A Computerized Doughty Predictor Framework for Corona Virus Disease: Combined Deep Learning based Approach

Dr. Ramya P[1*], and Dr. Venkatesh Babu S[2]
[1] Department of Information Technology
[e-mail: ramyajun2016@gmail.com]
[2] Department of Computer Science and Engineering
[1,2]Christian College of Engineering and Technology, Oddanchatram, Dindigul, Tamil Nadu-624619
[e-mail : venkateshflower6@gmail.com]
[*]Corresponding author: Dr.Ramya P

Received November 7, 2021; revised February 16, 2022; revised March 28, 2022; accepted June 1, 2022; published June 30, 2022

Abstract

Nowadays, COVID-19 infections are influencing our daily lives which have spread globally. The major symptoms' of COVID-19 are dry cough, sore throat, and fever which in turn to critical complications like multi organs failure, acute respiratory distress syndrome, etc. Therefore, to hinder the spread of COVID-19, a Computerized Doughty Predictor Framework (CDPF) is developed to yield benefits in monitoring the progression of disease from Chest CT images which will reduce the mortality rates significantly. The proposed framework CDPF employs Convolutional Neural Network (CNN) as a feature extractor to extract the features from CT images. Subsequently, the extracted features are fed into the Adaptive Dragonfly Algorithm (ADA) to extract the most significant features which will smoothly drive the diagnosing of the COVID and Non-COVID cases with the support of Doughty Learners (DL). This paper uses the publicly available SARS-CoV-2 and Github COVID CT dataset which contains 2482 and 812 CT images with two class labels $COVID^{+}$ and $COVID^{-}$. The performance of CDPF is evaluated against existing state of art approaches, which shows the superiority of CDPF with the diagnosis accuracy of about 99.76%.

Keywords: Adaptive Dragonfly Algorithm, Auto Augmentation, COVID-19 Predictor, Deep Learning, Ensemble Learning, Image Processing, UNet Segmentation

1. Introduction

A virus is a sub-microscopic agent which is coated with protein and it can be found anywhere like water, air, and soil. It has the potential to cause severe illnesses like smallpox and Acquired Immuno Deficiency Syndrome (AIDS) and also attacks certain cells of the human body such as respiratory systems, liver, and blood [1]. A new kind of coronavirus disease is an infectious disease caused by Severe Acute Respiratory Syndrome CoronaVirus 2 (SARS-CoV-2) which has been found in 2019 in Wuhan, China [2]. It has spread globally either by direct contact with infected people or indirect contact with the tools used by infected people. The major symptoms of COVID-19 include fever with cough, fatigue, and breathing problem, loss of taste and smell, and also leads to severe complications such as Middle East Respiratory Syndrome (MERS) and Severe Acute Respiratory Syndrome (SARS). According to the statement of WHO report, as of July 2020, more than 12 million people have been infected and around 552,050 people are lost their lives [3]. Since the lack of ICUs, many health care sectors even in developed countries failed to save human lives. As per the statement given by the main Disease Control Centers of the Chinese [1], till 9th April 2020, 13.8% of people were identified with severe conditions, and 4.7% people were identified with critical conditions. A higher mortality rate is detected from infected males than females, but a higher death rate is detected in both genders with rising age. Therefore, health sectors all over the world are taking part in the earlier diagnosis of COVID-19 cases to achieve less fatality rate.

Although, manual detection of COVID-19 is time-consuming [2]. A Polymerase chain reaction (PCR) laboratory test result sets the benchmark for recognizing covid-19 infection in the human body. But, the process of PCR takes 4-6 hours [3], to yield the test results which is a lengthy time compared to the time taken by the virus to spread. As a result, the infected person cannot be identified in time and tends to infect others unintentionally. To address the inadequacies present in the current COVID-19 tests, many physicians have started to utilize radiological images such as X-rays or CT scans for accurate diagnosis. As many hospitals and laboratories have X-ray machines, few radiologists suggest chest X-rays for diagnosis through which chest images of patients can be obtained easily. But, CT scans are more advantageous than X-rays with giving a more detailed picture of the patient's condition including soft tissues. So, CT images are the most preferable one among many radiologists. Many research contributors [4-9] have revealed the effectiveness of CT scans in COVID-19 diagnosis.

Therefore it is indispensable to build the computerized paradigm for detecting the COVID-19 disease at an earlier stage with a high precision rate to achieve a lower fatality rate and save the physician's valuable time. Over the decade, AI technologies have been incorporated [10] into medical diagnosis systems because of its extreme capability beyond the human potential in handling enormous datasets. Hence, an AI-based medical diagnosis system facilitates speedy, reliable, and accurate diagnosis [11]. Moreover, AI-based approaches eliminate the limitation faced by PCR test kits, testing cost, and waiting time to yield the test results. Many researchers have experimentally proved that deep learning-based computer-aided diagnosis has achieved remarkable performance in the field of medical science [12] such as classification of skin cancer, brain diseases, lung segmentation, and so on. The reason behind this great success is that the deep learning algorithms automatically learn the features from the given datasets whereas machine learning algorithms are based on extracting the features manually.

Many researchers have endeavored to design an effective classification model for diagnosing the covid-19 infected people from normal people [13-36]. They compete with one another to win the greater precision rate in a shorter time. Because lower false positive rate could help normal people to avoid unnecessary medical treatments which in turn saves medical

resources and helps the infected people to acquire medical treatments in time which reduces the fatality rate considerably. Therefore, greater accuracy with shorter time complexity is considered as the benchmark for measuring the performance of the medical diagnostic system.

To attain the above benchmarks, a Computerized Doughty Predictor Framework (CDPF) is developed for diagnosing the COVID-19 cases. The highlights of the proposed system are as follows:

- The proposed framework makes use of the deep learning approach such as Convolutional Neural Network (CNN) for extracting the features from chest CT images to improve the prediction accuracy.
- The Adaptive Dragonfly Algorithm (ADA) is introduced to capture the most significant features by eliminating the features which do not contribute much to the classification, so the time required to train the model will reduce considerably and also enhances the accuracy of classifiers.
- Ensemble classifiers are used in the name of Doughty learners to make a strong decision on the given sample which lowers the error rate.

The rest of the paper is organized as follows: Section II depicts the survey of existing art of approaches; Section III presents the detailed description of the proposed framework; Results and discussions are provided in Section IV. Finally, the conclusion and future enhancements are given in Section V.

2. Related Works

The recent novelties with the exploitation of AI technologies that exist in the field of designing Computer-Aided COVID-19 Diagnosis (CACD) systems have been explored. This exploration further helps the researchers to figure out the bottleneck that resists the performance of existing CACDs and acquire the adequacy to breakout such constraints through designing a powerful paradigm.

Several research works related to diagnosing the COVID-19 disease from CT and X-Ray images are been addressed. In [13], automatic detection of COVID-19 disease based on deep transfer learning from chest X-Ray images has been proposed and it shows an accuracy of about 98%. In [14], Wang et al. have employed the CNN mechanism over X-Ray images with three classes of labels to distinguish the normal, COVID-19, and pneumonia patients, which produces an efficiency of about 92.6%. In [15], Abbas et al. have presented a framework named DeTraC for predicting COVID-19 which is been carried out on Chest X-Ray (CXR) images and it is mainly designed to detect the irregularities in the image with the aid of class decomposition techniques and attained the disease recognition accuracy of 95.12%. Authors in [16] have collectively been involved in designing CNN based medical diagnosis system to help radiologists automatically diagnose people with Coronavirus from CXR images. The classification performance of the proposed system is assessed and obtained more than 66% testing accuracy. In [17], Khan et al. have proposed a deep learning network to automatically detect the people with coronavirus by extracting the essential features from CXR images of 284 COVID-19 cases, 330 pneumonia bacterial cases, 327 pneumonia viral cases, and 310 normal cases. Moreover, the proposed system is achieved 89.5% accuracy in the four classes' scenario. However, CXR images cannot discriminate the soft tissues perfectly that defeats the system performance to achieve 100% accuracy [18]. Thus, many researchers have contributed the advanced framework based on Chest CT (CCT) images rather than CXR images towards achieving greater specificity in non-COVID-19 cases.

A few incredible research works for the prediction of COVID-19 on CCT images have been briefed here. In [19], authors have developed an AI algorithm to detect COVID-19 from CCT images and could attain the specificity of 93%. The authors Barstugan et al. [20] have adopted machine learning methods for COVID-19 classification using 300 CT images and its execution is started with feature extractions followed by SVM-based classifications. The overall performance of the machine learning framework is evaluated and it acquired an accuracy rate of 99.68%. In [21], Li et al. have built the ResNet50 (COVNet) model to distinguish the COVID-19 cases from pneumonia and non-pneumonia cases using 4356 CCT images. The worthiness of the COVNet model is experimentally proved with a specificity of 96%. The authors Gozes et al. [22] have developed a comprehensive system for detecting the coronavirus using CT screening. The system is trained by employing the ResNet 50 network and also employed image processing mechanisms for lung segmentation. The specificity of the proposed system was attained as 98% with an AUC score of 0.9940.

The authors in [23] have demonstrated that UNet++ based training on segmented CT images achieved 98.85% of testing accuracy. In conclusion, the preliminary study has demonstrated that the CT images in the COVID-19 diagnosis system have high specificity [19] with sufficient generalizability and in addition to this, the AI-based diagnosis model could able to obtain the optimum results with a maximum of 99.00%+0.09 accuracy. Moreover, the study in [24] has stated that CT scans are spotting the hazy gray areas in the lungs which are the primary sign of COVID-19, and found that CT Screening has high sensitivity in diagnosing COVID-19. As a result, CT scans are deliberated as a primary resource for monitoring and evaluating the COVID-19 patients with the acute respiratory syndrome. There is a need for analyzing multiple images to monitor the progression of the disease for a patient manually which complicates the earlier detection [24] and failed to attain greater precision. Therefore, an Automated AI-based diagnosis system has come into practice to speed up the screening of many images with a high precision rate and also able to detect the patients even in asymptotic conditions.

