An integrated feature framework for automated segmentation of COVID-19 infection from lung CT images

Deepika Selvaraj1 | Arunachalam Venkatesan12 | Vijayalakshmi G. V. Mahesh2 | Alex Noel Joseph Raj3

1Department of Micro and Nano Electronics, School of Electronics Engineering, Vellore Institute of Technology, Vellore, India
2Department of Electronics and Communication, BMS Institute of Technology and Management, Bangalore, India
3Key Laboratory of Digital Signal and Image Processing of Guangdong Province, Department of Electronic Engineering, College of Engineering, Shantou University, Shantou, China

Abstract
The novel coronavirus disease (SARS-CoV-2 or COVID-19) is spreading across the world and is affecting public health and the world economy. Artificial Intelligence (AI) can play a key role in enhancing COVID-19 detection. However, lung infection by COVID-19 is not quantifiable due to a lack of studies and the difficulty involved in the collection of large datasets. Segmentation is a preferred technique to quantify and contour the COVID-19 region on the lungs using computed tomography (CT) scan images. To address the dataset problem, we propose a deep neural network (DNN) model trained on a limited dataset where features are selected using a region-specific approach. Specifically, we apply the Zernike moment (ZM) and gray level co-occurrence matrix (GLCM) to extract the unique shape and texture features. The feature vectors computed from these techniques enable segmentation that illustrates the severity of the COVID-19 infection. The proposed algorithm was compared with other existing state-of-the-art deep neural networks using the Radiopedia and COVID-19 CT Segmentation datasets presented specificity, sensitivity, sensitivity, mean absolute error (MAE), enhance-alignment measure (EMφ), and structure measure (Sm) of 0.942, 0.701, 0.082, 0.867, and 0.783, respectively. The metrics demonstrate the performance of the model in quantifying the COVID-19 infection with limited datasets.

KEYWORDS
artificial intelligence, computed tomography image, deep neural network, feature extraction, limited training points, segmentation, Zernike moment

1 INTRODUCTION
The novel coronavirus disease-2019 (COVID-19) is a lung infection caused by Severe Acute Respiratory Syndrome (SARS) and is transmitted from person to person easily. The first human COVID-19 case has been reported in Wuhan city, China in December 2019.1 This disease was traced during the sewage water study from Milan and Turin in Italy before China identified the first case and the related information was shared in a recent article.2 The World Health Organization (WHO) has reported 10.59 million COVID cases across 213 countries and territories, as of June 2020. A large percentage of the population at present is affected by this disease. The recovery and mortality rates are 54.22% and 4.91% approximately. All over the world, researchers from biomedical departments are involved in an effort to find the effective vaccine for the COVID-19.1 However, early detection is very
important for choosing the right type of treatment to control the disease and avoid the spread. AI has been crucial in helping and serving the automated diagnostics and public health surveillance to analyse the severity of the COVID-19 as well as in protecting the people from the virus spread.3

In the human body, the lungs are the major organs of the respiratory system. The disease in the lung will affect the entire air circulation system and may even result in death. The other major diseases that affect the lungs are lung cancer, tuberculosis, pneumonia.4,5 Also, some microorganisms such as fungus, bacteria, and virus may cause illness which requires proper medical treatment to cure the infection/disease. Likewise, the infection caused due to the COVID-19 virus requires the same. The most commonly used COVID-19 tests are RT-PCR and antigen testing. Reverse transcription polymerase chain reaction (RT-PCR) is the standard COVID-19 test used to detect the viral nucleic acid that provides the genetic information of the virus. The test requires nasopharyngeal and throat swabs collected from the infected person. The duration of the test is too long and the result of RT-PCR is affected by sampling errors and low viral load.6 The antigen test is designed to detect the spike protein that is responsible for facilitating the entry of virus into the human cell. Though the test is fast, its sensitivity is poor.

An alternate detection method for COVID-19 is based on medical imaging techniques that identify the infected location and infection rate. Medical imaging records the information in the lung region using chest radiography (X-rays) and Computed-Tomography (CT) scan.7-10 Compared to the X-rays, a CT scan is mostly preferred for detecting the radiographic features of COVID-19 from the three-dimensional view of the lung’s region.11 Further, the CT scans can be used to give two-dimension views: axial view, coronal view, and sagittal view for the proper COVID infection detection. CT scan imaging of lungs gives good sensitivity to examine the COVID-19 infected region even before the medical symptoms occur.12

The spread of the infection due to COVID-19 in the lungs are identified through Ground glass opacification (GGO) and pulmonary consolidation phases. GGO is identified in the early stage of the infection whereas pulmonary consolidation indicates the final stage of the disease. These phases are observed from the CT scan images of the COVID cases.13-15 The qualitative rate of the disease indicated in CT scans provides important information to guide against COVID-19.

Rajinikanth et al.16 suggested that infection visibility analysis using CT scans gives better capability and reliability in the detection of the disease using the two-dimensional (coronal) view. The dataset used is obtained from the Radiopedia database.17 The experimental analysis and the simulation of the model were performed in a MATLAB environment.

Accurate segmentation of radiographic features is important to quantify the disease in CT scan images. Segmentation of medical images needs marking manually by experts. As the number of COVID cases is increasing rapidly, it is difficult to label the infected region manually. Thus, automatic segmentation is needed from the CT scans to detect the infected region.18 From the above consideration, our contributions are as following:

1. We propose a Deep Neural Network (DNN) model to detect and segment the COVID-19 from the axial view CT scan image from Radiopedia and COVID-19 CT Segmentation database. Our model provides better performance with limited number of training points.
2. Using the region-specific approach, instead of taking all the training points or images,19 limited training points are chosen from the infected region as well as the background region of the CT scan image. Before selecting the training points, for each image, the shape and texture descriptors are extracted using Zernike moments and GLCM, respectively.
3. The performance of the proposed method is evaluated using the standard metrics for the different test images and evaluated metrics are compared with the existing models.19-24

The methodology needs enhancement techniques for CT scans of the COVID-19 dataset, feature extraction and appropriate classifier model to detect and segment the infected region. The rest of the paper is arranged as follows. Section 2 discusses the related works on the COVID-19 using AI techniques. Section 3 gives a detailed explanation and workflow of methodology and implementation. Sections 4 and 5 outline the experimental design, results and discussion. Section 6 provides the conclusion.

## 2 RELATED WORKS ON THE COVID-19 USING AI

A comprehensive list of work for COVID-19 image-based AI techniques are found in Maga et al.18 and Fu et al.25 Developing a deep network gives more benefits for automatic and fast segmentation of the medical images.26 Camouflaged object detection (COD) is introduced to identify the embedded object with their surroundings.27 COD is beneficial in medical image applications such as lung infection segmentation. Ali et al.28 proposed the integration approach consisting of unsupervised machine
learning (self-organizing map), dimensionality reduction (principal component analysis) and computational classification (Adam Deep Learning) to present a better classification performance. Particularly, the U-Net model works well in many of the segmentation tasks for medical images. There are a few applications that have adopted U-Net for liver, heart, and multi-organ segmentation. Chen et al. has proposed a deep learning model for multiple regions auto segmentation for COVID-19 and used the aggregated residual transformations to extract the features from the CT image. Contrastive learning is proposed to train the encoder module, which provides the expressive feature information from the publicly available CT image COVID-19 dataset. The Deep Learning model requires a lesser number of image samples for training the dataset to provide an automatic classification of COVID-19 infected images. Also, the authors concluded that the contrast learning method achieves better performance than ResNet-50. Narin et al. have analysed the ROC result of three Convolutional Neural Network (CNN) models such as ResNet-50, InceptionV3, and Inception-ResNetV2 using a chest X-ray image. Shi et al. have discussed pre-processing the COVID CT images using location-specific feature extraction and infection Size Aware Random Forest method (iSARF) to distinguish the infection level and then classification using random forest. The iSARF performed well by providing Sensitivity, Specificity and Accuracy of 0.907, 0.833 and 0.879, respectively. Yujin et al. proposed the patch-based CNN using a novel probabilistic Grad-CAM saliency map with a limited number of training points from the chest X-rays. Wang et al. proposed an opensource CNN model called COVIDNet, which is trained and tested by the COVIDx dataset. It is useful to identify the COVID-19 infection from chest X-rays and has achieved a sensitivity of 80%.

