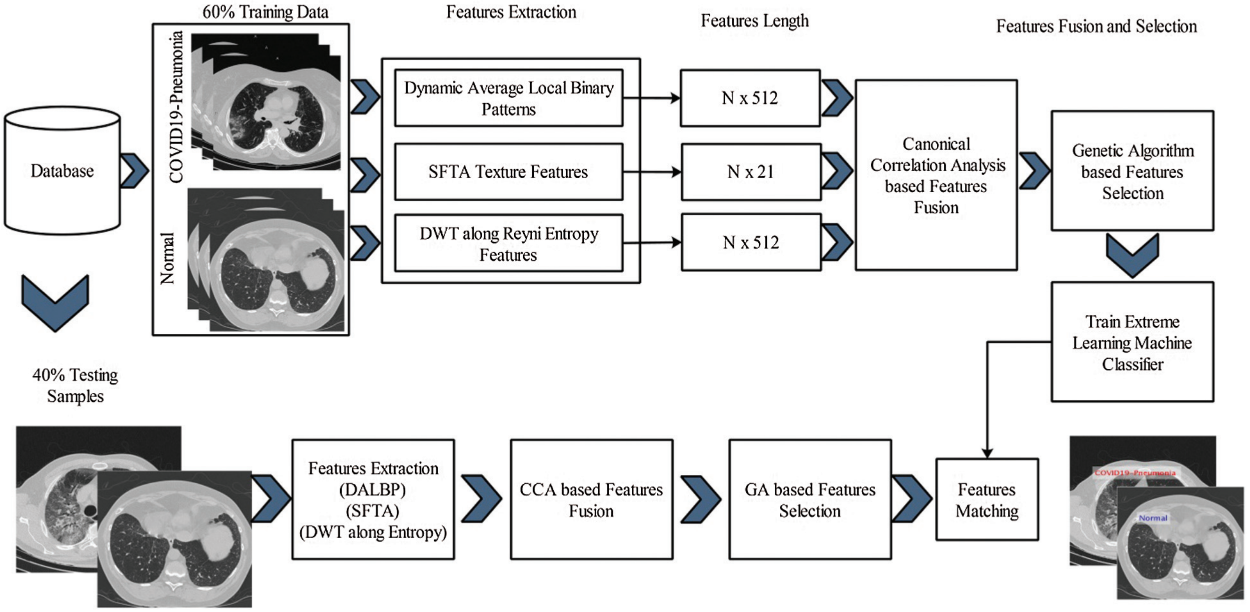

Database
60% Training Data
COVID19-Pneumonia
Normal
Features Extraction
Dynamic Average Local Binary Patterns
SFTA Texture Features
DWT along Reyni Entropy Features
Features Length
N x 512
N x 21
N x 512
Features Fusion and Selection
Canonical Correlation Analysis based Features Fusion
Genetic Algorithm based Features Selection
Train Extreme Learning Machine Classifier
40% Testing Samples
Features Extraction (DALBP) (SFTA) (DWT along Entropy)
CCA based Features Fusion
GA based Features Selection
Features Matching
COVID19-Pneumonia
Normal

Figure 1: A proposed architecture for prediction of COVID-19 positive chest CT scans using improved genetic algorithm based features selection

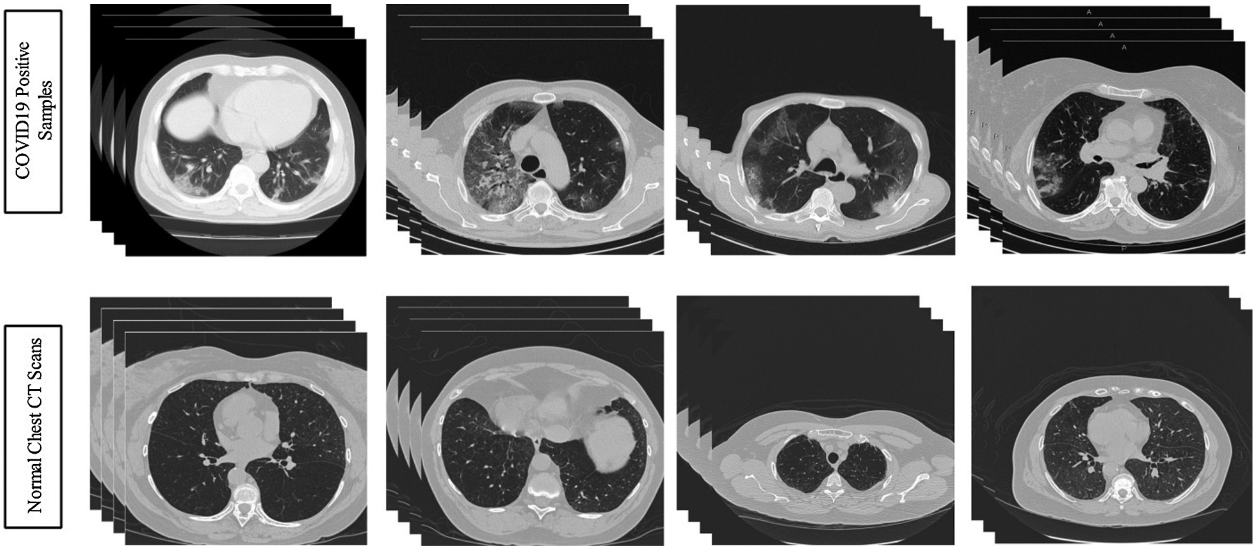


Figure 2: Sample images of COVID-19 positive and normal chest CT scans

3.2 Extraction of Classical Features

In terms of pattern recognition and machine-learning, features play pivotal roles in object representation. Many feature selection techniques for medical images are available [37]. However, some methods are better than others. Classical features are favored when data are lacking. We could not train an advanced machine-learning model such as a CNN; therefore, we extracted three features (the Dynamic Average LBP, the SFTA, and the DWT), and Entropy, as described below.