The following study has assessed the efficacy of various machine learning and deep learning algorithms applied over the radiological images to resolve the clinical diagnosis issues. According to the statement of the authors in [24], the prime abnormalities of COVID-19 infected patients are Ground-Glass Opacities (GGO), consolidations, and nodules. These features are easily identifiable through deep learning mechanisms. Alqudah et al. [28] have presented the hybrid AI systems with three classification models such as SVM, CNN, and Random Forest, and their system performance is achieved up to 95.2% accuracy with 93.3% sensitivity. In [30] Tang et al. and Farid et al. [31] have proposed the framework to predict the severity of COVID-19 disease in Chest CT images by extracting the quantitative features. The feature extractions were carried out with the help of Machine Learning algorithms and yielded detection accuracy of 87.5% and 96.07%.

The authors of [29] have attempted to build a novel detection model based on GAN and deep transfer learning. They have utilized three pre-trained models for classifications like AlexNet, GoogleNet, and RestNet18. The experimental results inferred that the RESTNet 18 is the major influencing mechanism to yield a greater accuracy rate. Jelodar et al. [25] and Yan et al. [26] have built the model based on LSTM to predict the COVID-19 confirmed cases. But they were unsuccessful to solve the problem of data irregularity leads to a greater misclassification rate. Sedik et al. [27] have taken the steps further to resolve data irregularity by deploying deep learning mechanisms namely CNN and LSTM over the augmented datasets. The augmentation includes rotation, flipping, and resizing as well as data augmentation based on conditional generative adversarial networks (CGANs), and their system efficacy is

evidenced with an average accuracy of 95%. The authors of [37] have applied the data set of augmented images on RestNet18 architecture to locate the abnormalities lies in the CT scans and its demonstration shows an accuracy of 99.4%. In [32], authors have proposed a hybrid framework in which VGG16, ResNet50, DenseNet121, and InceptionResNetV2 were utilized for extracting deep features from the chest images. Further, the extracted features are fed into different ML classifiers for final prediction and achieved 87.9% accuracy.

The Light CNN framework which is mentioned in [33] has shortened the training time with less classification accuracy whereas the works mentioned in [34] and [35] have greater recognition accuracy with high time complexity. The feature extraction mechanism may generate redundant and irrelevant features in feature space subjected to high computational overhead. Hence it necessitates the feature selection mechanism [3] which does not only enhances the classification accuracy but also minimizes the computational overheads. In [36], Singh et al. [36] have proposed a three-stage mechanism for COVID-19 diagnosis. In, first stage, deep transfer learning architecture based on VGG 16 was employed as a feature extractor. Then, the extracted features were selected by applying Principal Component Analysis (PCA). Finally, four different classifiers were used to discover the COVID-19 patients from normal people and they yielded the best accuracy of 95.7%.

The following key points have been derived from the above study and taken into the consideration in the proposed framework.

- CT screening plays a major role in COVID-19 diagnosis since it delivers high sensitivity and specificity rate.
- Automated AI-based diagnosis is desirable to replace the manual process which in turn speeds up the analyzing huge datasets.
- A deep learning-based diagnosis system can effectively recognize the prime abnormalities of COVID-19 in CT scans.
- However, Some DL-based approaches were slow and produced less recognition accuracy due to data irregularity.
- To overcome the limitations of DL-based approaches, Feature Selection (FS) method can be employed to eliminate the irrelevant features which enhance the classification accuracy and minimize the computational overhead.
- Before Feature Extraction, data augmentation techniques can be used to build a deep learning model with reduced overfitting [38].
- Ensemble classifiers can be applied to improve the accuracy of prediction further through aggregating the decisions of multiple base classifiers [39], [40].

3. System Materials and Methodologies

The Proposed framework (CDPF) is a Multi-phase process as shown in **Fig. 1**. Initially, the framework has been nourished with CT images which are collected from publically available SARS-CoV-2 and Github COVID CT datasets. Then, auto augmentation policy has been applied over the collected dataset which helps to improve the generalization performance of deep learning architecture by lessen the distance between training sets and validation set as well as any future testing sets. Besides, the augmented data sets are further enhanced by employing preprocessing mechanisms which involve noise removal (BLPF) and contrast enhancement (HE). Subsequently, the preprocessed images are subjected to UNet Architecture to perform semantic-based segmentation for separating the Region of Interest (ROI). Further, Convolutional Neural Network is used as a feature extractor to extract the features from the segmented parts. The resultant feature sets are further applied to the Adaptive Dragon Fly

algorithm to select the optimal feature sets. Finally, the optimized feature set is used for distinguishing COVID-19 cases from normal cases by employing Ada Boost Doughty Learners.

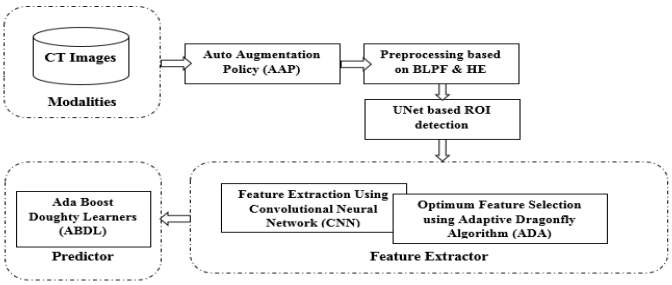

Fig. 1. Methodologies of CDPF framework

3.1 Dataset Description

The publicly available SARS-CoV-2 CT Scan dataset (Dataset 1) used in this research contains 2482 CT images with two class labels such as COVID+ and COVID. Similarly, open-source COVID CT-dataset (Dataset 2) from Github is also used for experimenting with the proposed framework to show its diagnosis effectiveness which has 300 COVID-19 CT images and 512 non-COVID-19 CT images. Dataset2 has been subjected to Auto Augmentation Policy which artificially expands the given dataset [41] for increasing the generalization performance of the proposed architecture.

3.2 Auto Augmentation Policy (AAP)

This policy automatically selects the suitable augmentation policy for the given dataset. In Search Space (S), a policy has several sub-policies which has three tuples $\{\mathcal{T}(t), \mu_t, \mathcal{P}_t\}$ where $\mathcal{T}(t)$ describes the transformation operations to be applied over the input image, Magnitude (μ_t) and Probability $(\mathcal{P}_t)$ of using the operations in each batch. Totally, 16 transformation operations are used in our search space and the range of μ_t is [0 to 9] and the range of $\mathcal{P}_t$ is [0, 0.1… 1]. Further, Search Algorithm is used to find better data augmentation policies based on higher validation accuracies. The following steps are iteratively performed in AAP as shown in **Fig. 2**. Finally, the best sub-policies are concatenated to yield the best augmentation policy for the given dataset.

- For every image in the dataset, sub policy is randomly chosen and applied to generate the transformed images.
- The resultant images are further used to train the child model with a learning rate of 0.00035 and weights of the controllers are in the range of (-0.1, 0.1).
- The child models are trained based on Recurrent Neural Network (RNN).
- The validation sets are used to evaluate the accuracy (R) which in turn act as a reward signal to train the RNN controller.
- The controller assigns a low probability for poor child models and a high probability for good child models which helps to determine the best sub policy.

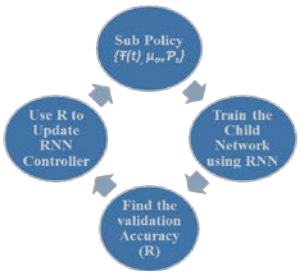

Fig. 2. Selection of best augmentation method using AAP

3.3 Image Preprocessing

The image preprocessing mechanism plays a vital role in medical image-based diagnosis. In the proposed framework, the Contrast Enhancement technique namely Histogram Equalization (HE) is applied to remove the uncertainties present in the captured image. The uncertainties are in the form of blurry images, shadow images, and inexact gray levels which results in an incorrect diagnosis. Initially, HE tries to find the histogram and local minima of the given image. Then, Histogram is divided based on the local minima. Finally, HE is applied over each partition. Before this, Bi-Lateral Low Pass Filter (BLPF) is applied to remove the noises and outliers present in the captured images. It is a nonlinear filter that preserves the sharp edges since the intensity of each pixel is replaced by the weighted average of intensity values of neighborhood pixels. The weight does not only depend on the distance of the pixels but also depends on the pixel intensities, depth distance, and so on. In BLPF, weights can be calculated based on Gaussian distribution.

3.4 UNet based ROI Detection

The major imaging features of CT Scans such as ground-glass opacity and occasional consolidation plaques in the lungs enable radiologists for diagnosing COVID 19 and evaluating the severity level of patients. It is a challenging task to automatically segment the lesions of COVID 19 because of their different characteristics appearances such as ground-glass opacity and consolidation plaques; irregular shapes and fuzzy boundaries; fewer lesions have very lower contrast when compared to their surrounding areas; moreover, it is a difficult and time-consuming task to mark the pulmonary infection in artificial [42]. It necessitates developing a deep learning approach to automatically segment the COVID 19 infected Areas (ROI) from CT chest images by extracting rich features**.** In this research, UNet architecture is used for image segmentation which performs semantic-based segmentation from which ROI can be detected. It uses the concept of de-convolution and uses shortcut connections among the encoder and decoder which provides high-resolution features to the de-convolution layers that leads to more precise segmentation of ROI which results in greater accuracy and it is experimentally proved and shown in the section 4.