Lung Infection Segmentation Deep Network (Inf-Net) used to detect the COVID-19 infection in the lungs, from the CT image slices where a parallel partial decoder is adapted to extract the high-level features. Later implicit reverse attention and explicit edge attention are used to enhance the features. Charmaine et al. have presented the deep learning model using the location-attention oriented approach to calculate the probability of COVID-19 infection region. Chest CT dataset is used for training the deep learning system.

A novel semi-supervised shallow learning model including Parallel Quantum-Inspired Self-supervised Network (PQIS-Net) with Fully-Connected (FC) layers for automatic segmentation of COVID-19 CT image is proposed in Konar et al. The patch-based classification was applied to the segmented images for the diagnosis of COVID-19 using the two publicly available datasets. The efficiency (F1-score and AUC) of the PQIS-Net was compared with pre-trained convolutional based models.

The truncated VGG-19 model was proposed to analyse the COVID-19 CT scans. The VGG-16 model was used to extract features from the CT images using fine-tuning. The feature selection was achieved through Principal Component Analysis (PCA) and the classification was done using four different classifier methods such as deep convolutional neural network (CNN), Bagging Ensemble with support vector machine (SVM), Extreme Learning Machine (ELM) and Online sequential ELM on 208 test images. From the different classifiers, Bagging Ensemble with SVM have achieved the following: an accuracy of 95.7%, AUC0.958 and F1-score 95.3%.

The three-phase COVID-19 CT image detection model is introduced in Ahuja et al. The modules are as follows: (a) Augmentation using stationary wavelets; (b) COVID-19 detection using pre-trained CNN model such as ResNet18, ResNet50, ResNet101, and SqueezeNet; (c) Abnormality localization in CT scan images. The experimental analysis showed that the pre-trained ResNet18 transfer learning model has given a better classification accuracy of 99.82% for training and 99.4% for testing.

A pipeline model is presented in Dey et al. with several sub-modules to classify the segmented region from the COVID-19 images. First, the COVID-19 CT images are segmented using the using Social-Group-Optimization and Kapur's Entropy thresholding, followed by K-means clustering and morphology-based segmentation. Next, a classification module is introduced to classify the segmented region. Here PCA based fusion technique is used to fuse the features and then fused features are trained with different classifiers such as Random Forest, k-Nearest Neighbours (KNN), Support Vector Machine with Radial Basis Function, and Decision Tree. Experimental results indicated an accuracy of 91% and 87% with Morphology-based segmentation and kNN classifier. A summary of the various techniques available for COVID-19 screening is presented in Table 1.

Wang et al. have proposed the weakly-supervised deep learning (UNet and DeCoVNet) using 3-Dimensional CT volumes to identify COVID-19. The literature indicates the successful evaluation of COVID-19 infection using deep learning and traditional machine learning methods by various researchers. But the study also clarifies that some of the regions are not evaluated in the classifier layer due to lack of training data samples and inefficient feature representation. The detailed design flow of the proposed framework is explained in the next section.
| Reference | Technique | Dataset | Findings |
|-----------|-----------|---------|----------|
| 33        | Aggregated residual transformations to learn a robust and expressive feature representation. | 110 COVID-19 CT image of size $512 \times 512$ are collected from 60 patients. | The model gives 0.95 precision and 0.89 accuracy. 12.7% of accuracy and 14.5% of precision are improvement with U-Net model. |
| 34        | Contrastive learning method. | MedSeg dataset: 110 CT COVID-19 images. COVID-19 CT dataset: 349 CT COVID-19 images 397 Non-COVID images. | ResNet-50 performance gives accuracy of 0.868, recall of 0.872, AUC of 0.931. |
| 35        | ResNet50, InceptionV3 and InceptionResNetV2. | Chest X-ray 100 images: 50 non-infected image and 50 infected images. All images are $224 \times 224$ pixel size. | ResNet50 model provides good classification performance with 98% accuracy than other two proposed models (97% accuracy for InceptionV3 and 87% accuracy for Inception-ResNetV2). |
| 36        | An infection Size Aware Random Forest method (iSARF). | COVID-19 images are taken from 1658 patients. | iSARF yielded sensitivity of 0.907, specificity of 0.833, and accuracy of 0.879. |
| 37        | A patch-based convolutional neural network approach. | COVID-19 dataset: 180 infected image of size $224 \times 224$. | Global approach gives 70.7% of accuracy, 60.6% of precision, 60.1% of recall, 59% of F1-score. Local approach gives 88.9% of accuracy, 83.4% of precision, 85.9% of recall, 84.4% of F1-score. |
| 19        | COVID-19 Lung Infection Segmentation Deep Network (Inf-Net) with a parallel partial decoder method is to aggregate the high-level features. | Dataset consists of 100 axial CT images from different COVID-19 patients. 45 CT images randomly selected as training, 5 CT images for validation, and the remaining 50 images for testing. | Inf-Net achieves 0.692 of sensitivity, 0.943 of specificity, 0.725 of sensitivity, 0.960 of specificity. |
| 40        | Semi-supervised shallow neural network model using Parallel self-supervised neural network model (PQIS-Net). | Variable size 2482 lung CT scans: 1252 COVID-19 infected CT image and 1230 Non-infected COVID image. 20 labelled COVID-19 CT image of size $512 \times 512$ which includes infection region masks, lung masks (left and right) and lung-infection pair masks. | Accuracy, precision, f1 score, and AUC are 0.931, 0.890, 0.826 and 0.982, respectively. |
| 41        | Truncated VGG-19 with fine-tuning method. Classification is done with four different classifier method such as deep convolutional neural network (CNN), Bagging Ensemble with support vector machine (SVM), Extreme Learning Machine (ELM) and Online sequential ELM. | CT Scan Dataset: 344 COVID-19 images, 358 Non-COVID-19 images. | Best performing classifier Bagging Ensemble with SVM achieves 95.7% accuracy, 95.8% precision, 0.958 AUC and 95.3% F1-score. |
| 42        | Transfer Learning method with argumentation: ResNet18, ResNet 50, ResNet101, SqueezeNet. | COVID-19:349 CT scan images and Non-COVID-19 CT: 397 CT scan images. | ResNet18 achieves the best performance model. Training accuracy is 99.82%, validation accuracy is 97.32% and testing accuracy is 99.4%. |
TABLE 1 (Continued)

| Reference | Technique                                                                 | Dataset                                                                 | Findings                                                                 |
|-----------|---------------------------------------------------------------------------|------------------------------------------------------------------------|-------------------------------------------------------------------------|
| 43        | Social-group-optimization with Kapur's entropy thresholding and then followed by k-means clustering and morphology-based segmentation. | COVID-19 dataset consists of 400 numbers of greyscale lung CT image (200 Non-infected image and 200 infected images). | Morphology-based segmentation and KNN gives more than 91% and 87% accuracy. |

