Classification of Covid-19 Diseases Through Lung CT-Scan Image Using the ResNet-50 Architecture

Lucky Indra Kesuma[*], Rudiansyah Rudiansyah

Department of Information System, Faculty of Computer Science, Universitas Sjakhyakirti
*luckyindra25@unisti.ac.id

ABSTRACT

Covid-19 is a disease of the respiratory tract caused by the Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2) virus. One way to diagnose Covid-19 can be done by examining lung abnormalities on the results of a Computed Tomography Scan (CT-Scan) of the lungs. However, the determination of the diagnostic results obtained requires high accuracy and quite a long time. For this reason, an automatic system is needed to make it easier for medical personnel to diagnose Covid-19 disease quickly and accurately. One way to do this with the help of a computer is pattern recognition. In this study, pattern recognition techniques were used which were divided into three stages, namely pre-processing, feature extraction and classification. The methods used in the pre-processing stage are grayscale and Contrast Limited Adaptive Histogram Equalization (CLAHE) to improve image quality and contrast. The extraction stage uses the Principal Component Analysis (PCA) method, because it can reduce data dimensions without eliminating important features in the data. For the classification stage, a deep learning-based method is used, namely the Convolutional Neural Network (CNN). The CNN architecture used in this study is Resnet-50. The method proposed in this research is evaluated by measuring the performance values of accuracy, recall, precision, F1-score, and Cohen Kappa. The results of the study indicate that the PCA method has worked optimally in dimension reduction, without losing important features on CT-scan images of the lungs. Besides that, the proposed method has succeeded in classifying Covid-19 very well, as seen from the accuracy, Recall, Precision, F1-Score and Cohen Kappa values above 90%.

Keywords: Classification, Covid-19, ResNet, PCA, Feature Extraction

1. INTRODUCTION

Covid-19 is a respiratory tract disease caused by the Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2). The Covid-19 disease was first reported in Wuhan, China, in December 2019 [1]. The SARS-CoV-2 virus is primarily transmitted through human contact, and the World Health Organization has proclaimed a global pandemic (WHO) [2]. Symptoms of Covid-19 can range from asymptomatic to mild and severe [3]. One way to diagnose Covid-19 disease can be done by examining lung abnormalities on the results of a Computed Tomography Scan (CT-Scan) of the lungs [4]. However, determining the diagnostic results requires high accuracy and a long time. For this reason, an automated system is needed to make it easier for medical personnel to diagnose Covid-19 disease quickly and accurately [5]. One of the automated systems with computer assistance in

detecting abnormalities in CT-Scan images of the lungs is to perform pattern recognition.

Pattern recognition techniques generally consist of several stages: preprocessing, feature extraction, and classification [6]. Preprocessing is the initial process in image detection that is carried out to improve image quality and eliminate noise and parts that are unimportant in the image [7]. The feature extraction stage is a step that is carried out to remove all noise and redundancy and will only retain the most informative features. The most frequently used feature extraction method is Principal Component Analysis (PCA) [10-11]. The PCA method helps reduce the image's dimensions without losing information, simplifying data analysis and visualization while accelerating the training process [7]. Several studies using PCA techniques include Fang et al. [9] reduced X-ray Absorption Spectroscopy (XAS) images, Darvish & Ezoji [10] reduced retinal images, Adiwijaya et al. [11] reduced DNA microarray data, Montalvo et al. [12] reduced agricultural images of plants, and T. et al. [13] reduced MR images of brain tumors. The results obtained using PCA yielded better scores than those without PCA.

Classification is the process of categorizing objects into different classes based on their similarity of characteristics [14]. Several studies that have classified the Covid-19 disease include research by Sharma et al. [15] used improved Random Forest (IRF), Support Vector Machine (SVM), and Random Forest (RF) methods to produce performance evaluation values in the form of precision, recall, and F1-score. For the IRF method, the resulting values are 76.5%, 76.5%, and 76.5%, respectively. Meanwhile, the SVM method is 71.2%, 76.5%, and 71.2%, while the IRF method is 60.1%, 60.1%, and 60.1%. Another study by Akinnuwesi et al. [16] using the Decision Tree (DT) method resulted in Accuracy, Precision, Recall, and F1-score values of 65.6%, 65.6%, 72.6%, and 65.9%, respectively. It can be seen from the two studies conducted that the results of the performance evaluation obtained on average are still below 80%. This is because the methods used in both studies are still conventional. The conventional method has a weakness in that it is not yet able to distinguish an object from other objects because the conventional method only studies the feature representation superficially.