3.2.1 Dynamic Average LBP

Dynamic Average LBP (DALBP) features are modifications of the original LBP features; averages are used rather than central values. As in the original LBP [38], the principal value is replaced by neighboring values based on a threshold but, in the improved version, an average value is computed based on the dynamic window size. First, we examined the original LBP features. Assume that $\xi(x, y)$ is an original chest CT image of dimensions 512 × 512, and $\xi\,(x, y) \in \mathbb{R}^d$. Consider the $\xi = \{\xi_1, \xi_2, \xi_3, \ldots, \xi_N\} \in \mathbb{R}$ input images and their labels $l = \{l_1, l_2, \ldots, l_N\}$. For image $\xi(x, y)$, the LBP binary code is defined as follows:

$$\psi_{X,r} = \sum_{n=0}^{n-1} s(p_n - p_c)2^n \tag{1}$$

$$s\,(v) = \begin{cases} 1, & v \geq 0 \\ 0, & v < 0 \end{cases} \tag{2}$$

where p_c denotes the central pixel value, p_n the neighbor pixel values of p_c, r the radius, and X the total number of neighbors. Suppose the central coordinates are $p_c = (0, 0)$; those of p_n are then:

$$p_n = \left(rCos\left(\frac{2\pi n}{N}\right), rSin\left(\frac{2\pi n}{N}\right)\right) \tag{3}$$

As the image dimensions are 512 × 512, an LBP code histogram is computed as follows:

$$h\,(k) = \sum_{i=1}^{\mathbb{M}} \sum_{j=1}^{\mathbb{N}} f\left(\psi_{X,r}\,(i, j)\,, k\right) \tag{4}$$

$$f\,(u, v) = \begin{cases} 1 & u = v \\ 0 & Otherwise \end{cases} \tag{5}$$

where $k \in [1, 2, 3, \ldots, K]$ represents the maximal LBP code value. As for the DALBP, we employ the averages of two 3 × 3 windows, not the central pixel value. For each 3 × 3 block, the average value is replaced by the neighboring pixels.

Consider an image of dimensions 512 × 512 and divide it into 18 overlapping blocks [39] that are each then further divided into two equal blocks (9-bit vectors in each block). Next, compute the averages and extract the binary bits as follows:

$$\phi\,(x_1, t) = \begin{cases} 0, & x_1 - t < 0 \\ 1, & x_1 - t \geq 0 \end{cases} \tag{6}$$

where ϕ is the “Signum” function and (x_1, t) are the parameters of ϕ. This calculates the binary features. From each block, 8 bits are computed because execution runs from 1 to 9, where 9 denotes the bit vectors of a block. Later, these binary bits (both blocks) are converted into decimals, as follows:

$$D_{b1}(i) = \sum_{k=1}^{\mathbb{B}} \boldsymbol{B}_{b1}(k) \times 2^{\mathbb{B}-k} \tag{7}$$

$$D_{b2}(i) = \sum_{k=1}^{\mathbb{B}} \boldsymbol{B}_{b2}(k) \times 2^{\mathbb{B}-k} \tag{8}$$

where $\mathbb{B}$ denotes the bit vector length (8 in this work), $b1$ denotes block 1, $b2$ denotes block 2, the $D_{b1}(i)$ are the decimal values of the block 1 bits, and the $D_{b3}(i)$ the decimal values of the block 2 bits. Finally, a histogram is built employing $D_{b1}(i)$ and $D_{b2}(i)$ with the aid of Eqs. (4) and (5). The length of each histogram is 256. Then, the values of both histograms are concatenated into single vectors to yield the final DALBP features:

$$H(h_1, h_2) = \begin{pmatrix} h_1 \\ h_2 \end{pmatrix} \tag{9}$$

where the length of each final feature vector $[H(h_1, h_2)]$ is $N \times 512$ (N is the number of training images used for feature extraction).

3.2.2 SFTA Features

SFTA features, also termed textural features, are also used to extract discriminatory information. In the medical context, SFTA features are often used to describe organs. The principal textural descriptor is the GLCM, but the use thereof is very time-consuming. We used accurate SFTA features that can be rapidly extracted [40]. An image is initially decomposed into a binary image via two-threshold segmentation:

$$T_{th}\left(u', v'\right) = \begin{cases} 1 & \textit{if } L_{\breve{b}} < \xi(x, y) \le L_{\breve{u}} \\ 0 & \textit{Otherwise} \end{cases} \tag{10}$$

where $L_{\breve{b}}$ denotes the lower bound of the threshold, $L_{\breve{u}}$ the upper bound, and $T_{th}\left(u', v'\right)$ the binary image. The SFTA feature vector is constructed in fractal dimensions as follows:

$$\delta(u, v) = \begin{cases} 1 & \textit{if } \exists\left(u', v'\right) \in \Psi_8(u, v): \\ & T_{th}\left(u', v'\right) = 0 \wedge \\ & T_{th}(u, v) = 1 \\ 0, & \textit{Otherwise} \end{cases} \tag{11}$$

where Ψ_8 denotes the set of pixels that are 8-connected to (u, v). Hence, after binary box-counting, we obtain a textural feature vector of dimensions $\mathrm{N} \times 21$.

3.2.3 DWT and Entropy

Discrete Wavelet Transform (DWT) is a well-known type of feature extraction that analyzes images at various scales and resolutions [41]. Let $g(x)$ be a continuous square-integral function. Mathematically, this is:

$$W_{\varphi}\left(s_1, s_2\right) = \int_{-\infty}^{+\infty} g\left(x\right) \varphi_{s_1, s_2}(x) dx \tag{12}$$

$$\varphi_{s_1, s_2}\left(x\right) = \frac{1}{\sqrt{s_1}} \varphi\left(\frac{x - s_2}{s_1}\right) \tag{13}$$

$$s_1 \in \mathbb{R}^{+ve}, \quad s_2 \in \mathbb{R}$$

where s_1 and s_2 are the scale and translation parameters of a real-valued φ. A discrete variation of Eq. (13) can be ensured by limiting s_1 and s_2 to a discrete lattice with $s_1 = 2^j$ and $s_2 = 2^j k$. Mathematically, this is:

$$DW_{g(n)} = \begin{cases} \mathbb{A}_{j,k}^{c}\left(n\right) = \sum_{n} g\left(n\right) \boldsymbol{F}_j^{*}(n - 2^j k) \\ \mathbb{D}_{j,k}^{c}\left(n\right) = \sum_{n} g\left(n\right) \boldsymbol{E}_j^{*}(n - 2^j k) \end{cases} \tag{14}$$

where $\mathbb{A}_{j,k}^{c}\left(n\right)$ is the coefficient of the approximation component, $\mathbb{D}_{j,k}^{c}\left(n\right)$ the coefficient of the detail component, and $\boldsymbol{F}_j^{*}$ and $\boldsymbol{E}_j^{*}$ the low- and high-pass coefficient filters. More detail is given in [41]. The discrete coefficient components $DW_{g(n)}$ employ the Renyi entropy for final feature extraction. Mathematically, the Renyi entropy is:

$$\mathbb{E}_{\alpha}^{h}\left(DW\right) = \frac{1}{1-\alpha} \log\left(\sum_{i=1}^{n} p_i^{\alpha}\left(DW_{g(n)}\right)\right) \tag{15}$$

The Renyi entropy feature vector is computed in row coefficients; the output vector is of dimensions $N \times 512$. Entropy replaces the zero and negative features with positive values.