FIGURE 1 Block diagram for the flow of binary classification for computed tomography image

3 | METHODOLOGY

The Datasets used in the proposed framework consists of 80 images of size 512 x 512 pixels (3 channels). The segmentation process of each CT image involves (a) pre-processing, (b) extracting the robust and sensitive features, (c) training the region-specific region, and (d) segmenting the region of interest (infected region) from the background. The block diagram of the proposed methodology is as shown in Figure 1.
3.1 Pre-processing and feature extraction methods

The pre-processing is based on the standard enhancement approach45 called adjust image intensity values (imadjust). This approach describes the expressive information from the COVID-19 CT image by adjusting the contrast of the GGO and pulmonary consolidation region. The infected pixels are brightened visually as shown in Figure 2. Also, the Contrast Improvement Index (CII)46 metrics are analysed to highlight the importance of pre-processing on the COVID dataset. The image quality (pixel intensity) is improved by 30% to 40% when compared to the original CT image as shown in Table 2. CII is computed based on Equations (1) and (2).

\[
CII = \frac{C_{\text{enhanced}}}{C_{\text{original}}} \quad (1)
\]

\[
C = \frac{I_f - I_b}{I_f + I_b} \quad (2)
\]

where \(C_{\text{enhanced}}\) and \(C_{\text{original}}\) are the contrast of the enhanced and original images respectively, \(C\) is the average value of the contrast in the enhanced or original image. \(I_f\) and \(I_b\) are the mean value for foreground and background contrast of the image. A higher CII value indicates improvement in image quality.

The enhanced images present significant details for extracting the features. The extraction of features is a description of each pixel of the CT image that contains descriptive information in the form of a stack of the vectors (feature vector). The feature vector helps to find out the output class (infected COVID-19 region and background region). The feature vector considered for the work is the integration of shape and texture features. The texture features are derived from the gray level co-occurrence matrix and shape descriptor features are extracted from the Zernike moment.

\[
Z_{nm} = \frac{m+1}{\pi} \sum_{x} \sum_{y} I(x,y) P_{n,m}(x,y), x^2 + y^2 \leq 1 \quad (3)
\]

where, \(P_{n,m}(x, y)\) is Zernike polynomial, \(I(x, y)\) is the image of size 5 \times 5 pixels, \(n\) is a non-negative integer, \(m\) is an integer such that \(0 \leq |m| \leq n\), and \(n - |m|\) is even. The function \(P_{n,m}(x, y)\) is the complex conjugate of the orthogonal basis function \(P_{n,m}(x, y)\) defined as:

\[
P_{n,m}(x, y) = P_{n,m}(r, \theta) = R_{n,m}(r)e^{im\theta} \quad (4)
\]

where, \(r = \sqrt{x^2 + y^2}\) and \(\theta = \tan^{-1}\frac{y}{x}\), with \(0 \leq \theta \leq 2\pi\), \(j = \sqrt{-1}\). The radial polynomial \(R_{n,m}(r)\) is defined as:

\[
R_{n,m}(r) = \sum_{q=0}^{n-|m|} (-1)^{q} \frac{(2n+1-q)!}{q!(n-|m|-s)!(n+|m|+1-s)!} r^{n-q} \quad (5)
\]

The Zernike moment obtained from the Equation (3) is a complex number represented as shown in Equation (6):
Table 2: CII values for the pre-processed COVID-19 CT images

| Number of images | Contrast for the original CT image | Contrast for the pre-processed image | CII for the pre-processed image |
|------------------|-----------------------------------|-------------------------------------|-------------------------------|
| 1                | 0.5984                            | 0.8166                              | 1.3646                        |
| 2                | 0.4264                            | 0.7348                              | 1.7234                        |
| 3                | 0.1650                            | 0.3279                              | 1.9872                        |
| 4                | 0.4808                            | 0.6698                              | 1.3931                        |
| 5                | 0.4846                            | 0.9497                              | 1.9599                        |

Note: Default CII value for the original image is 1.000.

Equation (7) is the magnitude of the Zernike moment $|Z_{nm}|$ that denotes the ZM shape descriptor for specified $n$ and $m$. Thus, a total number of $36|Z_{nm}|$ were obtained with $n$ ranging over 1 to 10 and matching combinations of $n$, $m$ with $m$ such that $0 \leq |m| < n$, and $n - |m|$ is even for every pixel. The magnitude of ZM calculated for every pixel to find the ZM feature maps are as displayed in Figure 3.

From the observation of Zernike features, we find that all the 36 different $|Z_{nm}|$ provide unique information about the infected and background region. To analyse the Zernike robust feature (ZM$_b$) and Zernike sensitive feature (ZM$_s$), each point is taken from both the regions with 36 ZM for different $n$, $m$ and are plotted as indicated in Figure 4 and the separation of the ZM$_b$ and ZM$_s$ is indicated in Equations (8) and (9).

$$Z_{nm} = R_{ZM} + I_{ZM}$$  \hspace{1cm} (6)

$$|Z_{nm}| = \sqrt{R_{ZM}^2 + I_{ZM}^2}$$  \hspace{1cm} (7)

$$ZM_b = \{ |Z_{0,0}|, |Z_{2,0}|, |Z_{4,0}|, |Z_{6,0}|, |Z_{8,0}|, |Z_{10,0}|, |Z_{10,8}| \}$$  \hspace{1cm} (8)

$$ZM_s = \{ |Z_{1,1}|, |Z_{2,12}| \text{ to } |Z_{5,5}|, |Z_{6,6}| \text{ to } |Z_{7,7}|, |Z_{8,8}|, |Z_{9,1}| \text{ to } |Z_{8,9}|, |Z_{10,10}| \}$$  \hspace{1cm} (9)

From the different combinations of ZM feature extraction image, we plot the intensity range for the set of pixels for the 1st image of the dataset. The graphical plot presented in Figure 5 represents number of pixels in x-axis and ZM features in y-axis. The above observation indicates that the Zernike features from background information are dense at the bottom region and the infected region information mostly occurs above a specific threshold.

To make the binary classification more accurate, we add an extra texture feature computed using the GLCM.

The texture feature is more important in medical imaging (CT image) for extracting the GGO and pulmonary consolidation features. GLCM consists of the relationship of the different angles between image pixels. If the gray level co-occurrence matrix obtained from an image be denoted as $q = [q(r, s |d, \theta)]$. In this point, GLCM is used to estimate the features of $r^{th}$ pixel frequency with the feature of $s^{th}$ pixel frequency by the length ($d = 1$) and direction ($\theta = 0$). The computation of the texture feature follows the same procedure as that of ZM, that is, computing the texture features for every pixel by moving a window of $5 \times 5$ with a stride of 1 over the pre-processed image.