In recent years, deep learning methods suitable for image classification have developed rapidly, one of which is the Convolutional Neural Network (CNN) [17]. The CNN method used is entirely accurate in classifying various types of images. Several studies have applied CNN in the classification of biomedical images, including Zhang et al. [18] performed a hyperspectral image classification, Qin et al. [19] performed a biological image classification, Jadoon et al. [20] performed a mammogram classification, Sajjad et al. [21] performed a classification brain tumors, Ghosh [22] classified diabetic retinopathy, and M. Zhao et al. [23] classified cloud shape.

Adding layers can improve CNN's performance by adding layers [10]. However, adding a CNN layer can result in increased layer complexity and gradient vanishing [24]. To overcome this, ResNet architecture [10, 27] can be used. The advantages of ResNet are that its performance does not decrease even though there are more layers, computational calculations become faster and good training capabilities. Various studies that use ResNet in the image classification process, namely Mustafa & Meehan [25], applied it to gender and age classification by obtaining an accuracy value of 85%. In another study, Zhu et al. [26] applied it to detecting ECG signals by obtaining an F-1 Score of 85%. Sarwinda et al. [24] applied it to the classification of

colorectal cancer by obtaining 80% accuracy, 87% sensitivity, and 83% specificity. Guo & Yang [27] applied it to the classification of skin lesion diseases by obtaining an accuracy value of 82.4%. M. Guo & Du [28] applied it to classifying Thyroid Ultrasound Standard Plane (TUSP) images by obtaining an accuracy value of 83.88%. From the advantages of the PCA and ResNet methods, this study will combine the two for classifying Covid-19 disease using an automatic lung CT-Scan.

2. MATERIAL AND METHODS

2.1 DATASET DESCRIPTION

The dataset used in this study is the Large Covid-19 CT-Scan Slice dataset which can be accessed for free via the https://www.kaggle.com/maedemaftouni/large-covid19-ct-slice-dataset page. This dataset has three classes, including 7,593 Covid-19 images from 466 patients, 6,893 normal images from 604 patients, and 2,618 CAP images from 60 patients. Some examples of CT-Scan images of the lungs for each label in this dataset can be seen in Figure 1.
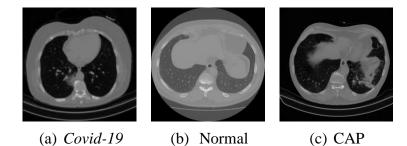

(a) *Covid-19*　　　(b) Normal　　　(c) CAP

FIGURE 1. Sample Lung CT-Scan Image for Each Label

2.2 PREPROCESSING DATA

At the preprocessing stage, the data aims to improve image quality and make classifying images easier. Some medical images often have low contrast and noise due to errors in various sources of interference, such as the imaging process and data acquisition, so the image is difficult to analyze visually [29]. Image quality improvement in this study is to increase the contrast of dark images. The preprocessing stages are carried out as follows:

2.2.1 GRAYSCALE

Before increasing the contrast of the image, the CT-Scan image of the lungs is first converted into a Grayscale image to facilitate the image processing. Grayscale is an image with a pixel intensity value based on the degree of gray, which only contains brightness information and no color information [30]. The grayscale image intensity scale is in the range $[0, 255]$.

2.2.2. CONTRAST LIMITIED ADAPTIVE HISTOGRAM EQUALIZATION (CLAHE)

CLAHE is one of the low contrast enhancement techniques in images that can effectively highlight the details of an image, especially in medical images [31]. In the CLAHE technique, the input image will be partitioned into contextual regions (tiles), then apply a Clip Limit (CL) on Histogram Equalization (HE) to those regions by limiting the increase in each pixel value. Next, redistribute the cut pixels across the histogram range [31]. CL is the maximum value of a histogram and can cause over brightness in a limited area.

2.3 PRINCIPAL COMPONENT ANALYSIS (PCA)

The dimensions of an image affect the accuracy of the classification and the use of computational time during training. The larger the dimensions of the image, it can reduce the more accuracy and computation time become long and heavy. PCA is one method to handle large data dimensions by reducing the data dimensions without losing essential information [32]. Other advantages are eliminating the correlation between features, reducing data redundancy, and reducing overfitting [33].