The UNet architecture has two paths such as at the left side named contracting path to obtain context information and at the right side named expansive path to recover feature map shown in **Fig. 3**. The contracting path follows the architecture of a convolutional network where performing two 3*3 convolution processes, each followed by Rectified Linear Unit (ReLU) and 2*2 max-pooling operations with stride 2 for downsampling. In the process of

downsampling, the number of feature channels are doubled from 64 to 256. In contrast, the expansive path involves upsampling followed by 2*2 convolution which minimizes the number of feature channels to half from 256 to 64, and performing two 3*3 convolution processes, each followed by Rectified Linear Unit.

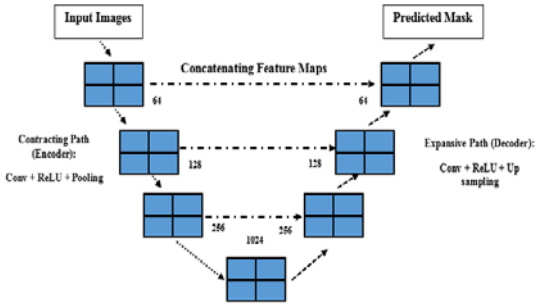

Fig. 3. UNet based ROI detection

At each step in the expansive path, the high-resolution feature maps of the contracting path are concatenated with the output of the upsampling process to achieve localization and obtain the accurate output. Finally, the 1*1 convolution process is performed at the final layer to map the resultant feature maps into the required number of classes. Now, the UNet architecture is trained by feeding the input images and their respective segmentation maps with the help of the soft-max function as given in (1).

$$p_c(x) = e^{a_c^{(x)}} * \frac{1}{\sum_{k=1}^{K} e^{a_c^{(x)}}} \tag{1}$$

Where $a_c^{(x)}$ denotes the activation in the feature channel *'c'* at the pixel *'x'* and *'K'* is the no of classes. The value of $p_c(x)$ is 1 for the classes *'K'* that has maximum activation $a_c^{(x)}$ and 0 for all other *'K'*. Then cross-entropy *(E)* is calculated using (2) to correct the deviation of the function $p_l^{(x)}$.

$$E=\sum_{N=\{1....K\}} \omega(x) \log p_t^{(x)}(x) \tag{2}$$

Where *'t'* denotes the truth label of each pixel *x* and *'ω'* denotes the weight map which is pre-computed using (3) to segment the ground truth image and force the network to learn the small separation borders. Then morphological operators are applied to compute the separation borders.

$$\omega(x) = \omega_f^{(x)} + \omega_0 * e^{\frac{-(g_1^{(x)}+g_2^{(x)})^2}{2*\alpha^2}} \tag{3}$$

Where $\omega_f^{(x)}$ denotes the weight map and $g_1^{(x)}$ and $g_2^{(x)}$ represent the distance to the border of the nearest cell and second the nearest cell respectively and the values of ω_0 and α set to 10 and 5 pixels respectively.

3.5 Feature Extractor as Convolutional Neural Network (CNN)

Many researchers have evidenced the efficacy of CNN architecture in many applications especially in image classification and medical image analysis. The major reason behind this success is that the CNN is capable of obtaining local features from the higher layer inputs and transfer them into the lower layers [3] for extracting complex features. But still have a strong prediction against COVID-19 cases, it needs to be trained using large datasets. Due to the unavailability of accurate and large public datasets, the proposed work uses CNN to extract the powerful features rather than the prediction.

Table 1. Parameters of CNN

Layers	K	K_s	S_s	Learnable Parameters (LPs)
CoOp_1	64	3 by 3	1 by 1	1792
MaP_1	-	2 by 2	2 by 2	-
CoOp_2	64	3 by 3	1 by 1	36928
MaP_2	-	2 by 2	2 by 2	-
CoOp_3	32	3 by 3	1 by 1	18464
MaP_3	-	2 by 2	2 by 2	-
CoOp_4	16	3 by 3	1 by 1	4624
MaP_4	-	2 by 2	2 by 2	-
CoOp_5	8	3 by 3	1 by 1	1160

The architecture of CNN comprises the input layer, several hidden layers, and output layer as shown in **Fig. 4**. The hidden layer subsequently performs the convolution operations *(CoOp)* to generate the feature maps followed by Maximum Pooling *(MaP)* operations for feature reduction. These set of operations are repeated 5 times in the proposed framework with varying numbers of filters (K) and with an ideal filter size (K_s) of 3 by 3 at each time as shown in **Table 1**. All convolution layers use the strider of size (S_s) 1 by 1 whereas all pooling layers use the strider of size 2 by 2. Similarly, ReLU (Rectified Linear Unit) is used as an activation function of all neurons in the network for increasing the nonlinearity in feature maps. The input layer of the architecture considers the size of the image is 224 by 224 and the total Learnable Parameters (LPs) of this architecture are 62968 which is earned using (4). From the last layer, a total of 800 features have been extracted from each input image.

$$\text{LPs}=(h(K)*w(K)*n(KP)+1)*n(KC) \quad (4)$$

Where $h(K)$ represents the height of filter, $w(K)$ represents the width of filter, $n(KP)$ specifies the number of filters in previous layer and $n(KC)$ denotes the number of filters in current layer.

Moreover, in CNN, kernels are often called as feature identifiers which are very useful in identifying specific features. The kernels are initialized with specific weights (wt) and weights of the kernels are iteratively updated using (5) based on the simplest update rule described in Stochastic Gradient Descent approach which adjusts the weights to minimize the loss.

$$wt=wt(i)-l(r)*g(r) \quad (5)$$

Where wt(i) denotes the learnable weight with an initial value which is a random number in the range of [0.0, sqrt(2/n)] in which 'n' denotes the number of inputs, l(r) is a learning rate with the value of 0.0001, g(r) is the gradient which is the partial derivative of loss function (∂L) with respect to learnable weight (∂(wt)) and L is the Mean Squared Error based loss function.

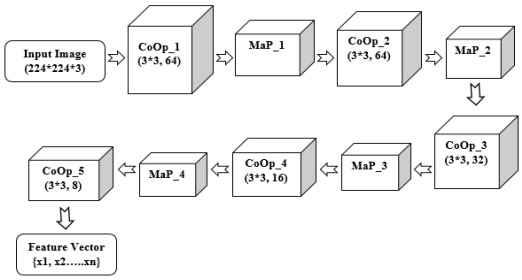

Fig. 4. CNN architecture

3.5.1 Optimum Feature Selection using Adaptive Dragonfly Algorithm (ADA)

The resultant feature sets may have irrelevant and redundant features which increase the misclassification rate and introduces the computational overhead. To address the above-said challenges, the Optimum Feature Selection (OFS) strategy is employed to eliminate irrelevant features which do not contribute much to the classification process. The Dragonfly Algorithm (DA) is a Meta-Heuristic algorithm that uses dynamic and static swarming behaviors to explore the search space and determine the optimum solution for the given problem. It also uses three principles of swarming as Separation, Alignment, and Cohesion as described in [45] and [46]. But in DA, the position of each dragonfly is updated based on the step vector, and the position vector leads to premature convergence [47] because the dragonflies are not allowed to keep track of the previously obtained solutions. Moreover, in DA, swarming factors can be adjusted by fixed parameters cause imbalance in local and global search. To address the above challenges, the authors of [47] have incorporated the following features into the typical DA which is used as Adaptive Dragonfly Algorithm (ADA) in the proposed framework for optimum feature selection.

- Pbest and Gbest variables are introduced to obtain the local best fitness value of dragonfly and global best fitness value.
- At each iteration, the fitness value of dragonfly is compared with current Pbest and if fitness value is better than current Pbest, then Pbest is assigned to fitness value.
- Similarly, the best fitness value obtained yet by all dragonflies is saved to Gbest.
- Dynamic curves are employed to tune the parameters of swarming factors.
- At the earlier stage, the ADA searches for huge search space to avoid premature convergence.
- At the later stage, to refine the final solutions, small regions are exploited.

In optimum feature selection process, ADA performs the following steps sequentially as shown in **Fig. 5**.

- Initially, the subset of features from the given set is selected based on the mRMR principle (Maximum Relevancy Minimum Redundancy). The selection is based on the value of probability ‘p’. If value of ‘p’ is greater than ‘ε’, then that feature is selected for ADA algorithm. Thereby, all top ranking features are being selected.
- Then selected subsets (S’) are subjected to ADA as initial set of populations to generate the promising candidate set.
- Now ADA algorithm begins to execute by initializing the parameters {t, Max_iter, Pb, Gb and E}.

- For Every Iteration, Fitness value for selected subset is calculated using SVM classifier.
- Subsequently Pbest (Pb) and Gbest (Gb) values are updated based on fitness value.
- The weights of swarming factors such as s(ω), a(ω), c(ω) and e(f) are updated using the dynamic curve f(t). The values of those parameters are decreased adaptively based on f(t).
- Then, values of Separation (S), Alignment (A), Cohesion (C), Food Source (F) and Enemy (E) are updated according to the equations given in Algorithm 1.
- Each dragon fly update its position vector Δ(X) using the formula described in Algorithm 1.
- The above mentioned steps are iteratively performed until the value of 't' reaches maximum iteration.
- Then, candidates with best fitness value are considered as the optimum features for the COVID-19 diagnosis.