3.3 Feature Fusion

All extracted features were fused employing canonical correlation analysis (CCA) [42]. The purpose was to enhance the learning capacity of each feature vector in terms of correct predictions of COVID-19-positive chest scans and normal scans. Suppose that $\Upsilon_1 \in R^{r\times n}$, $\Upsilon_2 \in R^{q\times n}$, and $\Upsilon_3 \in R^{s\times n}$ are three feature vectors extracted using the DALBP, SFTA, and DWT-plus-Entropy approaches, respectively. The dimensions are $N \times 512$, $N \times 21$, and $N \times 512$, respectively. Let $\Delta_{xx} \in R^{r\times n}$, $\Delta_{yy} \in R^{p\times n}$, and $\Delta_{zz} \in R^{s\times n}$ represent the covariance matrices of Υ_1, Υ_2, and Υ_3 respectively. Also, let $\Delta_{xy} \in R^{r\times q}$, $\Delta_{yz} \in R^{p\times s}$, and $\Delta_{xz} \in R^{r\times s}$ represent the between-sets covariance matrices. We consider Υ_1 and Υ_3 when computing the between-set covariance:

$$S = \begin{pmatrix} Cov\left(\Upsilon_1\right) & Cov(x, z) \\ Cov\left(z, \Upsilon_1\right) & Cov(z) \end{pmatrix} = \begin{pmatrix} \Delta_{xx} & \Delta_{xz} \\ \Delta_{zx} & \Delta_{zz} \end{pmatrix} \tag{16}$$

where $\Delta_{zx} = \Delta_{xz}^T$ and the overall covariance matrix is $S \in R^{(r+s)\times(r+s)}$. CCA seeks to compute linear combinations $\Upsilon_1^* = \mathrm{W}_x^T \Upsilon_1$ and $\Upsilon_3^* = \mathrm{W}_z^T \Upsilon_3$ that maximize the pairwise correlations between two feature vectors as:

$$Corr(\Upsilon_1^*, \Upsilon_3^*) = \frac{Cov(\Upsilon_1^*, \Upsilon_3^*)}{Var\left(\Upsilon_1^*\right).Var(\Upsilon_3^*)} \tag{17}$$

$$Cov\left(\Upsilon_1^*, \Upsilon_3^*\right) = \mathrm{W}_x^T \Delta_{xz} \mathrm{W}_z \tag{18}$$

$$Var\left(\Upsilon_1^*\right) = \mathrm{W}_x^T \Delta_{xx} \mathrm{W}_x \tag{19}$$

$$Var\left(\Upsilon_3^*\right) = \mathrm{W}_z^T \Delta_{zz} \mathrm{W}_{xz} \tag{20}$$

Next, Lagrange multipliers are used to solve the maximization problem between Υ_1^* and Υ_3^*; the following condition is satisfied:

$$Var\left(\Upsilon_1^*\right) = Var\left(\Upsilon_3^*\right) = 1 \tag{21}$$

Finally, the transformed features are combined as follows:

$$\Phi = \begin{pmatrix} \Upsilon_1^* \\ \Upsilon_3^* \end{pmatrix} \tag{22}$$

$$= \begin{pmatrix} W_{\Upsilon_1} & 0 \\ 0 & W_{\Upsilon_3}^T \end{pmatrix} \begin{pmatrix} \Upsilon_1 \\ \Upsilon_3 \end{pmatrix} \tag{23}$$

$$\Phi = \begin{pmatrix} W_{\Upsilon_1} \\ W_{\Upsilon_3}^T \end{pmatrix} \begin{pmatrix} \Upsilon_1 \\ \Upsilon_3 \end{pmatrix} = \Upsilon_1^* + \Upsilon_3^* \tag{24}$$

where Φ denotes the CCA fused vector. The process is repeated for Υ_2. Finally, we obtain a fused CCA vector of dimensions $N \times 776$ (in this work; the dimensions vary by the dataset).

3.4 Feature Selection

Feature selection involves selection of the best subset of input feature vectors based on a defined criterion. Use of a best subset improves learning and predictive accuracy, and reduces the computational time. We implemented a GA and an ELM fitness function. Most researchers use the Fine KNN and SVM for fitness calculations; however, we believe that the ELM is more efficient. The GA is Algorithm 1. The initial population size is 100, the number of iterations 1,000, the crossover rate (ψ_{cr}) 0.4, the mutation rate (ψ_{mr}) 0.01, and the selection pressure (β) 5. In Step 2, ELM-based fitness is calculated for K-Fold = 5 and $E_{loss} = |Network\ Output - Original\ Label|$. In Step 3, the selection is performed via a Roulette Wheel that follows the crossover and mutational steps. Step 2 is repeated until the desired accuracy is attained. The final robust vector is denoted $\Phi_{fs}(i)$ of dimensions $N \times K$ (here, $\mathrm{K} = 426$, this the length of the final feature vector).

```
Algorithm 1
Output: $\Phi_{fs}(i) \leftarrow Robust\ Vector$
Input: $\xi_{fd}(i) \leftarrow Fused\ Vector$
Step 1: Parameters Initialization
  —$Population \leftarrow N = 100$
  —$Iterations \leftarrow T = 1000$
  —$\psi_{cr} \leftarrow 0.4$
  —$\psi_{mr} \leftarrow 0.01$
  —$\beta \leftarrow 5$
Start
Step 2: Fitness Function
  —$Extreme\ Learning\ Machine$
  —$K\text{-}fold \leftarrow 5$
  —$E_{loss} \leftarrow |Network\ Output - Original\ Label|$
Step 3: Selection
  —$S \leftarrow \frac{s_i}{\sum(s_i)}$
$s_i \leftarrow \exp\left(-p_1 \times \frac{X_\beta}{S_L}\right)$
Step 4: Crossover (Uniform Crossover)
  —$\psi_{cr} \leftarrow CrossOver(L_1, L_2)$
Step 5: Mutation
  —$Type \leftarrow Uniform$
Step 6: Repeat Step 2
Step 7: $\Phi_{fs}(i) \leftarrow Best\ Features$
End
```