From GLCM, texture features such as entropy, variance, contrast, correlation, dissimilarity, energy and homogeneity are calculated. Among all these features, we considered the GLCM variance and contrast as shown in Equations (10) and (11). These two features are combined to form GLCM features $Gf_i$ that gives proper discrimination of infected region and background region from the CT image.

$$\text{Contrast} = \sum_{r=0}^{N-1} \sum_{s=0}^{N-1} q(r, s)(r - s)^2$$  \hspace{1cm} (10)

$$\text{Variance} = \sum_{r=0}^{N-1} \sum_{s=0}^{N-1} q(r, s)(r - \mu)^2$$  \hspace{1cm} (11)

$$Gf_i = [\text{variance}_i, \text{contrast}_i]$$  \hspace{1cm} (12)

where, $i$ refers to the number of training images, $\mu$ denotes mean and ‘N’ denotes the image size, $Gf_i$ refers the GLCM features (2 features). The intensity for the GLCM feature images (contrast and variance) is plotted as shown in Figure 6. It represents number of pixels in x-axis and GLCM features in y-axis. Here the red points denote infected region and green points denote background respectively. The observation gives clear distinction between the background region and infected region.
Now the shape features computed from the ZM and the texture features obtained from the GLCM are integrated to form a feature vector of 38 features as indicated in Equation (13).

\[ F_{si} = \{ ZM_{h,i}, ZM_{l,i}, G_{fi} \} \] (13)

The next section discusses about the formation of training dataset from \( F_{si} \) features.

### EXPERIMENTAL DESIGN

#### 4.1 COVID-19 dataset

COVID-19 dataset (axial view) is collected from References 17 and 50 to evaluate our implementation. The previous implementations have used X-rays and CT image datasets for segmentation and classification of the infected COVID regions.50,51 The COVID-19 CT dataset
used in this work contains 80 images of size $512 \times 512$ pixels (3 channel) along with the ground truth image (binary image). Ground truth images are useful for the identification of targets from the input dataset. From the dataset, 30 images are selected for training the network, the remaining 50 images are used for testing the trained network for performance evaluation.

### 4.2 Training and testing dataset

COVID-19 CT images contain additional background pixels (without infected region) other than infected pixels. Using the whole image as the training data gives testing results with less accuracy that reduce the overall performance. It should be able to classify the background region from the infected region during the testing. Hence, a balanced training dataset is required for the both infected region $F_i$ and background $B_f$ pixels along with all the features from the training images.

The works$^{19,33,35}$ have used the whole image as training data. The training result has delivered a performance greater than 90% as compared to the overall testing accuracy. However, as explained in Fan et al.$^{19}$ the overall sensitivity of the test dataset is reduced by not selecting the true positive target properly (infected pixels). Hence, we chose properly infected pixels and background pixels from the extracted images.

### 4.3 Region-specific for formation of training points

The dataset of training points is manually selected from the infected and background pixels from the 30 training images (after pre-processing and $F_{si}$) by using the region-specific approach. The set of features ($S_i$) is formed by marking the specific region (infection region ($F_i$) and background region ($B_f$)) for each feature from the training image dataset as shown in Figure 7.

We chose the $F_i$ region with 62 pixels and $B_f$ region as 456 pixels. This feature helps to classify the image pixels as infected or not from COVID-19. The algorithm for region-specific approach is explained below:

```plaintext
Input: $F_{si} = \{Z_{h,i}, Z_{l,i}, G_{fi}\}$; // i refers to number of training images
Output: $F_i$; // Infection region feature.
$B_f$; // Background region features from $F_{si}$.
$S_i$; // Input feature set for training.
Start
Initialize $S_{i,k} = []$;
for i = 1 to 30 // Number of training images
    for m = 1 to 38 // Number of features
        $F_i = F_{si}[j-517:j-456]$ // Vectorized format of Infected region points
        $B_f = F_{si}[j-517:j-62]$ // Vectorized format of Background region points
        $M_i = [F_i, B_f]$;
        $S_i = [S_i, M_i]$ // Training points features from 30 training image
    end
end
Stop
```

Region selection is helpful in getting balanced image information between background and infected pixels. The selected pixels are rearranged into the vector points for training. The total training points consist of 589 380 pixels with 519 371 background pixels and 70 009 infected pixels. In the total training points, each image contains 19 646 points of 38 features. A large number of points are taken into COVID infected region to reduce the misclassification of testing image features.

During the training process, the feature vector is labelled using the ground truth as shown with $T_i$ as the target vector represented in binary form.

$$f_{ik} = [S_i, T_i] \quad (14)$$

$$T_i = [\text{class 1}_i, \text{class 2}_i] \quad (15)$$
here $f_{i,k}$ is the formed training dataset, $S$ is the input features, $T$ is target for the corresponding input features, $i$ refers to the number of training images, $k$ refers to the number of training points ($k = 1$ to $589,380$ points from all features of $F_{si}$). From the $T$, class $1 = \text{'1'}$ and class $2 = \text{'0'}$ represents an infected pixel, class $1 = \text{'0'}$ and class $2 = \text{'1'}$ represents the background pixel.

Furthermore, to select a suitable classifier for the proposed work, the performance of different classifiers such as Support Vector Machine (SVM), Deep Neural Network (DNN), Decision Tree (DT), Logistic Regression (LR) and Gaussian Naive Bayes (GNB) are compared using the training data points. The performance of classifiers is evaluated from the confusion matrix using the standard metrics: Area Under Curve (AUC), Specificity, Sensitivity and Accuracy as presented in Table 3. From Table 3, it is observed that the DNN classifier outperforms the other classifier in terms of Accuracy, AUC and Sensitivity for

**FIGURE 5** Plots for Zernike moment features from specific region of the training image (Red points represent background and Green points represents infected pixels) [Color figure can be viewed at wileyonlinelibrary.com]
the given training points and corresponding confusion matrix for the \( f_{i,k} \) for DNN has been presented in Table 4. AUC is estimated from the ROC graph for the different classifiers that are as plotted in Figure 8.

### 4.4 Binary classification using DNN

The set of ZM and GLCM features extracted using the methodology as explained in Section 3 form the feature set. The corresponding ground truth data is considered as a target. A supervised learning method is implemented by training a Deep Neural Network (DNN). Our architecture is identical to the basic feed-forward network with multiple hidden layers. This work considers the architecture where the size of the input layer is 38 neurons (38 features), three hidden layers with 58 neurons per layer and binary classification output layer as shown in Figure 9.

The total number of learnable parameters (weights and biases) in the neural network is 9224. The first layer

---

**TABLE 3** Evaluation of training points metrics for different classifiers

| Different classifiers | AUC     | Sensitivity | Specificity | Accuracy (%) |
|-----------------------|---------|-------------|-------------|--------------|
| SVM                   | 0.8723  | 0.5549      | 0.8988      | 88.6         |
| GNB                   | 0.9107  | 0.4025      | 0.9735      | 83.3         |
| LR                    | 0.8187  | 0.5211      | 0.8950      | 88.2         |
| DT                    | 0.94    | 0.756       | 0.9593      | 93           |
| DNN                   | 0.9427  | 0.7678      | 0.9285      | 93.8         |

*Note: The significance of the bold numbers are highlighted results of the proposed algorithm.*
has 2262 (38 × 58 + 58), the second layer 3422 (58 × 58 + 58), the third layer 3422 (58 × 58 + 58) and the final layer with 118 (58 × 2 + 1). The loss function (cross-entropy loss) converges by modifying the parameters using Scaled Conjugate Gradient (SCG) training function. The three hidden layers are followed with the activation function called tan-sigmoid and the output layer involves the softmax operation that gives the probability of two outputs. The probability of the first neuron in the output layer represents the lungs infected region and the second neuron is complementary of the first probability called background region. The training required 8 minutes and 49 seconds to get the best error performance of 0.1367 at 100 epochs.