Mathematically, the PCA process is carried out by reducing the n-dimensional image feature space to k-dimensional ($k < n$) where the k-dimensional feature is a new orthogonal feature from the result of the original n-dimensional feature reconstruction, which is commonly called the Principal Component (PC) [34]. For example, an image is assumed to be an X matrix of size $m \times n$ with the following matrix entries [35]:

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1j} & \cdots & x_{1n} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i1} & \cdots & x_{ij} & \cdots & x_{in} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mj} & \cdots & x_{mn} \end{bmatrix} \tag{1}$$

Next, the input image is transformed into feature spaces represented in the form of a vector measuring $m \times 1$ as follows [34].

$$x_j = \begin{bmatrix} x_{1j} \\ x_{2j} \\ \vdots \\ x_{mj} \end{bmatrix} \tag{2}$$

The steps for reducing image features in detail using the PCA method are as follows:
1. Data standarization, this process aims to scale the data so that the average of each image feature space is 0 and the standard deviation is 1 [36]. This can avoid the influence of different image feature space dimensions [37]. In performing the data standardization process, first, calculate the mean ($\bar{x}_j$) and standard deviation (σ_j) for each image feature space using Equations (3) and (4) as follows [37]:

$$\bar{x}_j = \frac{1}{m}\sum_{i=1}^{m} x_{ij} \tag{3}$$

$$\sigma_j = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(x_{ij} - \bar{x}_j)^2} \tag{4}$$

Then, calculate the results of data standardization with Z-score standardization using Equation (5) below [35].

$$z_{ij} = \frac{x_{ij} - \bar{x}_j}{\sigma_j} \tag{5}$$

where $i = 1,2,\dots,m$ and $j = 1,2,\dots,n$, x_{ij} is the input matrix entry X in the ith row and j^{th} column, $\bar{x}_j$ is the average value of the image feature space j^{th}, σ_j is the standard deviation of the jth image feature space, and z_{ij} is the matrix entry resulting from the standardization of data in the ith row and j^{th} column.

2. Covariance matrix, covariance measures the strength of the correlation between two feature spaces in an image [36]. The covariance matrix is a matrix with the order $n \times n$ whose entries are the covariance values between the ith and j^{th} row image feature spaces [36]. The form of the covariance matrix can be seen through Equations (6) to (7) below [49].

$$C = \begin{bmatrix} \sigma_{11}{}^2 & Conv(Z_1,Z_2) & Conv(Z_1,Z_3) & Conv(Z_1,Z_4) \\ Conv(Z_2,Z_1) & \sigma_{22}{}^2 & Conv(Z_2,Z_3) & Conv(2,Z_4) \\ Conv(Z_3,Z_1) & Conv(Z_3,Z_2) & \sigma_{33}{}^2 & Conv(Z_3,Z_4) \\ Conv(Z_4,Z_1) & Conv(Z_4,Z_2) & Conv(Z_4,Z_3) & \sigma_{44}{}^2 \end{bmatrix} \tag{6}$$

To calculate the value of variance $(\sigma_j{}^2)$ and covariance $Cov(z_i,z_j)$ can use Equations (2.10) and (2.11) as follows.

$$\sigma_j{}^2 = \frac{1}{m}\sum_{i=1}^{m}(z_{ij})^2 \tag{7}$$

$$Cov(z_i,z_j) = \frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{n}(z_{ij})(z_{ji}) \tag{11}$$

where C is the covariance matrix, $\sigma_j{}^2$ is the variance of the jth row image feature space, and $Cov(z_i,z_j)$ is the ith row and j^{th} column image feature space covariance.

3. Eigenvalues and eigenvectors, the size of the covariance matrix's eigenvalues determines the PCA method's dimensionality reduction results [38]. In addition,

the eigenvectors generated in the image feature space must be orthogonal to each other, so there is no correlation between the image feature spaces [37]. So can use Equations (8) and (9) to determine the eigenvalues and eigenvectors [32]:

$$det(C - \lambda I) \tag{8}$$

$$Cv = \lambda v \tag{9}$$

where λ is the eigenvalue, is the eigenvector, and I is the identity matrix. The eigenvalues obtained will be sorted based on the largest to the smallest eigenvalues, $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$, while the eigenvectors are arranged in order of eigenvalues [36].

4. Principal component, the result of image dimension reduction using the PCA method is built from many PC, which can explain most of the diversity of data. The number of PC is selected based on the cumulative proportion of variance from each PC. The image feature space that contributes high will be maintained, and the low contribution image feature space will be eliminated [39]. To determine the PC matrix and the cumulative proportion of variance can use the following Equation (10) and (11) [37].

$$PC = Z^T V \tag{10}$$

$$Pk(PC_i) = \frac{\lambda_i}{\sum_{i=1}^{m} \lambda_i} \qquad ; i = 1, 2, \dots, m \tag{11}$$

where PC is the PC matrix, Z^T is the standardized data transpose matrix, V is the eigenvector matrix, $Pk(KU_i)$ is the cumulative proportion of the ith PC, and λ_i is the value i^{th} eigen.