3.5 The Extreme Learning Machine (ELM)

Given a robust feature vector after GA and appropriate labeling $\left\{(\Phi_i, l_i) \mid \Phi_i \in R^d, l_i \in R^m, i = 1, 2, 3, \ldots, N\right\}$ (where Φ_i denotes the selected features and l_i the target labels), the input features (weights) and target outputs that minimize the error [43] are defined as:

$$min_\beta \left\| H_l \beta - \widehat{\boldsymbol{O}} \right\|_{FN}^2 \tag{25}$$

$$H_l = \left\{O_{ij}(.)\right\}_{N \times L} \quad with\ O_{ij}(.) = O\left(w_j.\Phi_i + b_j\right) \tag{26}$$

where H_l denotes the hidden-layer, output weight matrix; $\widehat{\boldsymbol{O}}$ the target output; $\|.\|_{FN}^2$ the Frobenius Norm; L the hidden nodes; β the weight matrix; $O(.)$ the activation function; and w_j, b_j the weight and bias matrices for the jth node. The output weight matrix β is solved as follows:

$$\check{\beta} = \widetilde{H}_l.\widehat{\boldsymbol{O}} \tag{27}$$

$\widetilde{H}_l$ is the generalized inverse of H_l. Next, regularization is used to minimize the training error:

$$min_\beta \frac{1}{2}\mathbb{C} \left\| H_l \beta - \widehat{\boldsymbol{O}} \right\|_{FN}^2 + \frac{1}{2} \|\beta\|_{FN}^2 \tag{28}$$

where $\mathbb{C}$ is the the tradeoff between the training error and the norm of the output weight. The error between $\widehat{\boldsymbol{O}}$ and a target label l is:

$$Error = |\widehat{\boldsymbol{O}} - l| \tag{29}$$

Based on this error, $l = 1$ and $l = -1$ are the outputs. $l = 1$ indicates COVID-19 pneumonia and $l = -1$ a healthy lung. The model is trained using 60% of the data and the test images then passed to the scheme to select the best features, The chosen features are compared to those of the trained ELM classifier and the predictive outputs are both labels and numerical values. A few labeled images are shown in Fig. 3; these are predictions made during ELM testing.

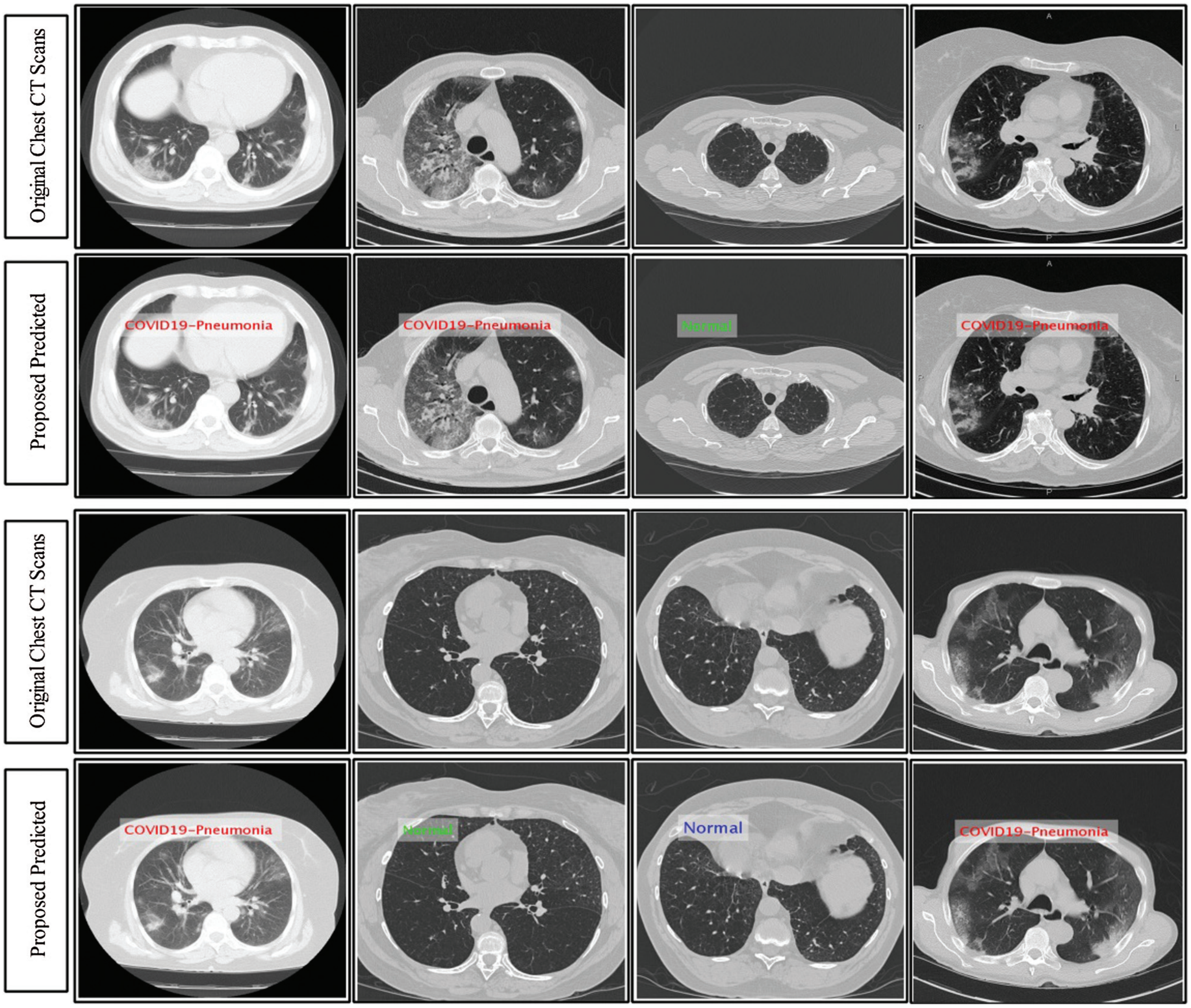


Figure 3: Proposed prediction results of ELM during the testing step

4 Experimental Setup and Results

We used publicly available chest CT images (Section 3.1). We extracted DALBP, SFTA, and DWT-plus-Entropy features, fused them using the CCA approach, employed a GA to select robust features, and delivered these to the ELM classifier for final predictions. We compared ELM performance to those of Logistic Regression, Q-SVM, Fine Gaussian, Fine Tree, and Cubic KNN in terms of accuracy, Precision, Recall, Specificity, and the F1 score. We used an Intel Core i7 8th generation CPU equipped with 16 GB of RAM and 8 GB of GPU running MATLAB 2019b software. We took a 60–40 approach with 10-fold cross-validation during both training and testing.

4.1 Results and Discussion

The predictions are presented in numerical form and as bar graphs. We explored the accuracies afforded by DALBP, SFTA, and DWT-plus-Entropy features; CCA fusion accuracy; and GA accuracy. For the DALBP features (Tab. 1) the highest accuracy was 84.52% using the Quadratic SVM classifier; the figure for the ELM classifier was 82.30%. The linear and Naïve Bayes accuracies were 82.42 and 81.49%. The accuracies were 80.20%, 81.21%, 80.14%, 81.63%, 79.38%, 81.52%, 78.12%, 79.68%, 80.72% and 76.71% respectively. For the SFTA textural features (Tab. 1), the accuracies were lower than the DALBP figures. The highest accuracy was 78.95%, and the accuracies afforded by the other listed classifiers were 75.27%, 73.78%, 71.62%, 71.96%, 77.52%, 75.06%, 72.94%, 77.52%, 74.22%, 74.36%, 79.42%, 73.21%, and 76.13% respectively. For the DWT-plus-Entropy features, the maximum accuracy was 82.60% (Naïve Bayes), higher than any SFTA figure. Thus, the DALBP features were optimal.