### 5 | RESULTS AND DISCUSSION

The trained DNN classifier is evaluated for the test image (50 images) from the COVID-19 dataset. After the pre-processing and $F_X$, the test vector is formed for each test image of size 38 × 256 × 256. All the experimental analysis is done in MATLAB R2020a run on Intel Corei3-2330 M CPU. The feature generation process takes around 250 minutes (about 4 hours) for the test set of 50 images and testing time for each test image is around 4 to 7 minutes.

The challenges of automatic segmentation from the CT image are:

1. CT Images are varying in terms of texture, shape, and position across different cases and consolidation is small, which may result in a false negative output from a whole CT image.
2. Due to the low contrast and blurriness within the GGO of the infection area, there is a significant

| TABLE 4 | Confusion matrix of training points for DNN classifier |
|----------|-------------------------------------------------------|
|          | $T_i$                                                  |
|          | Infected pixels | Background pixels | Accuracy  |
| Training output | Infected pixels | 46 469 (7.9%) | 12 684 (2.2%) | 93.8% |
|           | Background pixels | 23 540 (4.0%) | 506 687 (85.9%) |   |

**FIGURE 8** ROC for different classifiers [Color figure can be viewed at wileyonlinelibrary.com]

**FIGURE 9** Deep neural network (DNN) architecture

**FIGURE 10** Test dataset plot for morphological structuring element vs average accuracy [Color figure can be viewed at wileyonlinelibrary.com]
difficulty in the identification of the contours in the CT images.

3. Artificial intelligence (AI) model requires a greater number of training data to train the model and more human/specialist resources are required to collect and annotate the ground truths of the COVID-19 dataset.

As explained in Section 3, the feature vector of 38 features is given to the trained DNN model as input. The output of the classifier gives the probability score for two classes (infected and background). This probability score decides and falls on particular class 1 or class 0. The probability vector output is reshaped into the image size 256 × 256. Further, the image of size 256 × 256 is

**FIGURE 11** Segmentation results for the COVID-19 dataset. A, Column shows computed tomography axial view input image; B, Column shows ground truth image; C, Column shows the output from deep neural network model and D, Column shows overlay the GT contour on output image for visual understanding [Color figure can be viewed at wileyonlinelibrary.com]
compared with the ground truth image to detect and segment the COVID-19 regions and also evaluate the performance of the model.

To make the output binary image effective, the morphological operation is applied to the output binary image. Different morphological structuring element produces different test image results. Hence, we find the suitable structuring element with a proper size for all the test images to achieve the overall average accuracy and selecting the structuring element that gives high accuracy. The average accuracy vs morphological structuring element graph for COVID-19 CT image test data is shown in Figure 10. From the plot, it is determined that “imclose” disk structuring element of size ’2’ reaches the maximum accuracy. The test dataset results are displayed in Figure 11. In Figure 11d column, GT contours are overlaid on the output image which gives visual understanding of the non-distinguishable pixels. This arises due to the poor lightening between infected pixels and background pixels from the original image as shown in Figure 11 (3rd-row image) and also the thin occurrence of GGO and consolidation in Figure 11 (6th-row image). From the ablation study Table 5, 36 ZM shape descriptor training feature points alone do not achieve better results in Sensitivity and Mean IOU. Shape descriptor features do not capture the proper infected region (True Positive) information from the test image. so that, 2 GLCM texture features are added with 36 ZM features and trained the DNN model with 38 features. Evaluated metrics achieve better results with the combination of ZM and GLCM features (38 features) compared to the 36 features results. We have analysed each test image performance by standard metrics39 such as Accuracy (A), Sensitivity or Recall (R), Specificity, F1 score, Dice index, Precision (P), Mean IOU and score using the confusion matrix as represented in Table 6.

Additionally, three more important metrics are Enhance-alignment Measure (EMφ), Mean Absolute Error (MAE)38 and Structure measure (Sm)52 are used for performance evaluation. Enhance-alignment Measure (EMφ) is a newly proposed metric. It is useful to estimate the global and local similarities of two binary images as presented in Equation (16), where, W represents width, H represents the height of GT, F represents the final predicted output, (x, y) represents each pixel coordinate of GT and F. φ denotes the enhanced alignment matrix.

\[
EM_\phi = \frac{1}{W \times H} \sum_x \sum_y \phi(F(x,y), GT(x,y)) \tag{16}
\]

Mean Absolute Error (MAE), measures the error between the F and GT of each pixel in the image as

### Table 5 Performance evaluation for the test dataset (36 ZM features)

| Number of test images | Accuracy (A) = \(\frac{TP + TN}{TP + TN + FP + FN}\) | IOU score | Sensitivity or Recall (R) = \(\frac{TP}{TP + FN}\) | Specificity = \(\frac{TN}{TN + FP}\) |
|-----------------------|---------------------------------|-----------|---------------------------------|------------------|
| 1                     | 83.5                            | 0.393     | 0.815                            | 0.604            |
| 2                     | 85.6                            | 0.337     | 0.843                            | 0.59             |
| 3                     | 88.8                            | 0.459     | 0.936                            | 0.697            |
| 4                     | 91.5                            | 0.166     | 0.928                            | 0.547            |
| 5                     | 89.3                            | 0.333     | 0.925                            | 0.629            |
| 6                     | 82.2                            | 0.382     | 0.881                            | 0.631            |
| 7                     | 97.7                            | 0.058     | 0.895                            | 0.476            |
| 8                     | 86.3                            | 0.086     | 0.872                            | 0.479            |
| 9                     | 83                              | 0.66      | 0.922                            | 0.791            |
| 10                    | 83.3                            | 0.238     | 0.824                            | 0.531            |

### Table 6 Confusion matrix for test image 1

| GT                  | Infected pixels | Background pixels | Accuracy   |
|---------------------|-----------------|-------------------|------------|
| Testing output      | Infected pixels | 14 025 (21.4%)    | 2228 (3.4%)| 92.30%    |
|                     | Background pixels | 2818 (4.3%)     |             |           |

![Table 5](image-url)
TABLE 7  Performance evaluation for the random 10 test image from COVID-19 dataset (36 ZM and 2 GLCM features)