2.4 RESNET-50 ARCHITECTURE

The CNN method outperforms conventional methods in image classification systems [40]. One of the CNN architectures used in this study is the Residual Network (ResNet) developed by He et al. [41]. ResNet utilizes a "residual connection" in its layer to handle the gradient descent problem, thereby accelerating the convergence of deep networks [42]. This architecture has lower computational complexity than other architectures, although the depth is increased [43]. The deep network on the ResNet architecture proved to be better at performing classification tasks because it can extract more representative features [44]. The number of layers used by the ResNet architecture is 50 (seen in Figure 4), with 48 convolution layers, one max pooling layer, and one global average pooling layer. Some of the other CNN layers used in the ResNet-50 architecture are as follows:

2.4.1. CONVOLUTION LAYER

The convolution layer is the main layer on CNN that performs convolution operations to study the feature representation of the input image [56]. The

convolution layer applies several kernels to generate feature maps representing various input image characteristics [45]. To calculate the results of the convolution layer, we can use Equations (12) and (13) as follows [46]:

$$C_q^p = (A_p * K_q) + b_q \tag{12}$$

$$c_{i,j} = \left(\sum_{u=0}^{n-1} \sum_{v=0}^{n-1} (a_{u+i,v+j} \times k_{i+1,j+1}) \right) + b_q \quad ; \quad i,j = i, 2, \dots, n \tag{13}$$

where i is row, j is column, n is kernel height measurement, $*$ is convolution operation, A_p is p^{th} input matrix, K_q is q^{th} kernel matrix, b_q is bias for q^{th} kernel, and C_q^p is matrix of convolution results (feature maps) of the qth kernel on the pth input.

2.4.2 BATCH NORMALIZATION

Batch normalization is an additional layer that functions to normalize input values in the next layer, reduce the risk of overfitting, and speed up the training process [57], [59]. Batch normalization transforms the distribution of input values into a standard normal distribution with a mean of 0 and a variance of 1 [60]. Batch normalization works by normalizing the input image by subtracting each image input entry by the mini-batch average, then dividing it by the mini-batch variance shown in Equations (14) to (16) below [61]:

$$\mu_j = \frac{1}{m} \sum_{i=1}^{m} x_{ij} \tag{14}$$

$$\sigma_j^2 = \sqrt{\frac{1}{m} \sum_{i=1}^{m} (x_{ij} - \mu_j)^2} \tag{15}$$

$$\hat{x}_{i,j} = \frac{x_{i,j} - \mu_j}{\sqrt{\sigma_j^2 + \varepsilon}} \tag{16}$$

where $i = 1, 2, \dots, m$, $j = 1, 2, \dots, n$, where n is the number of mini-batches (columns), m is the number of data in one mini-batch (rows), x_{ij} is the input matrix entry X in the ith row and j^{th} column, $\hat{x}_{ij}$ is the input matrix entry X which has been normalized to the ith row and j^{th} column, and ε is the smallest positive constant.

2.4.3. POOLING LAYER

The pooling layer is a CNN layer that serves to help reduce the dimensions of the map features by maintaining most of the dominant features at each step of the pooling stage [42]. Several types of pooling layers are used, including max pooling and global average pooling. Max pooling works by taking the maximum value for each submatrix generated by shifting kernel pooling and stride [47]. Global average pooling works by taking the average value on the map feature measuring $n \times n$, which will be converted into a 1×1 matrix [42].

2.4.4. ACTIVATION FUNCTION

In CNN, an activation function is applied to the output in each convolution layer to add a nonlinear factor so that it can overcome complex problems such as increasing training speed [48]. Rectified Linear Unit (ReLU) and sigmoid functions are some examples of activation functions that are often used. ReLU is an activation function with lower computational cost and better gradient convergence [45]. ReLU makes all pixel values in the image with negative values become 0, whereas they will have the same value if the value is positive. The sigmoid activation function is an activation function where the input is a real number, and the output is in the interval $(0, 1)$ [42]. Mathematically, the ReLU and sigmoid activation functions are defined in Equations (17) and (18) [46,52].

$$r(z) = max(z, 0) = \begin{cases} z & if \; z \geq 0 \\ 0 & if \; z < 0 \end{cases} \tag{17}$$

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{18}$$

for $z \in (-\infty, +\infty)$, where $r(z)$ is the ReLU activation function, $\sigma(z)$ is the sigmoid activation function with $\sigma(z) \in (0,1)$.