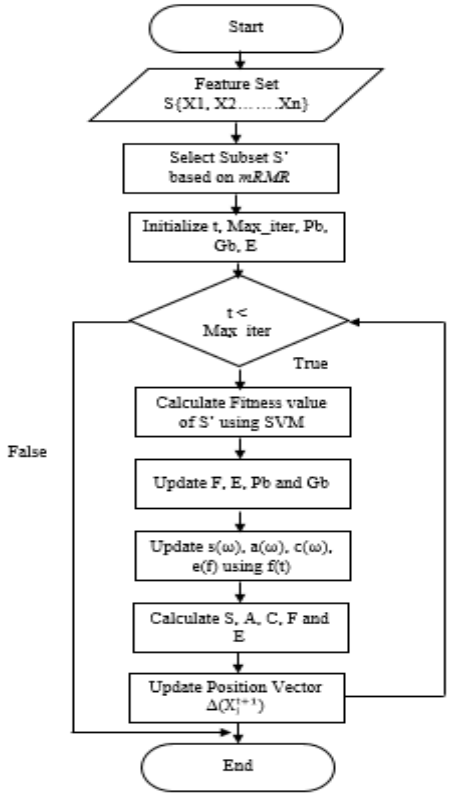

Fig. 5. Optimum feature selection using ADA

Algorithm 1 OFS using ADA

Input: Feature Set S{X1, X2……Xn}
Output: Optimum Feature Set Opt(S)
Begin
Calculate
$mRMR = \frac{1}{M}((\sum_{i=1}^{M} Rl(S(i), C) + Rl(R(m), C) - \frac{1}{M}(\sum_{i=1}^{M} Rd(S(i), R(m))$
Select Subset S' based on the probability of mRMR feature sets by

$p=-(1+r)*\frac{arctan(2*\frac{mRMR(X)}{f})}{\pi}+1$

Binary values are assigned to each feature based on the value of ε.

X(i)=1 if p> ε; 1≤i≤n

X(i)=0 otherwise

$\varepsilon= 1\text{-}0.5*r*\frac{1}{f}$

Initialize the Dragonflies to create population S'={X1, X2…..Xn} where features with value '1'

While t<Max_iter

Calculate the Fitness value of position vectors of dragonflies

Update food source, enemy, Pbest and Gbest

Update parameters s(ω), a(ω), c(ω) and e(f) by

$f(t)= Init(1-\frac{1}{1+e^{(-0.1*t+5)}})$

Calculate S, A, C, F, and E by

$S_i = -\sum_{k=1}^{N} X - X_k$

$A_i = \frac{\sum_{k=1}^{N} V_k}{N}$

$C_i = \frac{\sum_{k=1}^{N} X_k}{N}\text{-}X$

$F_i=F^{+}\text{-}X$

$E_i=E^{+}\text{-}X$

Update Position Vectors by

$\Delta(X_i^{t+1}) = (s(\omega)S_i^t + a(\omega)A_i^t + c(\omega)C_i^t + f(\omega)F_i^t + e(f)E_i^t)+wX_i^t+C_1r_1(Pb_i^t - X_i^t)*ln(\frac{1}{u})+ C_2(1\text{-}r_1)(Gb\text{-}X_i^t)$

$Pb_i^t = \frac{1}{I}\sum_{i=1}^{I} p_i^t$

Stop

Table 2. Nomenclature of ADA

S.No	Terms	Description
1	M	No. of Selected Features
2	S	Selected Features
3	C	Class Labels
4	R	Remaining Features
5	Rl	Relevancy between labels and features
6	Rd	Redundancy between two features
7	t	Iteration
8	*s(ω)*	Separation Weight initialized to 0.1
9	*a(ω)*	Alignment Weight initialized to 0.1
10	*c(ω)*	Cohesion Weight initialized to 0.7
11	*e(f)*	Enemy Factor initialized to 1
12	*S*	Separation
13	*A*	Alignment
14	*C*	Cohesion
15	*F*	Food Source
16	*E*	Enemy
17	*X*	Current Position of

		Dragonfly
18	X_k	Position of kth Neighbor
19	V_k	The velocity of kth Neighbor
20	N	Neighborhood Size
21	F^+	Position of Food Source
22	E^-	Position of Enemy
23	$f(\omega)$	Food Source Weight initialized to 1
24	w	Inertia Weight initialized to 0.9
25	Pb	Personal Best
26	Gb	Global Best
27	C_1, C_2	Cognitive and Social Parameters set to '2'
28	$r1$ *and* u	Random Values assigned in the range (0,1)
29	I	No. of Instances
30	f	No. of Features
31	$mRMR(X)$	Feature Set X obtained by mRMR

3.6 Predicting COVID-19 cases based on Ada Boost Doughty Learners (ABDL)

In the proposed framework, prediction is made based on the Ensemble Learners to improve the accuracy of the framework. The major reason for choosing the ensemble culture is that the decision of multiple classifiers will be better and effective than the single classifier. It is a two-step process in which a committee of learners is constituted at the first stage followed by fusing the decision of each learner to make a doughty and final prediction. The fusion is carried out in several ways like weighting methods and meta-learning methods. The proposed method utilizes the concept of the Adaptive Boosting process to fuse the decisions which is a weighted and iterative method. The classification ability of the Ada Boost method is significantly higher than other learners [48] since at every iteration it concentrates on more complex samples by reducing the weight of correctly classified samples and increase the probability of selecting the misclassified samples. Likewise, Ada Boost Doughty Learners (ABDL) concentrates on handling the more informative and complex samples by making subsequent learners correct the mistakes of its predecessor.

In ABDL, Doughty Learners are of Support Vector Machine (SVM), Naïve Bayes Classifiers (NBC), and Decision Tree (DT). Each classifier is subsequently executed for a predefined number of iterations as shown in **Fig. 6**. At the beginning, SVM classifier starts to execute with the input samples and initial weights. At the end of its execution, initial weights are updated for each input samples based on the prediction. Subsequently, NBC starts its execution by receiving the updated weights which are generated by SVM Classifier by which the NBC Classifier attempts to correct the mistakes of SVM Classifier. Then, the third learner DT classifier starts its execution by using the updated weights of NBC Classifier, Finally, prediction over the given sample is accomplished by fusing the decision of all doughty learners.

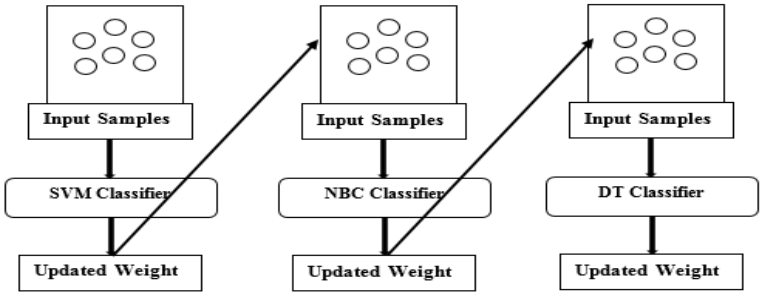

Fig. 6. Prediction based on ada boost doughty learners

The working method of ABDL is described as follows:

Step 1: Initially, equal weights are assigned to each input sample using the given formula.

$$\omega_t(i)=\frac{1}{N} \tag{6}$$

Where $\omega_t(i)$ represents the weight of the i^{th} sample in iteration 't' and 'N' represents the number of input sample.

Step 2: Now, Input samples with weights are subjected to the SVM classifier to perform prediction on the input samples.

Step 3: At the end of every iteration, the significance of the classifier is calculated by summing up the weights of every misclassified samples as shown in the given formula.

$$\check{S}_t(l)=\frac{1}{2}\log\left(\frac{1-sum(\omega_t^{(m(i))})}{sum(\omega_t^{(m(i))})}\right) \tag{7}$$

Where $\omega_t^{(m(i))}$ represents the weight of i^{th} misclassified sample in iteration 't' and $\check{S}_t(l)$ denotes the significance of Doughty learner 'l' in iteration 't' .

Step 4: At every iteration, the weights of correctly classified samples $\omega_t^{(c(i))}$ and misclassified samples $\omega_t^{(m(i))}$ are updated using (8) and (9). Through this, misclassified samples are associated with larger weights and correctly classified samples are associated with smaller weights.

$$\omega_t^{(c(i))}=\omega_{t-1}^{(c(i))}*e^{-\check{S}_t^{(l)}} \tag{8}$$

$$\omega_t^{(m(i))}=\omega_{t-1}^{(m(i))}*e^{\check{S}_t^{(l)}} \tag{9}$$

Step 5: The weights of all samples are normalized by using the given formula.

$$\omega_t^{(i)}=\frac{\omega_t^{(i)}}{sum(\omega_t^{(i)})} \tag{10}$$

Step 6: The steps 3 to 5 are executed repeatedly until reaches the maximum iteration with set of input samples. At this point, the weights of the input samples are taken from the previous iteration.

Step 7: Then the steps 3 to 6 are repeatedly executed by using other doughty learner NBC and then using DT with the input samples and updated weights of previous doughty learners. Thereby, subsequent learners are trying to correct the mistakes of its predecessor.

Step 8: The final decision over the given sample $þ_f(i)$ is made by summing up the decision of each learner.

$$cum(\omega(i))=\sum_{l=1}^{3} þ_l^{(i)} * \check{S}(l) \tag{11}$$

Where $cum(\omega\,(\mathrm{i}))$ represents the cumulative weight of the i[th] sample and $þ_l^{(i)}$ specifies the class label of i[th] sample by learner 'l'. Value of $þ_l^{(i)}$ is '1' if the class label is positive and the Value of $þ_l^{(i)}$ is '-1' if the class label is negative.

Step 9: The final decision of i[th] sample $þ_f(i)$ is made based on the value of $cum(\omega(i))$. If the cumulative weight is positive, then $þ_f(i)$ is COVID-19 + . Otherwise, $þ_f(i)$ is COVID-19.