| Number of test images | Accuracy (A) = $\frac{TP+TN}{TP+TN+FP+FN}$ | TP = TP + FP + TN | TN = TP + FP + TN | Mean IOU = $\frac{C_0 + C_1}{2}$ | Sensitivity or Recall (R) = $\frac{TP}{TP + FN}$ | Specificity = $\frac{TN}{TN + FP}$ | F1 score = $2 \times \frac{P \times R}{P + R}$ | Precision = $\frac{TP}{TP + FP}$ | Dice index | MAE | EMo | Sm |
|-----------------------|-------------------------------------------|---------------------|---------------------|-------------------------------|-----------------------------|-------------------------|-----------------|-----------------|-------------|-----|-----|-----|
| 1                     | 0.9230                                    | 0.90                | 0.73                | 0.815                         | 0.832                       | 0.95                    | 0.847           | 0.86            | 0.8028      | 0.1003 | 0.9033 | 0.8363 |
| 2                     | 0.9237                                    | 0.907               | 0.68                | 0.793                         | 0.876                       | 0.93                    | 0.815           | 0.76            | 0.7369      | 0.0945 | 0.9277 | 0.8431 |
| 3                     | 0.9610                                    | 0.95                | 0.66                | 0.805                         | 0.723                       | 0.98                    | 0.795           | 0.88            | 0.8121      | 0.1025 | 0.8512 | 0.8017 |
| 4                     | 0.9447                                    | 0.94                | 0.54                | 0.74                          | 0.835                       | 0.95                    | 0.705           | 0.61            | 0.7923      | 0.0678 | 0.7778 | 0.8411 |
| 5                     | 0.9553                                    | 0.95                | 0.56                | 0.75                          | 0.600                       | 0.99                    | 0.720           | 0.906           | 0.6479      | 0.0863 | 0.8122 | 0.7501 |
| 6                     | 0.9579                                    | 0.95                | 0.74                | 0.845                         | 0.882                       | 0.96                    | 0.851           | 0.82            | 0.7488      | 0.0703 | 0.9366 | 0.8416 |
| 7                     | 0.9788                                    | 0.97                | 0.3                 | 0.62                          | 0.330                       | 0.98                    | 0.432           | 0.66            | 0.673       | 0.0394 | 0.9646 | 0.6475 |
| 8                     | 0.9394                                    | 0.938               | 0.31                | 0.624                         | 0.346                       | 0.98                    | 0.473           | 0.81            | 0.6366      | 0.0808 | 0.7627 | 0.6103 |
| 9                     | 0.8771                                    | 0.848               | 0.61                | 0.729                         | 0.828                       | 0.89                    | 0.759           | 0.70            | 0.6665      | 0.1624 | 0.8422 | 0.7391 |
| 10                    | 0.9248                                    | 0.910               | 0.67                | 0.79                          | 0.852                       | 0.94                    | 0.704           | 0.76            | 0.6945      | 0.1128 | 0.8962 | 0.8212 |
indicated in Equation (17). Structure measure \((S_m)\) measures the structural similarity between the GT mask and prediction map as represented in Equation (18). Where \(\alpha \in [0, 1]\), the default setting \(\alpha = 0.5\) represents the balance factor between region-aware similarity \((S_r)\) and object-aware similarity \((S_o)\). The performance of each test image is evaluated by metrics as listed in Table 7.

\[
\text{MAE} = \frac{1}{W \times H} \sum_{x} \sum_{y} |F(x,y) - \text{GT}(x,y)| \quad (17)
\]

\[
S_m = \alpha \cdot S_o + (1 - \alpha) \cdot S_r \quad (18)
\]

We have compared our method with some of the existing works that employ deep network for the segmentation of the COVID-19 infected region from the CT image dataset as presented in Table 8. In Ronneberger et al.,\textsuperscript{20} standard U-Net model is evaluated using 45 training images. The outcome is low in terms of sensitivity, specificity, MAE, Dice and EM\(_{\phi}\). The modernized version of the U-Net model is described in References \textsuperscript{21}–\textsuperscript{24} which resulted in a poor evaluation of the Dice index, sensitivity compared to our developed method. In our method, MAE is quite higher than the Semi-Inf-Net and Inf-Net model reported in Fan et al.,\textsuperscript{19} EM\(_{\phi}\), \(S_m\) and Sensitivity are moderately 3%, 2.12%, 3.3% lesser compared to Semi-Inf-Net model. However, the overall average comparison of our test image result is much improved and superior with a limited number of training points as shown in Table 9. Also, Table 9 shows a performance improvement with our method in each of the metrics compared to other existing works. The proposed model improved the sensitivity by 31.2%, 10%, 6.5%, 18%, 4.3% and 1.3% compared with U-Net model, Attention-UNet, Gated-UNet, Dense-UNet, U-Net++, and Inf-Net respectively. Likewise, the percentage improvements of other metrics in relation to other state-of-the-art is presented in Table 9.

### Table 8 Comparative performance evaluation of other existing methods

| Existing reference | Methods | Sensitivity | Specificity | Dice | MAE | EM\(_{\phi}\) | \(S_m\) |
|--------------------|---------|-------------|-------------|------|-----|----------|-------|
| Ronneberger et al\textsuperscript{20} | U-Net\textsuperscript{a} | 0.534 | 0.858 | 0.439 | 0.186 | 0.625 | 0.622 |
| Oktay et al\textsuperscript{21} | Attention-UNet\textsuperscript{a} | 0.637 | 0.744 | 0.583 | 0.112 | 0.739 | 0.744 |
| Schlemper et al\textsuperscript{22} | Gated-UNet\textsuperscript{a} | 0.658 | 0.725 | 0.623 | 0.102 | 0.814 | 0.725 |
| Li et al\textsuperscript{23} | Dense-UNet\textsuperscript{a} | 0.594 | 0.655 | 0.515 | 0.184 | 0.662 | 0.655 |
| Zhou et al\textsuperscript{24} | U-Net++\textsuperscript{a} | 0.672 | 0.722 | 0.581 | 0.120 | 0.720 | 0.722 |
| Deng-Ping Fan et al\textsuperscript{19} | Inf-Net | 0.692 | 0.781 | 0.682 | 0.082 | 0.838 | 0.781 |
| Deng-Ping Fan et al\textsuperscript{19} | Semi-Inf-Net | 0.725 | 0.800 | 0.739 | 0.064 | 0.894 | 0.800 |
| Our method | DNN (specific background and infected region)\textsuperscript{b} | 0.701 | 0.942 | 0.757 | 0.082 | 0.867 | 0.783 |

\(\textsuperscript{a}\)Models are analysed in Reference \textsuperscript{19} using 45 training images and 50 test images.\textsuperscript{10}

\(\textsuperscript{b}\)\textsuperscript{589 824} training point from 30 images and 20 test images,\textsuperscript{17} 30 test images.\textsuperscript{19}

### Table 9 Percentage improvement compared between our model with existing work

| Models          | Sensitivity | Specificity | Dice   | MAE   | EM\(_{\phi}\) | \(S_m\) |
|-----------------|-------------|-------------|--------|-------|--------------|-------|
| U-Net           | 31.2%       | 9.79%       | 72.43% | 44%   | 38.7%        | 25.8% |
| Attention-UNet  | 10.04%      | 26.61%      | 29%    | 73%   | 17.3%        | 5.2%  |
| Gated-UNet      | 6.53%       | 29.93%      | 21%    | 80%   | 6.5%         | 8%    |
| Dense-UNet      | 18.01%      | 43.8%       | 46%    | 44.4% | 30.9%        | 19%   |
| U-Net++         | 4.31%       | 30.47%      | 30%    | 68%   | 20.4%        | 6%    |
| Inf-Net         | 1.30%       | 20.6%       | 10.9%  | —     | 3.4%         | 0.25% |
| Semi-Inf-Net    | —           | 17.75%      | 2.4%   | —     | —            | —     |
| Avg. improvement with our method | 10.19% | 25.5% | 30.2% | 44.2% | 16.7% | 9.17% |

Note: —, Unavailability of data.
6  |  CONCLUSION

In this paper, we propose a DNN model for COVID-19 detection from the lung’s CT axial view image. The proposed method has adopted the ZM for shape features, GLCM for texture feature and specific region for selecting the training points, precisely to extract the unique information from the CT image. Our design gives an effective detection of the COVID-19 infection in the lungs and an appropriate tool for radiologists to define the infection stage/percentage. Our model provided better results with limited number of training points. The average test dataset performance reached 70%, 94%, 86% and 78% in terms of Sensitivity, Specificity, EM$_φ$ and S$_m$, respectively. With the proposed method, the average performance is improved in terms of Sensitivity, Specificity, Dice and EM$_φ$ by 10.9%, 26%, 24.7%, and 16.5%, respectively in comparison to other popular deep networks such as U-Net, Gated-U-Net, Dense-U-Net, U-Net++, Inf-Net, Semi-U-Net. Moreover, these popular deep networks need a larger number of images for training dataset to maintain the model performance. From the results and higher evaluation metrics, it is evident that the proposed model performed significantly well with much smaller number of training points in the training dataset. The limitation of our model is the difficulty in detecting the GGO from the poor contrast CT images which require additional features and enhancement methods to provide more detailed information. In future, the inner structure information can be extracted using unique texture features and added to the training dataset to improve the performance of the DNN model and also to overcome the limitations.