2.4.5. DENSE LAYER

The dense layer is a layer that functions to classify images that have passed the feature extraction process into a predetermined label. This layer works by feeding all the outputs in the previous layer to all its neurons, where each neuron will produce one output to the next layer and then be processed into an activation function [49]. The Equation (19) can be used to calculate the output of the dense layer [50].

$$Z = W^T \times X + b \tag{19}$$

where Z is the output matrix, W is the weight matrix in the dense layer neurons, X is the input from the previous layer, and b is the bias.

2.5 TRAINING AND TESTING

After going through the image dimension reduction process using PCA, the CT-Scan image of the lungs is ready to be processed to the next stage, image classification into three classes, namely Covid-19, normal, and CAP. Image

classification is a process of categorizing images into one of the labels determined using a specific model [51]. The lungs have two stages in applying the ResNet-50 architecture to the CT-Scan image classification process: training and testing. Before the training and testing stages, the PCA images are divided into 2 data sets: training data and test data.

2.5.1. TRAINING DATA

At this stage, it is carried out to build a ResNet-50 classification model by training the architecture on the training data so that patterns or features of the data can be recognized and studied. The first step at this stage is to initialize the parameters to be used, namely the number of epochs and batch size. Next, the PCA image is divided into training data and validation data. Then, initialize the weight value in the first epoch. Next, training data and validation data are extracted into the ResNet-50 architecture.

2.5.2. TESTING DATA

At this stage, the model that has been generated through the training stage is tested against new data or test data to see how successful the proposed architecture is. Finally, the best weight in the ResNet-50 model will be used to predict the class of the CT-Scan image of the lungs.

2.6 EVALUATION

In this study, the performance of the ResNet-50 model was evaluated by measuring the accuracy, sensitivity, specificity, F1-score, Intersection over Union (IoU), and cohen's kappa. Accuracy is a measure of architectural accuracy based on the ratio of the number of CT-Scan images of the lungs correctly classified by the model [52]. Sensitivity is a measure of the model's ability to correctly classify positive classes, while specificity is a measure to see the model's ability to classify negative classes correctly. F1-Score is a measure to see the match between the actual class and the predicted results by calculating the harmonic average between precision and sensitivity [42], [53], [54]. Cohen's kappa (k) is a measure to measure the agreement between the actual class and the predicted results on a nominal scale with two or more classes [55]. The higher cohen's kappa value, the better the similarity, whereas the lower the value indicates the possibility of the prediction results occurring by chance [56].

The workflow of the proposed method in this study is shown in Figure 2.
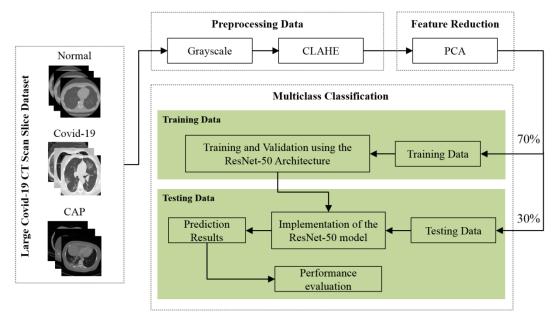
FIGURE 2. Proposed Method Workflow

3. RESULT AND DISCUSSION

The data used in the research is a CT-Scan image of the lungs measuring 150×150 pixels from the Large Covid-19 CT-Scan Slice dataset. The pre-processing stages carried out in this study can be seen in Figure 3.

In Figure 3, it can be seen that there is an input image with three different class labels, namely CAP, covid and normal. The input image is improved image quality using Grayscale and CLAHE. First, grayscale stages are carried out to simplify the image management process. After the Grayscale process, the image quality is improved using the CLAHE method. Furthermore, images that have been pre-processed need to be reduced in image dimensions to obtain important features without losing information using PCA. The graph of image dimension reduction using PCA can be seen in Figure 4.