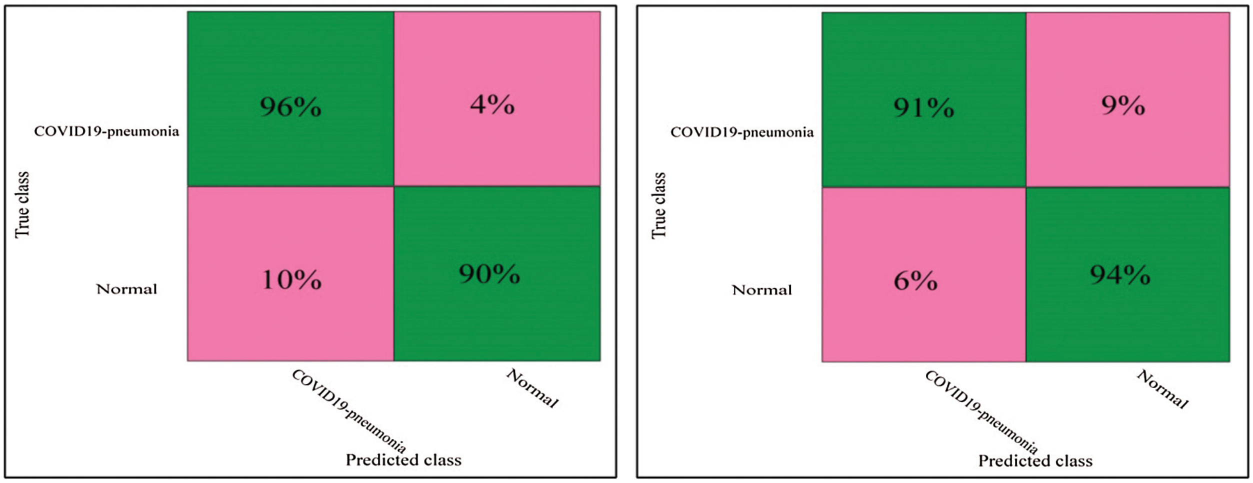
Table 1: Prediction accuracy of COVID-19 pneumonia and normal cases using separate features without fusion and selection

Classifier	Features bases accuracy		
	DALBP	SFTA	DWT-Entropy
ELM	82.30	**78.95**	81.62
Linear SVM	82.42	75.27	80.09
Naïve Bayes	81.49	73.78	**82.60**
Quadratic SVM	**84.52**	71.62	77.51
Cosine KNN	80.20	71.96	79.43
Medium KNN	81.21	77.52	78.63
ESDA	80.14	75.06	79.61
MG SVM	81.63	72.94	80.33
Logistic regression	79.38	77.52	78.41
Fine tree	81.52	74.22	79.08
Fine Gaussian SVM	78.12	74.36	77.92
Cubic KNN	79.68	79.42	77.51
Cubic SVM	80.72	73.21	75.04
EBT	76.71	76.13	75.91

The CC-based fusion approach was then employed for prediction. The ELM classifier was best: 92.8% accuracy, 93.81% precision, 94% specificity, and an F1 score of 0.93. The worst classifier was the EBT (86.1% accuracy). All results are shown in Tab. 2. The classifier accuracies were 92.7%, 92.6%, 92.2%, 92.2%, 91.5%, 90.8%, 90.6%, 90.3%, 90.2%, 90.2%, 89.8%, 89.7%, and 86.1%. Thus, CCA-based fusion improved accuracy by about 10%. The fusion accuracies of ELM and LSVM are shown in Fig. 4 (a confusion matrix).

Table 2: Proposed prediction results of multi-type features fusion

Classifier	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	AUC	F1 score (%)
ELM	**92.8**	**93.81**	92.5	**94.0**	1.00	**93.15**
Linear SVM	92.7	90.57	93.0	90.0	0.98	91.77
Naïve Bayes	92.6	90.48	92.5	90.0	0.96	91.48
Quadratic SVM	92.2	92.00	92.0	92.0	0.99	92.00
Cosine KNN	92.2	90.48	92.5	90.0	0.96	91.48
Medium KNN	91.5	90.29	91.5	90.0	1.00	90.89
ESDA	90.8	89.42	91.0	89.0	0.99	90.20
MG SVM	90.6	89.32	90.3	90.5	0.99	91.60
Logistic regression	90.3	91.75	90.5	92.0	1.00	91.12
Fine tree	90.2	90.82	90.0	91.0	1.00	90.41
Fine Gaussian SVM	90.2	90.00	90.0	90.0	0.95	90.00
Cubic KNN	89.8	91.67	90.0	92.0	0.95	90.83
Cubic SVM	89.7	89.90	89.5	90.0	0.99	89.70
EBT	86.1	91.95	86.5	93.0	0.92	89.14



Confusion Matrix of LSVM after Features Fusion

Confusion Matrix of ELM Classifier after Features Fusion

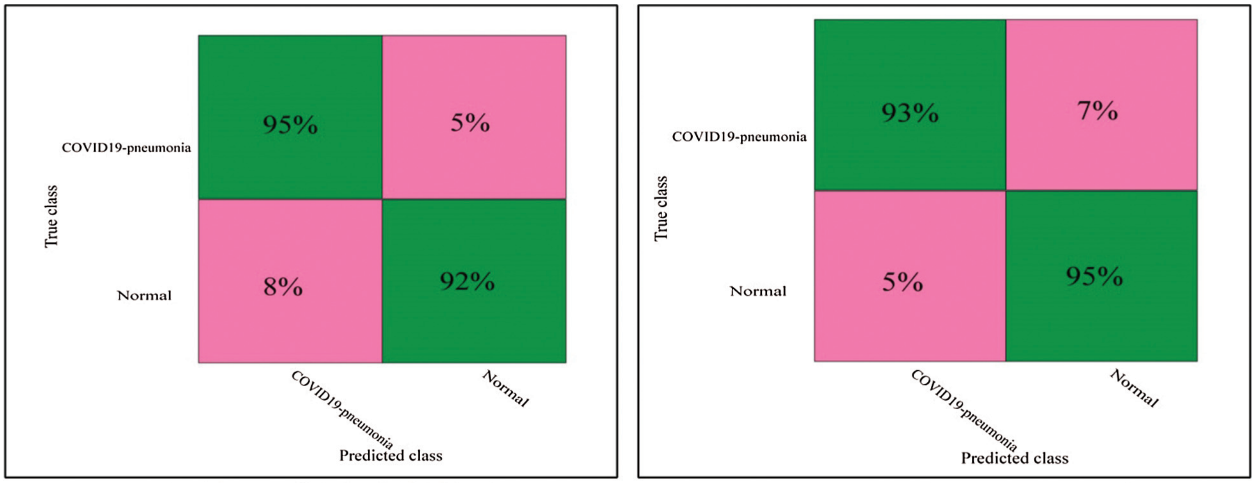
Figure 4: Confusion matrix of LSVM and ELM after multi-type features fusion