4. Experimental Evaluation and Discussions

The proposed framework is experimentally evaluated by the datasets (DS) with the ratio of 60:20:20 which are described in **Table 3**. The dataset_2 (D2) is augmented artificially with the help of an auto augmentation policy because it poses the problem of data imbalance. That is the count of the positive case is very fewer than the negative case which suffers the proposed model during the training phase. So, the auto augmentation policy facilitated the training phase by extending the positive case count of 300 in D2 to 1458 in D3 and the negative case count of 512 in D2 to 1464 in D3 to balance the class labels and improved the validation accuracy.

Table 3. Datasets

Datasets	No. of COVID-19 cases	No. of Non-COVID-19 cases
Dataset_1 (D1)	1252	1230
Dataset_2 (D2)	300	512
Augmented Dataset_2 (D3)	1458	1464

The policy learned from CIFAR-10 data is used in the experiment. Totally 25 sub-policies are used randomly to augment the dataset which performs two transformation operations over the input image in order as described in [50]. The auto augmentation policy is carried out using the PIL library in python. The original image and its augmentation are shown in **Fig. 7**. For example, the first sub-policy performs sequential operations of Invert followed by contrast. The invert operation does not use any μ_t information but its $\mathcal{P}_t$ is 0.1. Subsequently, contrast is applied which has $\mathcal{P}_t$ of 0.2 and μ_t of 6. Likewise, all 25 sub-policies are randomly chosen to augment the dataset. In the experiment, the dataset has been split into 60%, 20%, and 20% for training, validation, and testing respectively that is 489, 496 and 496 samples of D1, 487, 162 and 162 samples of D2 and 1753, 584 and 584 samples of D3 are been used for training, validation and testing the model respectively. In addition, during the training phase, the data set is split into batches where the batch size is 20 and the number of training epochs is 100. To improve the diagnosis accuracy of the proposed framework further, the datasets are subjected to BLPF for noise removal and subsequently applied the HE for contrast enhancement. Subsequently, the preprocessed images are fed into UNet Architecture to detect the Region of Interest (ROI) for facilitating the further process. Both BLPF and HE approaches are carried out in python with the help of matplot, cv2, and skimage libraries. The resultant

sample images are shown in **Fig. 8**.

The resultant images are subjected to CNN architecture to extract the features from which around 800 features are extracted from each input image. The resultant feature set is optimized to shorten the computational overhead by reducing the feature dimensions from which 221, 160, and 233 features are obtained from D1, D2, and D3 respectively. Further, ABDL algorithm is executed to classify the COVID-19^{+} and COVID-19^{-} cases from the given datasets D1, D2, and D3 with greater accuracy. The steps used in the CDPF are illustrated in Algorithm 2. The Experiment is carried out with Python and Keras API of Tensorflow 2 on Ryzen7-3.20GHz Processor, Graphical Processing Unit (GPU) NVIDIA RTX 3060 with 6GB and 16GB RAM respectively. The performance of the proposed framework is evaluated by the metrics such as Accuracy *($\tilde{a}_{ç}$)*, Sensitivity *($\hat{s}_{\eta}$)* or Recall *($\check{r}_{c}$)*, Precision *($Þ_{\acute{r}}$)*, Specificity ($\hat{s}_{\beta}$) or True Negative Rate ($f_{\eta r}$), False Positive Rate ($f_{\beta r}$), True Positive Rate ($f_{\beta r}$), F1-Score (F^{1}_{Score}), Receiver Operator Characteristic *(ŖôC)* and Training *(t_{r})* and Testing *($t_{\acute{s}}$)* Time of proposed framework which are described in **Table 4**, and the results are tabulated using 5-fold cross-validation technique.

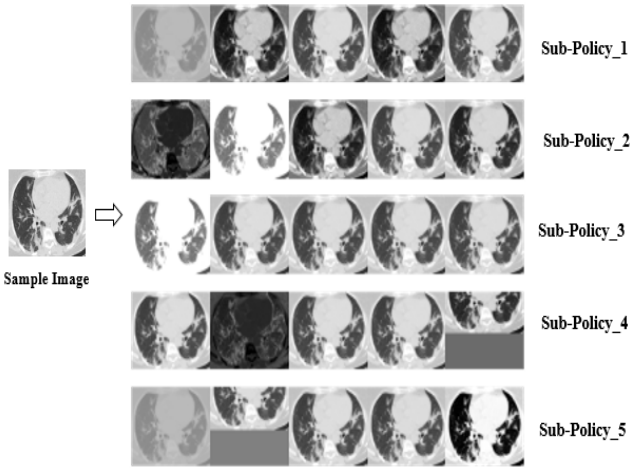

Fig. 7. Auto augmentation using CIFAR-10 policy

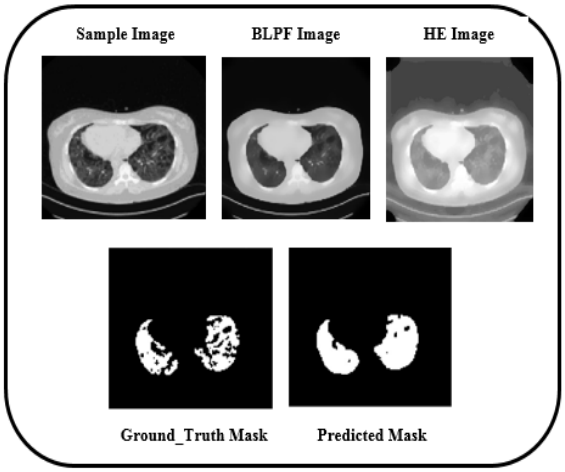

Fig. 8. Image processing mechanism BLPF_HE_Unet

Algorithm 2 Steps used in CDPF

Input: Datasets DS {D1, D2}
Output: prediction of COVID-19 $^{+}$ and *COVID-19* $^{-}$ cases.
Begin
Do
For each image in D2
Apply Auto Augmentation Policy and generate augmented dataset D3.
End
Append D3 to input datasets DS {D1, D2, D3}
Do
For each image in DS
Execute BLPF then perform HE and Generate preprocessed datasets DS_P
End
Do till reaches Max_Iteration
Apply UNet based segmentation on each image in DS_P
End
Then apply CNN on DS_Seg for feature extraction FS {X1, X2….Xn)
Then apply ADA to get reduced feature set
FS_R {X1, X2…..Xn}
Split DS_Seg into train, test and validation set
Do till reaches Max_Iteration
Execute ABDL with DS_Seg and FS_R for predicting COVID-19 $^{+}$ and *COVID-19* $^{-}$ cases.
End
Stop

Table 4. Evaluation metrics

Metrics	Formula	Description
$\tilde{a}_{ç}$	$\frac{Tƥ + Tƞ}{Tƥ + Tƞ + Fƥ + Fƞ}$	Tƥ- True Positive Tƞ- True Negative Fƥ-False Positive Fƞ- False Negative
$\hat{s}_{ƞ}$ or $\check{r}_{c'}$	$\frac{Tƥ}{Tƥ + Fƞ}$	
$\hat{s}_{ƥ}$ or $f_{ƞr}$	$\frac{Tƞ}{Tƞ + Fƥ}$	
F^{1}Score	$2 * \frac{Þ\acute{r} * \check{r}c'}{Þ\acute{r} + \check{r}c'}$	
$Þ_{\acute{r}}$	$\frac{Tƥ}{Tƥ + Fƥ}$	
$f_{ƥr}$	$\frac{Fƥ}{Fƥ + Tƞ}$	
ŖôC	$f_{ƥr}$ vs. $f_{ƥr}$	

Table 5. Overall performance of CDPF

DS	$\tilde{a}_{ç}$	$\hat{s}_{ƞ}$	$\hat{s}_{ƥ}$	$Þ_{\acute{r}}$	$\check{r}_{c'}$	F^{1}Score
D1	99.72	99.68	99.76	0.998	0.997	0.997
D2	96.18	93.00	98.05	0.965	0.930	0.947
D3	99.79	99.79	99.80	0.998	0.998	0.998

4.1 Significance of BLPF_HE_UNet in CDPF

The Proposed framework has yielded greater accuracy for diagnosing COVID-19 cases from Non-COVID-19 cases which are experimentally proved and the results are shown in **Table 5**. From **Table 5**, it has proved that the CDPF framework has correctly predicted 6172 samples out of 6216 samples including both positive and negative samples from D1, D2, and D3 with a misclassification rate of 0.7%. The application of Image Processing Mechanism includes BLPF, HE, and UNet based segmentation impacts the performance of the CDPF framework which is shown in **Table 6**.

Table 6. Confusion matrix of proposed CDPF vs. CDPF without BLPF_HE_UNet

Data Sets	Proposed CDPF				CDPF without BLPF_HE_UNet			
	Ƭþ	ƒɳ	ƒþ	Ƭɳ	Ƭþ	ƒɳ	ƒþ	Ƭɳ
D1	50.28%	0.16%	0.12%	49.44%	44.32%	6.12%	7.25%	42.30%
D2	34.36%	2.59%	1.23%	61.82%	28.33%	8.62%	18.84%	44.21%
D3	49.79%	0.10%	0.10%	50.00%	46.48%	3.42%	7.19%	42.92%

From the above table, the proposed framework CDPF with raw data that is without applying BLPF_HE_UNet could predict the cases correctly with Ƭþ of 44.32% and Ƭɳ of 42.30% for D1, Ƭþ of 28.33% and Ƭɳ of 44.21% for D2 and Ƭþ of 46.48% and Ƭɳ of 42.92% for D3. When applying BLPF_HE_UNet in CDPF, it has improved the accuracy by reducing the misclassification rate of the proposed model from ƒɳ (6.12%) and ƒþ (7.25%) to ƒɳ (0.16%) and ƒþ (0.12%) for D1, from ƒɳ (8.62%) and ƒþ (18.84%) to ƒɳ (2.59%) and ƒþ (1.23%) for D2, from ƒɳ (3.42%) and ƒþ (7.19%) to ƒɳ (0.10%) and ƒþ (0.10%) for D3. The ROI detection through segmentation is the major influencing factor in determining the performance of the medical diagnosis system [42], [43]. When absence of segmentation in the diagnosis system, it greatly decreases the diagnosis ability of the proposed model from 99.72% to 86.62% for D1 as given in **Table 6**, because it needs to examine the entire lung image rather than ROI.