In Figure 4, it can be seen that the greater the number of PC in the image, the value of the cumulative variance proportion is getting closer to 100%. In this study, the PC value on the CT-Scan image of the lungs is 50, so the cumulative variance value is around 95%. From the cumulative variance value, it can be said that the image dimension reduction performed on the CT-Scan image of the lungs is insignificant, thus making essential features in the image not lost. Some examples of images from the dimensional reduction stages using PCA on three labels of CT-Scan images of the lungs with a PC value of 50 can be seen in Figure 5.
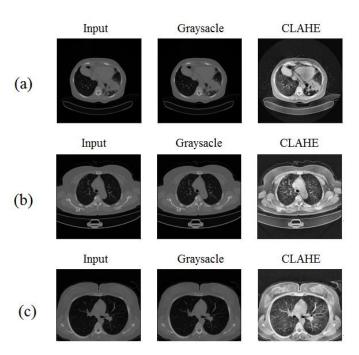
FIGURE 3. Preprocessing Results on CT-Scan Images with Labels
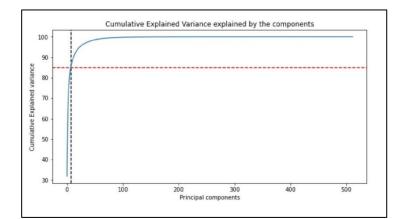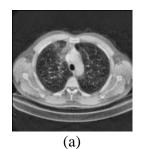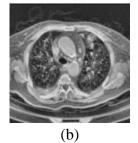(a) CAP (b) Covid (c) Normal
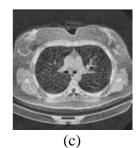

FIGURE 4. Dimensional Reduction Using PCA

FIGURE 5. PCA Results on CT-Scan Images with The Label
(a) Covid (b) CAP (c) Normal

The results of the image reduction are continued by classifying CT-Scan images using the ResNet-50 architecture into three different labels. The ResNet-50 architecture used can be seen in Figure 6.
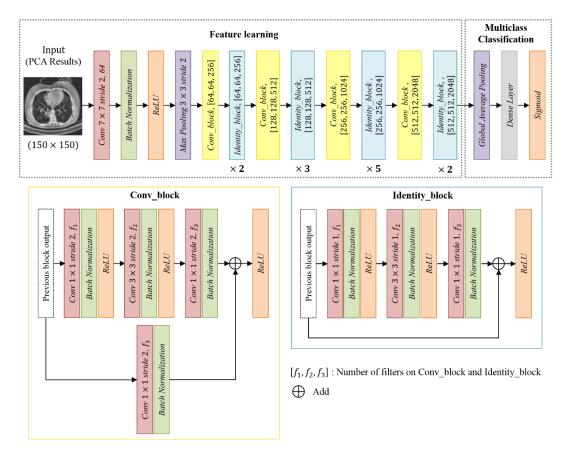

FIGURE 6. ResNet 50 Architecture for CT-Scan Image Classification

Figure 6 shows that the ResNet-50 architecture was built into two stages: feature learning and multi-class classification. First, the PCA image of 50×50 is used as input for the feature learning stage. Then the convolution operation process is carried out using a kernel measuring 7×7 stride two, and the number of filters is 64. After that, the feature map results from the convolution layer are normalized using batch normalization. The normalized feature map is used as input for the ReLU activation function. ReLU is used to overcome complexity and speed during the training process. Next, the dimension reduction process uses max polling 3×3 and stride 2. The results from the max poll will be used as input for the first convolution block using the number of filters, namely 64, 64, and 256. The convolution block consists of a convolution layer with a 1×1 kernel, batch normalization, a convolution layer with a 3×3 kernel, and ReLU. Besides that, the feature maps merging process also occurs using the add operation. The results of the first convolution block are then used as input for the identity block process using the number of filters, namely 64, 64, and 256. The identity block consists of a convolution layer with a 1×1 kernel, batch normalization, a convolution layer with a 3×3 kernel, and ReLU without the feature map merging process. For the second to fourth convolution blocks, the same process as the first convolution block is carried out by using the number of filters according to Figure 6. In the second to fourth identity blocks, the same process is

carried out as in the first identity block, only the number of filters differs. The feature map of the feature learning process is used as input for the multi-class classification stage. This stage consists of global average pooling, dense layer and sigmoid activation function. After the ResNet-50 model was formed, training and testing were continued using the architecture.