We used an improved GA to select the best features for final prediction (Section 3.4 and Algorithm 1). The predictive performances of several classifiers are shown in Tab. 3. All improved after implementation of feature selection. The ELM classifier was robust (accuracy 93.9%, precision 93.14%, specificity 95%, recall 94%, and F1 Score 0.94) The next best classifier was the linear SVM (LSVM) (93.4% accurate). Fig. 5 shows the ELM/Linear SVM confusion matrix;

feature fusion and selection improved performance. Fig. 6 shows the receiver-operator curves for prediction of healthy and COVID-19 pneumonia CT images; the AUCs were verified. In summary, selection of the best features afford excellent prediction of COVID-positive and normal chest CT scans.

Table 3: Proposed prediction results after best features selection using improved genetic algorithm and ELM

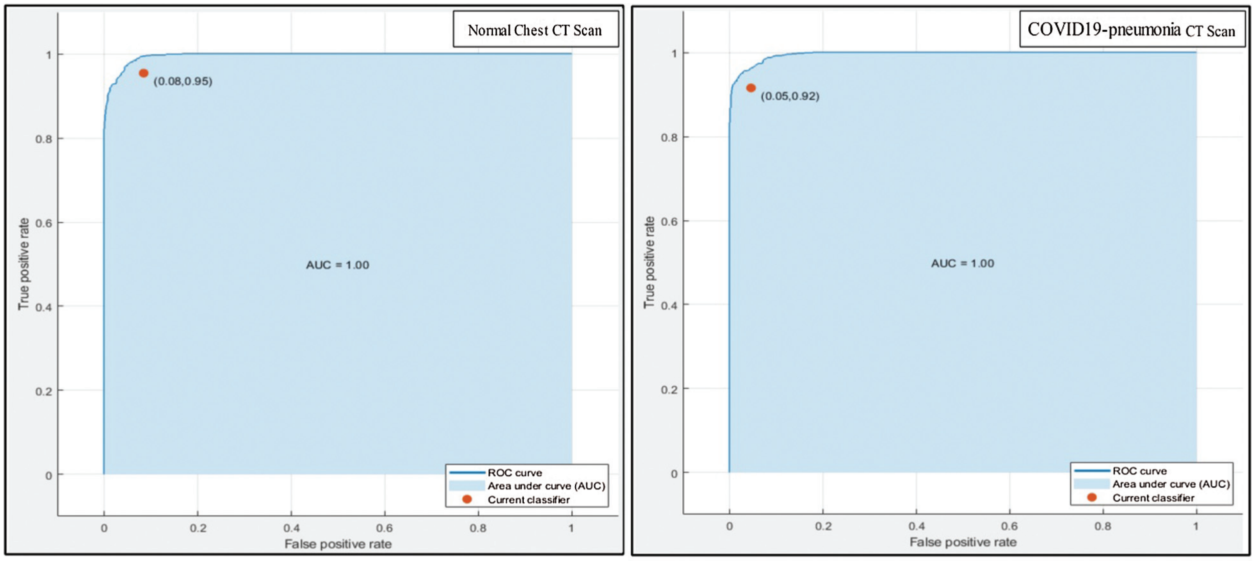
Classifier	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	AUC	F1 score (%)
ELM	**93.9**	**93.14**	**94.0**	**95.0**	1.00	**93.57**
Linear SVM	93.4	92.03	93.6	92.0	1.00	92.81
Naïve Bayes	93.7	93.05	93.9	94.0	0.98	93.47
Quadratic SVM	93.6	93.07	93.5	93.0	0.99	93.28
Cosine KNN	93.6	93.07	93.5	93.0	0.99	93.28
Medium KNN	93.5	93.07	93.5	93.0	1.00	93.28
ESDA	93.4	92.16	93.0	92.0	0.99	92.58
MGSVM	93.4	92.23	93.5	92.0	0.99	92.86
Logistic regression	93.4	93.12	93.5	94.0	1.00	93.31
Fine tree	93.2	89.19	93.5	88.0	0.98	91.29
Fine Gaussian SVM	93.0	92.16	93.0	92.0	0.98	92.58
Cubic KNN	92.9	92.16	93.0	92.0	1.00	92.58
Cubic SVM	92.8	92.16	93.0	92.0	0.98	92.58
EBT	89.5	90.72	89.5	91.0	0.94	90.11



Confusion Matrix of LSVM after GA based Features Selection

Confusion Matrix of ELM Classifier after GA based Features Selection

Figure 5: Confusion matrix of LSVM and ELM after best features selection using improved genetic algorithm



ROC Curve of Normal Predicted Chest CT Scans for ELM Classifier

ROC Curve of COVID19-pneumonia Predicted Chest CT Scans for ELM Classifier

Figure 6: Representation of ELM performance based on ROC plots

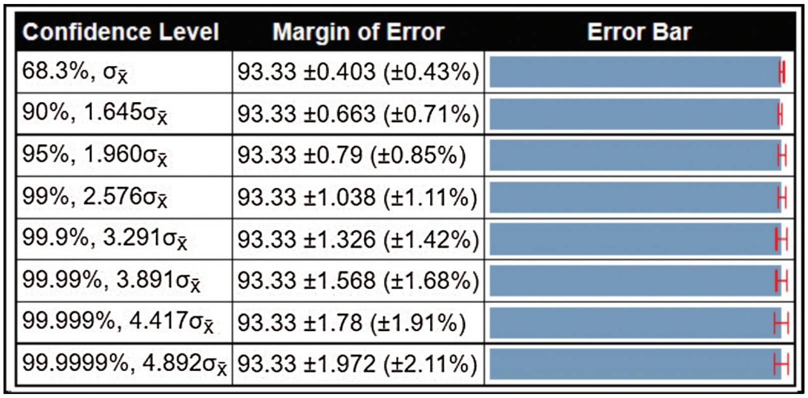
5 Analysis and Conclusion

Tab. 1 shows the predictive accuracies of various features prior to fusion and selection. The highest accuracies were 84.52, 78.95, and 82.60 for the DALBP, SFTA, and DWT-plus-Entropy features. After CCA-mediated fusion, the figures rose by 10%. The highest accuracy after fusion was 92.5% (Tab. 2). Accuracy was further improved by selection (Tab. 3 and Figs. 5 and 6). We analyzed the utility of selection by calculating standard errors of the mean (SEMs) (Tab. 4) for the ELM and the other three top classifiers. The minimum accuracy of ELM after 100 iterations was 92.76% and the highest accuracy 93.90%; the SEM was 0.4040. Thus, a minor change occurred after execution. The error bars (Fig. 7) are based on the SEMs and confidence levels; the scheme is scalable.