4.2 Effectiveness of Auto Augmentation Policy

Then, the further comparison is made to show the effectiveness of the auto augmentation technique in the proposed framework CDPF. The purpose of employing AAP in the proposed framework is to introduce the sense of generalization in the model results free of overfitting. The implementation of AAP has extended the number of samples of D2 by geometric distortions and balanced the size of D2 as given in **Table 3**.

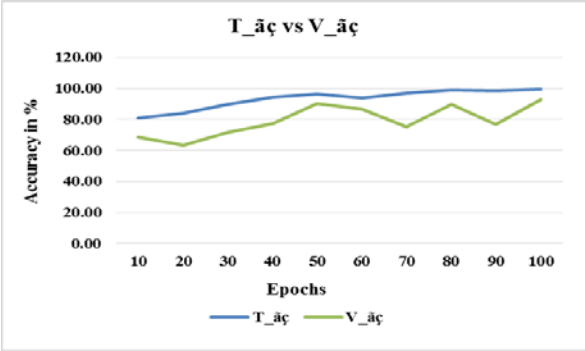

Fig. 9. T_ ãç vs. V_ãç of D2

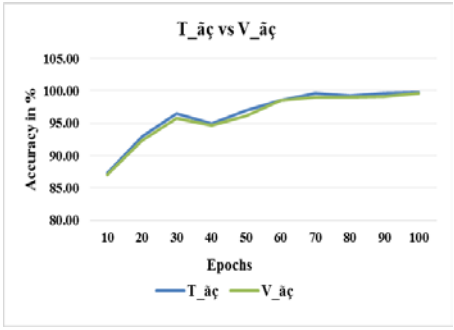

Fig. 10. T_ ãç vs. V_ãç of D3

In **Fig. 9**, the chart is drawn by measuring the Training accuracy *(T_ ãç)* and validation accuracy *(V_ ãç)* of the proposed framework CDPF for D2 at each epoch from 1 to 100. The dataset D2 tends the proposed model to overfit because of possessing a fewer number of samples and imbalance among class labels. So, D2 has increased the distance between training and validation results as shown in **Fig. 9**. At the 100th epoch, the values of *T_ ãç* and *V_ ãç* are 99.59% and 92.59% respectively. Moreover, the auto augmented dataset D2 named D3 has resolved the overfitting problem by increasing both training and validation accuracy up to 99.71% and 99.66% respectively and also shortened the distance between *T_ ãç* and *V_ ãç* as shown in **Fig. 10**.

4.3 Potential of Adaptive Dragonfly Algorithm in CDPF

In addition, the performance of ADA is quite notable since it has reduced the dimension of features from 800 to 221, 160, and 233 respectively for the datasets D1, D2, and D3. The reduced dimension is not only increasing the classification accuracy as shown in **Fig. 11** but also reducing the Training *(t_r)* and Testing *(t_s)* Time of the proposed framework as shown in **Fig. 12**. In **Fig. 11**, overall accuracy is measured for CDPF without ADA (**ãç**) and CDPF with ADA (ãç_ADA). The accuracy of *ãç_ADA* with an optimized feature set is 2.32%, 1.16%, and 1.96% higher than the accuracy of CDPF without applying ADA for the datasets D1, D2, and D3 respectively. Similarly, Training *(t_r)* and Testing *(t_s)* Time of CDPF_ADA and CDPF have been measured as shown in **Table 7**.

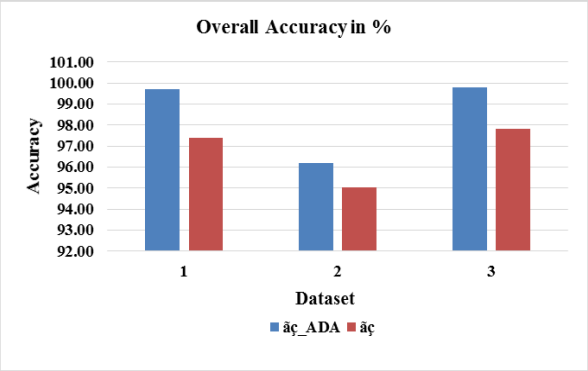

Fig. 11. Comparison of $\tilde{a}_{ç}$_ADA vs. $\tilde{a}_{ç}$

Table 7. Computational efficiency of CDPF_ADA

Samples	AAP	BLPF_HE	UNet	CNN	ABDL_FFS	ADA	ABDL_OFS	ţr of CDPF in sec	ţr of CDPF_ADA in sec
100	18.7	7.01	54.5	16.4	412.4	12.5	46.7	509.00	155.80
200	34.3	12.86	100.1	30.0	839.9	22.9	85.8	1017.12	285.87
300	46.7	15.19	144.1	39.3	1155.5	31.2	113.0	1400.92	389.50
400	49.9	16.62	149.6	41.5	1196.7	33.2	124.6	1454.29	415.47
500	55.0	21.54	161.8	45.8	1412.6	36.7	137.5	1696.67	458.24
600	59.0	24.60	172.2	49.2	1440.1	39.4	147.6	1745.14	492.00
700	68.9	31.57	195.2	60.3	1519.4	45.9	172.2	1875.26	574.00
800	85.5	32.05	249.3	74.8	1594.4	57.0	213.7	2036.00	712.23
900	89.0	33.39	259.7	77.9	1721.4	59.4	222.6	2181.43	741.90
1000	103.9	38.95	302.9	90.9	1657.3	69.2	259.7	2193.97	865.56

The training time is calculated to show the computational efficiency of proposed system which includes the execution time (e_t) of {AAP, BLPF_HE, UNet, CNN, ADA and ABDL}. The values in **Table 7** are calculated using (12) and (13) which demonstrates the potential of ADA in proposed framework. The application of ADA in CDPF yields reduced (optimized) feature dimension of 28%, 20% and 30% for D1, D2 and D3 respectively. Such optimized features shortens the time to train the model as well as reduces the overall computation time of proposed framework as shown in **Table 7**.

$$ţr \text{ of CDPF\_ADA} = e_t(\text{AAP}) + e_t(\text{BLPF\_HE}) + e_t(\text{UNet}) + e_t(\text{CNN}) + e_t(\text{ADA}) + e_t(\text{ABDL\_OFS}) \quad (12)$$

$$ţr \text{ of CDPF} = e_t(\text{AAP}) + e_t(\text{BLPF\_HE}) + e_t(\text{UNet}) + e_t(\text{CNN}) + e_t(\text{ABDL\_FFS}) \quad (13)$$

Where e_t represents the execution time, e_t(ABDL_FFS) specifies the execution time of ABDL with Full Feature Set (FFS) and e_t(ABDL_OFS) denotes the execution time of ABDL with Optimized Feature Set (OFS).

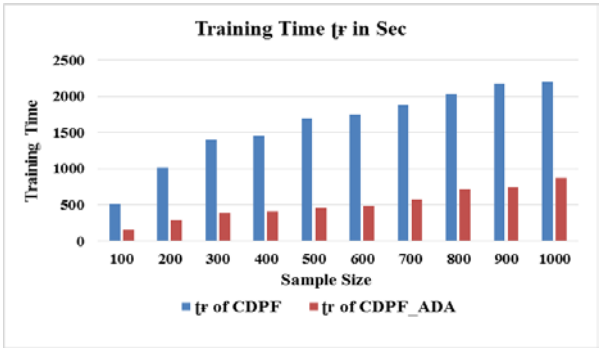

Fig. 12. Comparison of CDPF_ADA vs. CDPF

From **Fig. 12**, it can be observed that the employment of ADA in CDPF has significantly reduced the time required to train the model. It has taken 865.56 s to train the model with 1000 samples which is around 2.5 times faster than the CDPF without applying ADA.

Similarly, the testing efficiency of proposed framework is demonstrated by calculating the testing time (t_s) which includes the execution time of ABDL and excludes the time consumed by AAP, BLPF_HE, UNet, CNN and ADA. The t_s of CDPF in **Fig. 13** represents execution time of ABDL with Full Feature Set and t_s of CDPF_ADA represents execution time of ABDL with Optimized Feature Set (OFS). With the help of ADA, optimized feature sets are fed into the ABDL which makes our proposed model superior with testing time of 12.19s for 700 samples which is around 3.7 times faster than the testing time of CDPF without applying ADA as given in **Fig. 13**.

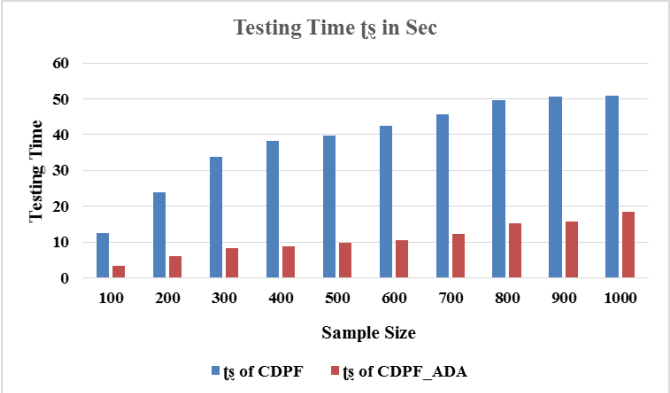

Fig. 13. Comparison of CDPF_ADA vs. CDPF

Furthermore, ŖôĆ curve is used to compare the overall performance of the proposed system. The curve is plotted using the values of False Positive Rate (f_{pr}) and True Positive Rate (t_{pr}). The Area under ŖôĆ curve (AUC) was calculated to be 99.9% and 99.9% for the datasets D1 and D3 respectively as shown in **Fig. 14**. So, the AUC of the proposed framework is near to 1 which denotes it has a good measure of separability among positive and negative cases.