The training process used as many as 17,104 CT-Scan images of the lungs resulting from PCA. The data is divided into training and validation data, where the training data is 16,422 while the validation data is 682 images. This study, 50 epochs were used, 8 batch sizes and 2052 iterations. Loss (error) graphs and the accuracy of training results on training and validation data can be seen in Figure 7 and 8.
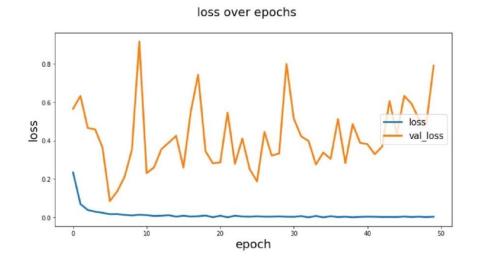

FIGURE 7. Graph of The Loss (Error) Value of The Resnet-50 Architecture Training Process

In Figure 7, it can be seen that the graph of the error (loss) value for the training data is getting closer to 0 for each epoch. The error (loss) value increases and decreases continuously in the graphic validation data until the 50th epoch. From the two graphs of the error value (loss), it can be concluded that the ResNet-50 architectural model is still overfitting.

From Figure 8, it can be seen that the graph of the accuracy value for the training data has increased continuously until the 50th epoch. In addition, Figure 8 also shows that the graph of accuracy values for unstable validation data always increases and decreases until the 50th epoch. From the two graphs of accuracy values, it can be concluded that the ResNet-50 architectural model is still overfitting. Overfitting models usually rely on training data, so it is not very easy to generalize to new data. During the training process, the best weights are stored for use at the testing stage.
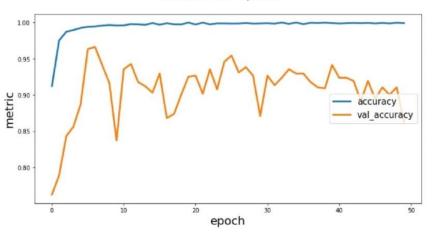
FIGURE 8. The Accuracy Graph of The Resnet-50 Architecture Training Process

In the testing stage, the ResNet-50 architectural model was tested using new data, namely test data, which amounted to 3365 CT-Scan images of the lungs. The data has been pre-processed first. The testing process results can be seen using the Receiver Operating Characteristics (ROC) curve in Figure 9.
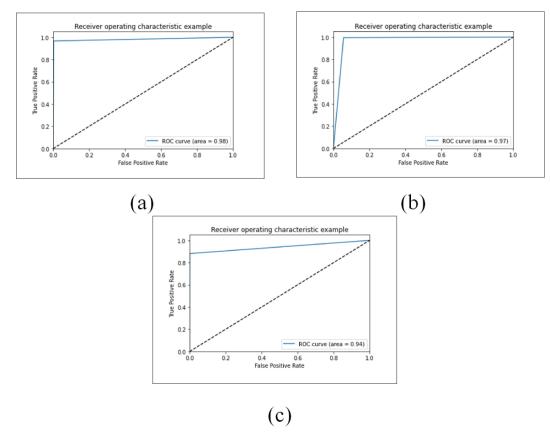

(a)

(b)

(c)

FIGURE 9. ROC Curve of The Results of The Resnet-50 Architecture Testing
Process (a) Covid (b) Normal (c) CAP

Figure 9 shows the ROC curve of the results of the ResNet50 architectural model testing process for three classes, namely covid, normal, and CAP. Figure 9(a) shows that the ROC graph in the Covid class has a long distance from the random classifier diagonal line than the normal and CAP classes. In Figure 9(c). The ROC graph for the CAP class is closest to the random classifier diagonal line compared to the other two classes. The area under the curve (AUC) value is calculated from the area under the ROC graph. If the AUC value is close to 1, the classification model has outstanding accuracy. Based on Figure 9(a), it can be seen that the covid class has the highest AUC value of 0.98. For the normal class, the AUC value gets the second highest position, 0.97, as shown in Figure 9(b). The CAP class obtains the smallest AUC value of 0.94, shown in Figure 9(c). Based on the AUC value obtained from the testing process, it can be concluded that the ResNet 50 architectural model has excellent accuracy in classifying Covid-19 using CT-Scan images of the lungs. The results obtained next are the comparison of the performance evaluation of the proposed method with the previous research, which is shown in Table 1 below:

TABLE 1.
Comparisons Proposed Method with the Previous Research

Methods	Year	Accuracy (%)	Recall (%)	Precision (%)	F1-Score (%)	Cohen's Kappa (%)
CNN and XGBoost [57]	2020	95,07	95,09	94,99	95	90
Corodet [58]	2021	94,2	92,5	94,94	91,32	-
ResNet and Data Augmentation [59]	2021	96	97	95	96	-
ResNet-50 [60]	2021	88,52	88,4	88	87,4	-
Xception[61]	2022	92,47	92,5	92,5	95	-
Deep Fitur Concatenation (DFC) [62]	2022	96,13	**97,04**	94,37	95,69	-
Proposed Method	**2022**	**96,64**	96,64	**96,82**	**96,64**	**94,50**