Table 4: Analysis of proposed selection features based prediction accuracy on ELM and three other top classifiers

Classifier	Min accuracy (%)	Average accuracy (%)	Max accuracy (%)	SEM (%)
ELM	**92.76**	**93.33**	**93.90**	**0.4030**
LSVM	92.24	92.82	93.40	0.4101
Naïve Bayes	92.49	93.09	93.70	0.4277
Quadratic SVM	92.38	92.99	93.60	0.4313

It is clear that fusion of multi-type features is valuable. This increases the number of predictors and enhances predictive accuracy. However, a few irrelevant features were added; if these are removed, accuracy is not compromised. Removal was effected via feature selection. We used an improved GA and an ELM to select the best features and improve predictive accuracy. In future work, we will seek a more efficient feature selection algorithm to improve accuracy further. Moreover, we will seek to build a larger image dataset that we will use to train a CNN.



Confidence Level	Margin of Error	Error Bar
68.3%, $\sigma_{\bar{x}}$	93.33 ±0.403 (±0.43%)	
90%, 1.645$\sigma_{\bar{x}}$	93.33 ±0.663 (±0.71%)	
95%, 1.960$\sigma_{\bar{x}}$	93.33 ±0.79 (±0.85%)	
99%, 2.576$\sigma_{\bar{x}}$	93.33 ±1.038 (±1.11%)	
99.9%, 3.291$\sigma_{\bar{x}}$	93.33 ±1.326 (±1.42%)	
99.99%, 3.891$\sigma_{\bar{x}}$	93.33 ±1.568 (±1.68%)	
99.999%, 4.417$\sigma_{\bar{x}}$	93.33 ±1.78 (±1.91%)	
99.9999%, 4.892$\sigma_{\bar{x}}$	93.33 ±1.972 (±2.11%)	

Figure 7: Confidence interval of ELM after best features selection using improved GA

Funding Statement: This research was supported by Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (P0012724, The Competency Development Program for Industry Specialist) and the Soonchunhyang University Research Fund.

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

References

[1] N. Zhu, D. Zhang, W. Wang, X. Li, B. Yang *et al.,* "A novel coronavirus from patients with pneumonia in China," *2019 New England Journal of Medicine*, vol. 328, no. 8, pp. 727–723, 2020.

[2] A. E. Gorbalenya, S. C. Baker, R. Baric, R. J. d. Groot, C. Drosten *et al.,* "Severe acute respiratory syndrome-related coronavirus: The species and its viruses-a statement of the Coronavirus Study Group," 2020.

[3] K. Roosa, Y. Lee, R. Luo, A. Kirpich, R. Rothenberg *et al.,* "Real-time forecasts of the COVID-19 epidemic in China from february 5th to february 24th," *2020 Infectious Disease Modelling*, vol. 5, no. 5, pp. 256–263, 2020.

[4] C. Sohrabi, Z. Alsafi, N. O'Neill, M. Khan, A. Kerwan *et al.,* "World health organization declares global emergency: A review of the 2019 novel coronavirus (COVID-19)," *International Journal of Surgery*, vol. 76, pp. 71–76, 2020.

[5] S. B. Stoecklin, P. Rolland, Y. Silue, A. Mailles, C. Campese *et al.,* "First cases of coronavirus disease 2019 (COVID-19) in France: Surveillance, investigations and control measures, January 2020," *Eurosurveillance*, vol. 25, pp. 2000094, 2020.

[6] W. Wang, J. Tang and F. Wei, "Updated understanding of the outbreak of 2019 novel coronavirus (2019-nCoV) in Wuhan," *China Journal of Medical Virology*, vol. 92, no. 4, pp. 441–447, 2020.

[7] L. Brunese, F. Mercaldo, A. Reginelli and A. Santone, "Explainable deep learning for pulmonary disease and coronavirus COVID-19 detection from X-rays," *Computer Methods and Programs in Biomedicine*, vol. 196, no. 20, pp. 105608, 2020.

[8] D. Dansana, R. Kumar, A. Bhattacharjee, D. J. Hemanth, D. Gupta *et al.,* "Early diagnosis of COVID-19-affected patients based on X-ray and computed tomography images using deep learning algorithm," *Soft Computing*, vol. 5, pp. 1–9, 2020.

[9] A. Ulhaq, J. Born, A. Khan, D. P. S. Gomes, S. Chakraborty *et al.,* "Covid-19 control by computer vision approaches: A survey," *IEEE Access*, vol. 8, pp. 179437–179456, 2020.

[10] A. Gupta, S. Gupta and R. Katarya, "InstaCovNet-19: A deep learning classification model for the detection of COVID-19 patients using Chest X-ray," *Applied Soft Computing*, vol. 4, pp. 106859, 2020.

[11] M. A. Khan, M. S. Sarfraz, M. Alhaisoni, A. A. Albesher, S. Wang *et al.,* "StomachNet: Optimal deep learning features fusion for stomach abnormalities classification," *IEEE Access*, vol. 8, pp. 197969–197981, 2020.

[12] M. A. Khan, S. Rubab, A. Kashif, M. I. Sharif, N. Muhammad *et al.,* "Lungs cancer classification from CT images: An integrated design of contrast based classical features fusion and selection," *Pattern Recognition Letters*, vol. 129, pp. 77–85, 2020.

[13] M. A. Khan, I. Ashraf, M. Alhaisoni, R. Damaševičius, R. Scherer *et al.,* "Multimodal brain tumor classification using deep learning and robust feature selection: A machine learning application for radiologists," *Diagnostics*, vol. 10, no. 8, pp. 565, 2020.

[14] L. Lan, C. Ye, C. Wang and S. Zhou, "Deep convolutional neural networks for WCE abnormality detection: CNN architecture, region proposal and transfer learning," *IEEE Access*, vol. 7, pp. 30017–30032, 2019.

[15] V. Murugappan and R. Sabeenian, "Texture based medical image classification by using multi-scale gabor rotation-invariant local binary pattern (MGRLBP)," *Cluster Computing*, vol. 22, no. S5, pp. 10979–10992, 2019.

[16] M. Sharif, M. Attique Khan, M. Rashid, M. Yasmin, F. Afza *et al.,* "Deep CNN and geometric features-based gastrointestinal tract diseases detection and classification from wireless capsule endoscopy images," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 3, pp. 1–23, 2019.