ACKNOWLEDGEMENTS

This research was financially supported by The Research Start-Up Fund Subsidized Project of Shantou University, China, Grant No: NTF17016. The authors would like to thank Vellore Institute of Technology, Vellore for providing lab facilities.

DATA AVAILABILITY STATEMENT

COVID-19 CT image dataset that support the findings of this study are openly available in https://radiopaedia.org/articles/COVID-19-3?lang=us. Codes are deposited into the public github database https://github.com/deepikas517/Auto-segment-COVID-19-CT-images.git. Remaining materials and supporting data of this article are included within the article.

ORCID

Deepika Selvaraj @ https://orcid.org/0000-0003-0348-6627
Arunachalam Venkatesan @ https://orcid.org/0000-0002-7797-4749
Alex Noel Joseph Raj @ https://orcid.org/0000-0003-1505-3159

REFERENCES

1. Shi F, Wang J, Shi J, et al. Review of Artificial intelligence techniques in imaging data acquisition, segmentation and diagnosis for COVID-19. IEEE Reviews in Biomedical Engineering. 2020;1-1. https://doi.org/10.1109/rbme.2020.2987975.
2. La Rosa G, Iaconelli M, Mancini P, Bonanno Ferraro G, Veneri C, Bonadonna L, Lucentini L, Suffredini E. First detection of SARS-CoV-2 in untreated wastewaters in Italy. Science of The Total Environment. 2020;736:139652. https://doi.org/10.1016/j.scitotenv.2020.139652.
3. Fong SJ, Dey N, Chaki J. Artificial Intelligence for Coronavirus Outbreak. Singapore: Springer; 2020. https://doi.org/10.1007/978-981-15-5936-5
4. Syrjala H, Broas M, Ohtonen P, Jarti T, Pääkkö E. Chest magnetic resonance imaging for pneumonia diagnosis in outpatients with lower respiratory tract infection. European Respiratory Journal. 2017;49(1):1601303. https://doi.org/10.1183/13993003.01303-2016.
5. Zech JR, Badgeley MA, Liu M, Costa AB, Titano JJ, Oermann EK. Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study. PLOS Medicine. 2018;15(11):e1002683. https://doi.org/10.1371/journal.pmed.1002683.
6. Xie X, Zhong Z, Zhao W, Zheng C, Wang F, Liu J. Chest CT for Typical Coronavirus Disease 2019 (COVID-19) Pneumonia: Relationship to Negative RT-PCR Testing. Radiology. 2020;296(2):E41-E45. https://doi.org/10.1148/radiol.2020200343.
7. Lee JH, Kim DJ, Cho MK. Computed tomography apparatus and method of controlling X-ray by using the same, ed: Google Patents, 2017. https://patents.google.com/patent/US9655584B2/en
8. Forthmann P, Pfeiferer G. Augmented display device for use in a medical imaging laboratory. Google Patents, 2019.
9. Jensen VT. Method and system of acquiring images with a medical imaging device, ed: Google Patents, 2009. https://patents.google.com/patent/US20060293582
10. Scheib S. Dosimetric end-to-end verification devices, systems, and methods. U.S. Patent 9,643,029, issued May 9, 2017. https://patents.google.com/patent/US20150085993
11. Li Y, Xia L. Coronavirus Disease 2019 (COVID-19): Role of Chest CT in Diagnosis and Management. American Journal of Roentgenology. 2020;214(6):1280-1286. https://doi.org/10.2214/ajr.20.22954.
12. Salehi S, Abedi A, Balakrishnan S, Gholamrezanezhad A. Coronavirus Disease 2019 (COVID-19): A Systematic Review of Imaging Findings in 919 Patients. American Journal of Roentgenology. 2020;215 (1):87-93. https://doi.org/10.2214/ajr.20.23034.
13. Huang C, Wang Y, Li X, et al. Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. The Lancet. 2020;395(10223):497-506. https://doi.org/10.1016/s0140-6736(20)30183-5.
14. Wang L, Wang Y, Ye D, Liu Q. Review of the 2019 novel coronavirus (SARS-CoV-2) based on current evidence. International Journal of Antimicrobial Agents. 2020;55(6):105948. https://doi.org/10.1016/j.ijantimicag.2020.105948.
15. Li K, Fang Y, Li W, Pan C, Qin P, Zhong Y, Liu X, Huang M, Liao Y, Li S. CT image visual quantitative evaluation and clinical classification of coronavirus disease (COVID-19). European Radiology. 2020;30(8):4407-4416. https://doi.org/10.1007/s00330-020-06817-6.

16. Rajnikanth V, Dey N, Raj ANJ, Hassanien AE, Santosh KC, Raja N. Harmony-search and otus based system for coronavirus disease (COVID-19) detection using lung CT scan images. arXiv preprint arXiv:2004.03431;2020.

17. https://radiopaedia.org/articles/COVID-19-3?lang=us. Last Accessed date April 5, 2020.

18. Gaál G, Maga B, Lukács A. Attention u-net based adversarial architectures for chest x-ray lung segmentation. arXiv preprint arXiv:2003.10304; 2020. https://arxiv.org/pdf/2003.10304.pdf

19. Fan DP, Zhou T, Ji GP, et al. Inf-net: automatic COVID-19 lung infection segmentation from CT images. IEEE Trans Med Imaging. 2020;39:2626-2637. https://doi.org/10.1101/2020.04.22.20074948

20. Ronneberger O, Fischer P, Brox T. U-net: convolutional networks for biomedical image segmentation. International Conference on Medical Image Computing and Computer-Assisted Intervention. Cham: Springer; 2015, October:234-241. https://doi.org/10.1007/978-3-319-24574-4_28

21. Oktay O, Schlemper J, Folgoc LL, Lee M, Heinrich M, Kainz B, Grochowski M, Glocker B, Rueckert D. Attention gated networks: Learning to look for abnormalities. arXiv preprint arXiv:1804.03999; 2018. https://arxiv.org/pdf/1804.03999.pdf

22. Schlemper J, Oktay T, Schaap M, Heinrich M, Kainz B, Glocker B, Rueckert D. Attention gated networks: Learning to leverage salient regions in medical images. Medical Image Analysis. 2019;53:197-207. https://doi.org/10.1016/j.media.2019.01.012.

23. Li X, Chen H, Qi X, Dou Q, Fu C-W, Heng P-A. H-DenseUNet: Hybrid Densely Connected UNet for Liver and Tumor Segmentation From CT Volumes. IEEE Transactions on Medical Imaging. 2018;37(12):2663-2674. https://doi.org/10.1109/tmi.2018.2845918.

24. Zhou Z, Siddiquee MMR, Tajbakhsh N, Liang J. U-net++: A nested u-net architecture for medical image segmentation. Deep Learning in Medical Image Analysis and Multimodal Learning. Cham: Springer; 2018:3-11. https://doi.org/10.1007/978-3-030-49482-7_1.

25. Fu H, Fan DP, Chen G, Zhou T. COVID-19 imaging-based AI research collection. 2020. https://github.com/HzFu/COVID19_imaging_AI_paper_list.