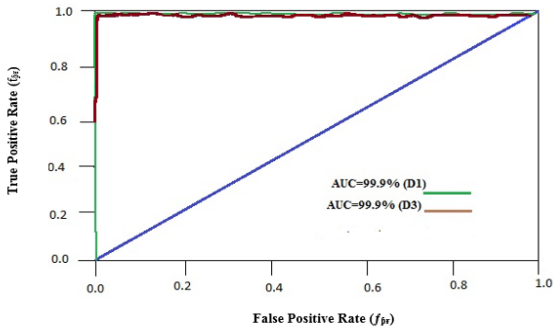

Fig. 14. RôC of CDPF

4.4 Efficiency of Ada Boost Doughty Learners

From the experimental results, it is clearly shown that the combination of Deep Learning and Ensemble Machine learning algorithms has more perfection in diagnosing COVID-19 cases with a reduced misclassification rate. In the proposed CDPF framework, CNN needs to be trained using large datasets to give a strong prediction. So, ensemble learning (ABDL) is used as a predictor and CNN is used as a feature extractor. To show the efficacy of ABDL in the CDPF system, the comparison is made against CNN architecture and all base learners. Among these, the proposed ABDL outperforms all existing approaches as shown in **Fig. 15**.

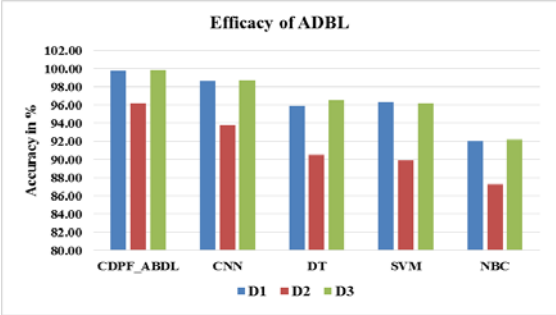

Fig. 15. Efficacy of ABDL

In **Fig. 15**, it is demonstrated that the classification using ABDL has yielded the greater accuracy rate of 99.72%, 96.18%, and 99.79% for the datasets D1, D2, and D3 respectively. The accuracy obtained by ABDL is around 1.5% greater than that of CNN architecture. The performance of DT and SVM classifiers seems the same for all datasets but they degrade the accuracy rate is up to 4.31%. Likewise, the NBC classifier could not correctly predict the maximum number of cases than its predecessors. It has obtained an average accuracy rate of 90.53% which is 8.03% lower than the ADBL in the CDPF framework.

Table 8. Comparison of CDPF with Existing Art of Approaches

Author	Architecture	AUC	$\tilde{a}_r$	$\hat{s}_n$	$\hat{s}_p$
Stephanie et al. [19]	Deep Learning based AI	94.9%	90.8%	84%	93%
Li et al. [21]	ResNet50 (COVNet)	96%	-	90%	96%
Tang et al. [30]	ML Algorithms	98%	89%	91%	85.8%
Singh et al. [36]	VGG16+PCA+Bagging	95.8%	95.7%	96.3%	94.8%
Samir et al. [49]	WDNN	-	99.04%	99.11%	-
Islam et al. [3]	CNN+LSTM	99.9%	99.4%	99.3%	99.2%
	Proposed CDPF	**99.9%**	**99.76%**	**99.74%**	**99.78%**

The results summarized in **Table 8**, show the comparative study of current work with some existing works related to the COVID-19 prediction in terms of $\tilde{a}_{ç}$, $\hat{s}_{\eta}$, $\hat{s}_{\beta}$ and AUC. Some of the works [19, 21, 30, and 36] have yielded lower and moderate $\tilde{a}_{ç}$, $\hat{s}_{\eta}$, $\hat{s}_{\beta}$ and AUC in the range of 85.8% to 98%. And some of the works [3] and [49] have attained greater $\tilde{a}_{ç}$, $\hat{s}_{\eta}$, $\hat{s}_{\beta}$ and AUC with the range of 99.04% to 99.9%. The proposed work CDPF has given the good performance with an average range of 99.74% to 99.9% for $\tilde{a}_{ç}$, $\hat{s}_{\eta}$, $\hat{s}_{\beta}$ and AUC. Overall, the result of the proposed framework CDPF is superior compared to all existing art of approaches.

5. Conclusion

As COVID-19 infections are spread rapidly, many countries are struggling to defeat the outbreak of disease. It necessitates detecting every single positive case at an earlier stage. So, many researchers have tried to develop the medical intelligent system by utilizing the potential of DL and ML for diagnosing the COVID-19 cases. In this work, the CDPF model has been proposed for diagnosing COVID-19 cases from CT images. The proposed Model has utilized the segmentation capability of UNet, the feature extraction capability of CNN, and the prediction capability of Ensemble Learners (ABDL). The performance of CDPF is further enhanced by employing ADA to select the most relevant features for accurate predictions. In addition, AAP has been utilized for reducing the overfitting and generalization issues. The proposed model was experimented on three datasets {D1, D2, and D3} and could identify the disease with an accuracy rate of 99.72%, 96.18%, and 99.79%. Although, the experimental work has revealed that the proposed framework works superior compared to other existing frameworks in terms of higher $\tilde{a}_{ç}$, $\hat{s}_{\eta}$, $\hat{s}_{\beta}$ and AUC of 99.76%, 99.74%, 99.78%, and 99.9% respectively. With the hope, the developed framework would be an effective diagnosis mechanism to reduce the workload of typical medical diagnosis with a lower false rate. In future work, the diagnosis ability of the proposed work can be further enhanced to perform a perfect diagnosis from the images comprising multiple disease symptoms. Further enhancement can be made by employing Pre-trained CNN models to extract better features. Finally, the performance of the proposed system can be verified with radiologists.

References

[1] A. Waleed Salehi, P. Baglat, G. Gupta, “Review on Machine and Deep Learning Models for the Detection and Prediction of Coronavirus,” *Materials Today: Proceedings*, vol. 33, pp. 3896-3901, 2020. Article (CrossRef Link)

[2] Sedik, Ahmed & Hammad, Mohamed & Abd El-Samie, Fathi & Gupta, B B & Abd El-Latif, Ahmed, “Efficient Deep Learning Approach for Augmented Detection of Coronavirus Disease,” *Neural Computing and Applications*, 2021. Article (CrossRef Link)

[3] Md. Zabirul Islam, Md. Milon Islam, Amanullah Asraf, “A combined deep CNN-LSTM network for the detection of novel coronavirus (COVID-19) using X-ray images,” *Informatics in Medicine Unlocked*, vol. 20, 2020. Article (CrossRef Link)

[4] Wu J, Wu X, Zeng W, Guo D, Fang Z, Chen L, Huang H, Li C., “Chest CT Findings in Patients With Coronavirus Disease 2019 and Its Relationship With Clinical Features,” *Invest Radiol.*, vol. 55, no. 5, pp. 257-261, 2020. Article (CrossRef Link)

[5] Fang Y, Zhang H, Xie J, Lin M, Ying L, Pang P, Ji W, “Sensitivity of chest CT for COVID19: comparison to RT-PCR,” *Radiology*, vol. 296, no. 2, pp. E115–E117, 2020.

[6] Li Y, Xia L, “Corona virus disease 2019 (COVID-19): role of chest CT in diagnosis and management,” *Am J Roentgenol*, vol. 214, no. 6, pp. 1280–1286, 2020.