[17] A. Mahbod, G. Schaefer, I. Ellinger, R. Ecker, A. Pitiot *et al.,* "Fusing fine-tuned deep features for skin lesion classification," *Computerized Medical Imaging and Graphics*, vol. 71, no. 4, pp. 19–29, 2019.

[18] M. A. Khan, M. A. Khan, F. Ahmed, M. Mittal, L. M. Goyal *et al.,* "Gastrointestinal diseases segmentation and classification based on duo-deep architectures," *Pattern Recognition Letters*, vol. 131, pp. 193–204, 2020.

[19] M. I. Sharif, J. P. Li, M. A. Khan and M. A. Saleem, "Active deep neural network features selection for segmentation and recognition of brain tumors using MRI images," *Pattern Recognition Letters*, vol. 129, no. 10, pp. 181–189, 2020.

[20] M. A. Khan, T. Akram, M. Sharif, A. Shahzad, K. Aurangzeb *et al.,* "An implementation of normal distribution based segmentation and entropy controlled features selection for skin lesion detection and classification," *BMC Cancer*, vol. 18, no. 1, pp. 638, 2018.

[21] A. Majid, M. A. Khan, M. Yasmin, A. Rehman, A. Yousafzai *et al.,* "Classification of stomach infections: A paradigm of convolutional neural network along with classical features fusion and selection," *Microscopy Research and Technique*, vol. 83, no. 5, pp. 562–576, 2020.

[22] M. A. Khan, M. I. Sharif, M. Raza, A. Anjum, T. Saba *et al.,* "Skin lesion segmentation and classification: A unified framework of deep neural network features fusion and selection," *Expert Systems*, vol. 3, pp. e12497, 2019.

[23] I. D. Apostolopoulos and T. A. Mpesiana, "Covid-19: Automatic detection from x-ray images utilizing transfer learning with convolutional neural networks," *Physical and Engineering Sciences in Medicine*, vol. 43, no. 2, pp. 1, 2020.

[24] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang *et al.,* "Mobilenets: Efficient convolutional neural networks for mobile vision applications," arXiv preprint arXiv: 1704.04861, 2017.

[25] L. Li, L. Qin, Z. Xu, Y. Yin, X. Wang *et al.,* "Artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT," *Radiology*, vol. 296, no. 2, 2020.

[26] F. Shan, Y. Gao, J. Wang, W. Shi, N. Shi *et al.,* "Lung infection quantification of covid-19 in ct images with deep learning," arXiv preprint arXiv: 2003.04655, 2020.

[27] N. Tsiknakis, E. Trivizakis, E. E. Vassalou, G. Z. Papadakis, D. A. Spandidos *et al.,* "Interpretable artificial intelligence framework for COVID-19 screening on chest X-rays," *Experimental and Therapeutic Medicine*, vol. 20, no. 2, pp. 727–735, 2020.

[28] A. Narin, C. Kaya and Z. Pamuk, "Automatic detection of coronavirus disease (covid-19) using X-ray images and deep convolutional neural networks," arXiv preprint arXiv: 2003.10849, 2020.

[29] E. E. D. Hemdan, M. A. Shouman and M. E. Karar, "Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray images," arXiv preprint arXiv: 2003.11055, 2020.

[30] V. Rajinikanth, S. Kadry, K. P. Thanaraj, K. Kamalanand and S. Seo, "Firefly-algorithm aupported scheme to detect COVID-19 lesion in lung CT scan images using shannon entropy and markov-random-field," arXiv preprint arXiv: 2004.09239, 2020.

[31] S. Wang, B. Kang, J. Ma, X. Zeng, M. Xiao *et al.,* "A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19)," *MedRxiv*, vol. 57, no. 7, pp. 1451, 2020.

[32] A. Jaiswal, N. Gianchandani, D. Singh, V. Kumar and M. Kaur, "Classification of the COVID-19 infected patients using DenseNet201 based deep transfer learning," *Journal of Biomolecular Structure and Dynamics*, pp. 1–8, 2020.

[33] W. M. Shaban, A. H. Rabie, A. I. Saleh and M. Abo-Elsoud, "A new COVID-19 patients detection strategy (CPDS) based on hybrid feature selection and enhanced KNN classifier," *Knowledge-Based Systems*, vol. 205, no. 1, pp. 106270, 2020.

[34] W. M. Shaban, A. H. Rabie, A. I. Saleh and M. Abo-Elsoud, "Detecting COVID-19 patients based on fuzzy inference engine and deep neural network," *Applied Soft Computing*, vol. 99, no. 1, pp. 106906, 2020.

[35] M. J. Horry, S. Chakraborty, M. Paul, A. Ulhaq, B. Pradhan *et al.,* "COVID-19 detection through transfer learning using multimodal imaging data," *IEEE Access*, vol. 8, pp. 149808–149824, 2020.

[36] M. F. Aslan, M. F. Unlersen, K. Sabanci and A. Durdu, "CNN-based transfer learning-BiLSTM network: A novel approach for COVID-19 infection detection," *Applied Soft Computing*, vol. 98, no. 1, pp. 106912, 2020.

[37] A. Rehman, M. A. Khan, Z. Mehmood, T. Saba, M. Sardaraz *et al.,* "Microscopic melanoma detection and classification: A framework of pixel-based fusion and multilevel features reduction," *Microscopy Research and Technique*, vol. 83, no. 4, pp. 410–423, 2020.

[38] T. Ojala, M. Pietikainen and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[39] T. Tuncer, E. Aydemir and S. Dogan, "Automated ambient recognition method based on dynamic center mirror local binary pattern: DCMLBP," *Applied Acoustics*, vol. 161, pp. 107165, 2020.

[40] A. F. Costa, G. Humpire-Mamani and A. J. M. Traina, "An efficient algorithm for fractal analysis of textures," in *2012 25th SIBGRAPI Conf. on Graphics, Patterns and Images*, Ouro Preto, Brazil, pp. 39–46, 2012.

[41] D. R. Nayak, R. Dash and B. Majhi, "Brain MR image classification using two-dimensional discrete wavelet transform and AdaBoost with random forests," *Neurocomputing*, vol. 177, no. 8, pp. 188–197, 2016.

[42] Q.-S. Sun, S.-G. Zeng, Y. Liu, P.-A. Heng and D.-S. Xia, "A new method of feature fusion and its application in image recognition," *Pattern Recognition*, vol. 38, no. 12, pp. 2437–2448, 2005.

[43] H. Dai, J. Cao, T. Wang, M. Deng and Z. Yang, "Multilayer one-class extreme learning machine," *Neural Networks*, vol. 115, no. 12, pp. 11–22, 2019.