26. Angulakshmi M, Lakshmi Priya GG. Automated brain tumour segmentation techniques- A review. International Journal of Imaging Systems and Technology. 2017;27(1):66-77. https://doi.org/10.1002/ima.22211.

27. Fan DP, Ji GP, Sun G, Cheng MM, Shen J, Shao L. Camouflaged object detection. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, WA, USA, USA: IEEE; 2020:2777-2787. https://doi.org/10.1109/CVPR42600.2020.00285

28. Ali MNV, Sarowar MG, Rahman ML, Chaki J, Dey N, Tavares JMR. Adam Deep Learning With SOM for Human Sentiment Classification. International Journal of Ambient Computing and Intelligence. 2019;10(3):92-116. https://doi.org/10.4018/ijaci.2019070106.

29. Shan F, Gao Y, Wang J., Shi W, Shi N, Han M, Shi Y. Lung infection quantification of covid-19 in CT images with deep learning. arXiv preprint arXiv:2003.04655; 2020. https://arxiv.org/ftp/arxiv/papers/2003/2003.04655.pdf

30. Ye C, Wang W, Zhang S, Wang K. Multi-Depth Fusion Network for Whole-Heart CT Image Segmentation. IEEE Access. 2019;7:23421-23429. https://doi.org/10.1109/access.2019.2899635.

31. Liu Z, Song Y-Q, Sheng VS, Wang L, Jiang R, Zhang X, Yuan D. Liver CT sequence segmentation based with improved U-Net and graph cut. Expert Systems with Applications. 2019;126:54-63. https://doi.org/10.1016/j.eswa.2019.01.055.

32. Dong X, Lei Y, Wang T, Thomas M, Tang L, Curran WJ, Liu T, Yang X. Automatic multiorgan segmentation in thorax CT images using U-net-GAN. Medical Physics. 2019;46(5):2157-2168. https://doi.org/10.1002/mp.13458.

33. Chen X, Yao L, Zhang Y. Residual attention U-net for automated multi-class segmentation of COVID-19 chest CT images. arXiv preprint arXiv:2004.05645; 2020. https://arxiv.org/pdf/2004.05645.pdf

34. Chen X, Yao L, Zhou T, Dong J, Zhang Y. Momentum contrastive learning for few-shot covid-19 diagnosis from chest CT images. arXiv preprint arXiv:2006.13276; 2020. https://arxiv.org/pdf/2006.13276.pdf

35. Narin A, Kaya C, Pamuk Z. Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. arXiv preprint arXiv:2003.10849; 2020. https://arxiv.org/pdf/arXiv.preprint arXiv:2003.10849.pdf

36. Shi F, Xia L, Shan F, Wu D, Wei Y, Yuan H, Shen D. Large-scale screening of covid-19 from community acquired pneumonia using infection size-aware classification. arXiv preprint arXiv:2003.09860; 2020. https://arxiv.org/ftp/arxiv/papers/2003/2003.09860.pdf

37. Oh Y, Park S, Ye JC. Deep Learning COVID-19 Features on CXR Using Limited Training Data Sets. IEEE Transactions on Medical Imaging. 2020;39(8):2688-2700. https://doi.org/10.1109/tmi.2020.2993291.

38. Wang L, Wong A. COVID-Net: a tailored deep convolutional neural network Design for Detection of COVID-19 cases from chest X-ray images. arXiv preprint arXiv:2003.09871; 2020. https://arxiv.org/pdf/2003.09871.pdf

39. Butt C, Gill J, Chun D, Babu BA. RETRACTED ARTICLE: Deep learning system to screen coronavirus disease 2019 pneumonia. Applied Intelligence. 2020;1-7. https://doi.org/10.1007/s10489-020-01714-3.

40. Konar D, Panigrahi BK, Bhattacharyya S, Dey N. Auto-Diagnosis of COVID-19 Using Lung CT Images with Semi-Supervised Shallow Learning Network. Research Square. 2020. https://doi.org/10.21203/rs.3.rs-34596/v1.

41. Singh M, Bansal S, Ahuja S, Dubey RK, Panigrahi BK, & Dey N. Transfer Learning Based Ensemble Support Vector Machine Model for Automated COVID-19 Detection Using Lung Computerized Tomography Scan Data. Research Square. 2020. https://doi.org/10.21203/rs.3.rs-32493/v1.

42. Ahuja S, Panigrahi BK, Dey N, Rajnikanth V, Gandhi TK. Deep transfer learning-based automated detection of COVID-19 from lung CT scan slices. Applied Intelligence. 2020. https://doi.org/10.1007/s10489-020-01826-w.

43. Dey N, Rajnikanth V, Fong SJ, Kaiser MS, Mahmud M. Social group optimization-assisted Kapur's entropy and morphological
segmentation for automated detection of COVID-19 infection from computed tomography images. *Cognitive Computation*. 2020;12(5):1011-1023. https://doi.org/10.1007/s12559-020-09751-3.

44. Wang X, Deng X, Fu Q, Zhou, Feng J, Ma H, Liu W, Zheng C. A weakly-supervised framework for COVID-19 classification and lesion localization from chest CT. *IEEE Transactions on Medical Imaging*. 2020;39(8):2615–2625. https://doi.org/10.1109/tmi.2020.2995965.

45. Chaki J, Dey N. *A beginner’s guide to image preprocessing techniques*. Boca Raton: CRC Press; 2018. https://doi.org/10.1201/9780429441134

46. Subramani B, Veluchamy M. MRI brain image enhancement using brightness preserving adaptive fuzzy histogram equalization. *International Journal of Imaging Systems and Technology*. 2018;28(3):217-222. https://doi.org/10.1002/ima.22272.

47. Raj ANJ, Mahesh VG. Zernike-moments-based shape descriptors for pattern recognition and classification applications. *Advanced Image Processing Techniques and Applications*. Hershey: IGI Global; 2017:90-120. https://doi.org/10.4018/978-1-5225-2053-5.ch004

48. Adapa D, Joseph Raj AN, Alisetti SN, Zhuang Z, Ganesan K, Naik G. A supervised blood vessel segmentation technique for digital Fundus images using Zernike Moment based features. *PLOS ONE*. 2020;15(3):e0229831. https://doi.org/10.1371/journal.pone.0229831.

49. Barstugan M, Ozkaya U, Ozturk S. Coronavirus (covid-19) classification using ct images by machine learning methods. *arXiv preprint arXiv:2003.09424*; 2020. https://arxiv.org/ftp/arxiv/papers/2003/2003.09424.pdf

50. COVID-19 CT segmentation dataset. https://medicalsegmentation.com/covid19/. Accessed April 11, 2020.

51. Cohen JP, Morrison P, Dao L, Roth K, Duong TQ, Ghassemi M. Covid-19 image data collection: Prospective predictions are the future. *arXiv preprint arXiv:2006.11988*; 2020. https://arxiv.org/pdf/2006.11988.pdf

52. Fan DP, Cheng MM, Liu Y, Li T, Borji A. Structure-measure: a new way to evaluate foreground maps. *Proceedings of the IEEE International Conference on Computer Vision*. Venice, Italy: IEEE; 2017:4548-4557.

**How to cite this article:** Selvaraj D, Venkatesan A, Mahesh VGV, Joseph Raj AN. An integrated feature frame work for automated segmentation of COVID-19 infection from lung CT images. *Int J Imaging Syst Technol*. 2020;1–19. https://doi.org/10.1002/ima.22525