[7] Chung M, Bernheim A, Mei X, Zhang N, Huang M, Zeng X, Cui J, Xu W, Yang Y, Fayad ZA, "CT imaging features of 2019 novel coronavirus (2019-nCoV)," *Radiology*, vol. 295, no. 1, pp. 202–207, 2020.
[8] Li M, Lei P, Zeng B, Li Z, Yu P, Fan B, Wang C, Li Z, Zhou, J, Hu S, "Coronavirus disease (covid-19): spectrum of CT findings and temporal progression of the disease," *Acad Radiol*, vol. 25, no. 5, pp. 603–608, 2020.
[9] Long C, Xu H, Shen Q, Zhang X, Fan B, Wang C, Zeng B, Li Z, Li X, Li H, "Diagnosis of the coronavirus disease (covid-19): rRT-PCR or CT," *Eur J Radiol*, vol. 126, p. 108961, 2020. Article (CrossRef Link)
[10] Nour Eldeen M. Khalifa and Mohamed Hamed N. Taha and Aboul Ella Hassanien and Sally Elghamrawy, "Detection of Coronavirus (COVID-19) Associated Pneumonia based on Generative Adversarial Networks and a Fine-Tuned Deep Transfer Learning Model using Chest X-ray Dataset," 2020. Article (CrossRef Link)
[11] Gaál G, Maga B, Lukács A, "Attention U-Net based adversarial architectures for chest X-ray lung segmentation," 2020. Article (CrossRef Link)
[12] LeCun Y, Bengio Y, Hinton G, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–44, 2015.
[13] I. D. Apostolopoulos and T. A. Mpesiana, "Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks," *Phys. Eng. Sci. Med.*, no. 0123456789, pp. 1–6, 2020.
[14] Wang L, Lin ZQ, Wong A, "COVID-net: a tailored deep convolutional neural network design for detection of COVID-19cases from chest X-Ray images," *Sci Rep.*, vol.10, no. 19549, 2020. Article (CrossRef Link)
[15] Abbas A, Abdelsamea MM, Gaber MM, "Classification of covid-19 in chest x-ray images using DeTrac deep convolutional neural network," *Appl Intell*, vol. 51, pp. 854–864, 2021. Article (CrossRef Link)
[16] Rajaraman S, Antani SK, "Training deep learning algorithms with weakly labeled pneumonia chest X-ray data for COVID-19 detection," *medRxiv.*, 2020.
[17] Khan AI, Shah JL, Bhat M, "CoroNet: a deep neural network for detection and diagnosis of covid-19 from chest X-ray images," 2020. Article (CrossRef Link)
[18] Tingting Y, Junqian W, Lintai W, Yong X, "Three-stage network for age estimation," *CAAI Trans Intell Technol*, vol. 4, no. 2, pp. 122–126, 2020.
[19] Harmon, S.A., Sanford, T.H., Xu, S. et al., "Artificial intelligence for the detection of COVID-19 pneumonia on chest CT using multinational datasets," *Nat Commun*, vol. 11, 2020, Article no. 4080. Article (CrossRef Link)
[20] Barstugan M, Ozkaya U, and Ozturk S, "Coronavirus (COVID-19) Classification using CT Images by Machine Learning Methods," 2020. Article (CrossRef Link)
[21] Li L, Qin L, Xu Z, Yin Y, Wang X, Kong B, et al., "Artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT," *Radiology*, vol. 2020, p. 200905, 2020.
[22] O. Gozes, M. Frid-Adar, H. Greenspan et al., "Rapid AI development cycle for the coronavirus (COVID-19) pandemic: initial results for automated detection & patient monitoring using deep learning CT image analysis," 2020. Article (CrossRef Link)
[23] J. Chen, L. Wu, J. Zhang et al., "Deep learning-based model for detecting 2019 novel coronavirus pneumonia on high resolution computed tomography: a prospective study," *Scientific Reports*, vol. 10, 2020. Article (CrossRef Link)
[24] Varalakshmi Perumal, Vasumathi Narayanan, Sakthi Jaya Sundar Rajasekar, "Prediction of COVID-19 with Computed Tomography Images using Hybrid Learning Techniques," *Disease Markers*, vol. 2021, pp. 1-15, 2020, Article ID. 5522729. Article (CrossRef Link)
[25] Jelodar H, Wang Y, Orji R, Huang S, "Deep Sentiment classification and topic discovery on novel coronavirus or COVID-19 online discussions: NLP using LSTM recurrent neural network approach" *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 10, pp. 2733-2742, 2020. Article (CrossRef Link)

[26] Yan B, Tang X, Liu B, Wang J, Zhou Y, Zheng G, Zou Q, Lu Y, Tu W, "An improved method of COVID-19 case fitting and prediction based on LSTM," *Cmc-Computers Materials & Continua*, vol. 64, no. 3, pp. 1473-1490, 2020.
[27] Sedik A, Iliyasu AM, El-Rahiem A, Abdel Samea ME, Abdel-Raheem A, Hammad M, Peng J, El-Samie A, Fathi E, El-Latif AAA, "Deploying machine and deep learning models for efficient data-augmented detection of COVID-19 infections," *Viruses*, vol. 12, no. 7, pp.769, 2020.
[28] Alqudah AM, Qazan S, Alquran HH, Alquran H, Qasmieh IA, Alqudah A, "Covid- 2019 detection using X-ray images and artificial intelligence hybrid systems," *Jordan Journal of Electrical Engineering*, vol. 6, no. 2, pp. 168-178, 2020.
[29] Loey M, Smarandache F, Khalifa NEM, "Within the lack of chest COVID-19 X-ray dataset: a novel detection model based on GAN and deep transfer learning," *Symmetry (Basel)*, vol. 12, pp. 651, 2020. Article (CrossRef Link)
[30] Tang Z, Zhao W, Xie X, Zhong Z, Shi F, Liu J, She D, "Severity assessment of coronavirus disease 2019 (COVID-19) using quantitative features from chest CT images," *arXiv:2003.11988*, 2020.
[31] Farid AA, Selim GI, Khater HAA, "A novel approach of CT images feature analysis and prediction to screen for corona virus disease (COVID-19)," *International Journal of Scientific Engineering and Research*, vol. 11, no. 3, pp. 1-9, 2020.
[32] Shamsi Jokandan A, Asgharnezhad H, Shamsi Jokandan S, Khosravi A, Kebria PM, Nahavandi D, Nahavandi S, Srinivasan D, "An uncertainty-aware transfer learning-based framework for Covid-19 diagnosis," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 4, pp. 1408-1417, 2021.
[33] Polsinelli M, Cinque L, Placidi G, "A light CNN for detecting COVID-19 from CT scans of the chest," *Pattern recognition letters*, vol. 140, no. 95-100, 2020.
[34] Dan-Sebastian B, Delia-Alexandrina M, Sergiu N, Radu B, "Adversarial graph learning and deep learning techniques for improving diagnosis within CT and ultrasound images," in *Proc. of 2020 IEEE 16th international conference on intelligent computer communication and processing (ICCP)*, IEEE, pp. 449–456, 2020.
[35] Mishra AK, Das SK, Roy P, Bandyopadhyay S, "Identifying COVID19 from chest CT images: a deep convolutional neural networks based approach," *Journal of Healthcare Engineering*, 2020.
[36] Singh M, Bansal S, "Transfer learning based ensemble support vector machine model for automated COVID-19 detection using lung computerized tomography scan data," *Medical & biological engineering & computing*, vol. 59, no. 4, pp. 825-839, 2021.
[37] Ahuja S, Panigrahi BK, Dey N, Rajinikanth V, Gandhi TK, "Deep transfer learning - based automated detection of COVID-19 from lung CT scan slices," *Applied Intelligence*, vol. 51, pp. 571-585, 2021. Article (CrossRef Link)
[38] Mikołajczyk, Agnieszka & Grochowski, Michał, "Data augmentation for improving deep learning in image classification problem," in *Proc. of 2018 International Interdisciplinary PhD Workshop (IIPhDW)*, pp. 117-122, 2018. Article (CrossRef Link)
[39] L.I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*, John Wiley & Sons, 2004.
[40] R. Polikar, "Ensemble based systems in decision making," *Circuits Syst. Mag., IEEE*, vol. 6, pp. 21–45.
[41] Cubuk, Ekin & Zoph, Barret & Mane, Dandelion & Vasudevan, Vijay & Le, Quoc, "Auto Augment: Learning Augmentation Strategies from Data," pp. 113-123, 2019. Article (CrossRef Link)
[42] Ruiyong Zheng, Yongguo Zheng, Changlei Dong-Ye, "Improved 3D U-Net for COVID-19 Chest CT Image Segmentation, Scientific Programming," *Scientific Programming*, vol. 2021, 2021. Article (CrossRef Link)
[43] Roja Ramani, Duvvada & Ranjani, S, "U-Net Based Segmentation and Multiple Feature Extraction of Dermascopic Images for Efficient Diagnosis of Melanoma," *Computer Aided Intervention and Diagnostics in Clinical and Medical Images*, pp. 81-101, 2019. Article (CrossRef Link)

[44] Ronneberger, Olaf & Fischer, Philipp & Brox, Thomas, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *Proc. of Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015L*, pp. 234-241, 2015. Article (CrossRef Link)
[45] Mirjalili S, "Dragonfly algorithm: a new meta-heuristic optimization technique for solving single objective, discrete, and multi-objective problems," *Neural Computing and Application*, vol. 27, no. 4, pp. 1053–1073, 2016. Article (CrossRef Link)
[46] Reynolds CW, "Flocks, herds, and schools: a distributed behavioral model, in seminal graphics," pp. 273–282, 1998.
[47] Hammouri, Abdelaziz & Mafarja, Majdi & Al-Betar, Mohammed & Awadallah, Mohammed & Doush, Iyad, "An improved Dragonfly Algorithm for feature selection," *Knowledge-Based Systems*, vol. 203, 2020. Article (CrossRef Link)
[48] Tsai, Jung-Kai, and Chih-Hsing Hung, "Improving Ada Boost Classifier to Predict Enterprise Performance after COVID-19," *Mathematics*, vol. 9, no. 18, p. 2215, 2021. Article (CrossRef Link)
[49] Elmuogy, Samir, Hikal, Noha A., and Hassan, Esraa, "An Efficient Technique for CT scan Images Classification of COVID-19," vol. 1, pp. 5225 – 5238, 2021.
[50] Lee, Sin-ae & Cho, Hyun & Cho, Hyun-chong, "A Novel Approach for Increased Convolutional Neural Network Performance in Gastric-Cancer Classification Using Endoscopic Images," *IEEE Access*, vol. 9, pp. 51847-51854, 2021. Article (CrossRef Link)

Dr. P. Ramya is working as an Assistant Professor in the Department of Information Technology in Christian College of Engineering and Technology, Tamil Nadu, India since 2012. She has received her B.E and M.E Degree in Computer Science and Engineering from Anna University, Chennai. She has completed her doctorate degree in Information and Communication Engineering at Anna University, Chennai. Her research interests include Big Data, Clod Computing, Machine Learning and Cyber Security.

Dr. S. Venkatesh Babu is working as an Associate Professor in the Department of Computer Science and Engineering in Christian College of Engineering and Technology, Tamil Nadu, India, since 2006. He has received his B.E Information Technology and M.E Computer Science and Engineering Degree from Anna University, Chennai. He has completed his doctorate degree in Information and Communication Engineering at Anna University, Chennai. His research interests include Machine Learning, Cyber Security, and Information Processing.