Table 1 shows that the proposed method has higher performance values for accuracy, precision, F1-score, and cohen's kappa values compared to other studies, which are above 94%. Saad et al. [62] obtained the highest precision value using the DFC method of 97.04%. From Table 1, it can be concluded that the accuracy of the PCA and ResNet-50 combination model for the classification of Covid-19 using CT-Scan images of the lungs is excellent, as seen from the accuracy value above 90%. The combination model of PCA and ResNet 50 has been excellent in classifying Covid-19, as seen from the precision value above 90%. The average harmonic of the precision and recall of the PCA and ResNet-50 combination model is very good, as can be seen from the F1-score value above 90%. A cohen's kappa value above 90% indicates that the results of the Covid-19 classification from the PCA and ResNet-50 combination model are by the class label that experts have given. From the performance value used in this study, it can be concluded that the PCA and ResNet-50 combination model is excellent in classifying Covid-19 using CT-Scan images of the lungs.

4. CONCLUSION

Based on the results and discussion, it can be concluded that the PCA method used for the pre-processing stage has worked optimally in reducing dimensions without losing essential features in the CT-Scan image of the lungs. The classification results using the ResNet-50 architecture have also worked very well in classifying Covid-19 based on the performance of accuracy, recall, precision, F1-score, and cohen's kappa, which are above 90%. Based on a comparison of several studies, the combination of the PCA method and the ResNet-50 architecture is most appropriate for classifying Covid-19 in the lung CT-scan dataset in terms of accuracy, recall, precision, F1-score, and cohen's kappa, which are above 90%.

ACKNOWLEDGEMENTS

REFERENCES

[1]   N. Zhu *et al.*, "A Novel Coronavirus from Patients with Pneumonia in China, 2019," *N. Engl. J. Med.*, vol. 382, no. 8, pp. 727–733, 2020, doi: 10.1056/nejmoa2001017.

[2]   K. Yuki, M. Fujiogi, and S. Koutsogiannaki, "COVID-19 pathophysiology: A review," *Clin. Immunol.*, vol. 215, no. April, 2020, doi: 10.1016/j.clim.2020.108427.

[3]   M. Rahimzadeh, A. Attar, and S. M. Sakhaei, "A Fully Automated Deep Learning-based Network For Detecting COVID-19 from a New And Large Lung CT Scan Dataset," *medRxiv*, pp. 1–38, 2020.

[4]   F. Song *et al.*, "Emerging Coronavirus 2019-nCoV Pneumonia.," *Radiology*, 2019.

[5]   S. Ahuja, B. K. Panigrahi, N. Dey, V. Rajinikanth, and T. K. Gandhi, "Deep transfer learning-based automated detection of COVID-19 from lung CT scan slices," *Appl. Intell.*, vol. 51, no. 1, pp. 571–585, 2021, doi: 10.1007/s10489-020-01826-w.

[6]   G. Kumar and P. K. Bhatia, "A detailed review of feature extraction in image processing systems," *Int. Conf. Adv. Comput. Commun. Technol. ACCT*, pp. 5–12, 2014, doi: 10.1109/ACCT.2014.74.

[7]   A. L. Aswathy, A. H. S., and V. C. Vinod, "COVID-19 diagnosis and severity detection from CT-images using transfer learning and back propagation neural network," *J. Infect. Public Health*, vol. 14, no. 10, pp. 1435–1445, 2021, doi: 10.1016/j.jiph.2021.07.015.

[8]   S. A. F. Sayed, A. M. Elkorany, and S. Sayed Mohammad, "Applying Different Machine Learning Techniques for Prediction of COVID-19 Severity," *IEEE Access*, vol. 9, pp. 135697–135707, 2021, doi: 10.1109/ACCESS.2021.3116067.

[9]   Z. Fang, W. Hu, M. Wang, R. Wang, S. Zhong, and S. Chen, "X-ray Absorption Spectroscopy Combined with Machine Learning for Diagnosis of Schistosomiasis Cirrhosis," *Biomed. Signal Process. Control*, vol. 60, 2020,

doi: 10.1016/j.bspc.2020.101944.

[10] J. Darvish and M. Ezoji, "Morphological Exudate Detection in Retinal Images using PCA-based Optic Disc Removal," *J. AI Data Min.*, vol. 7, no. 4, pp. 487–493, 2019, doi: 10.22044/JADM.2019.1488.

[11] Adiwijaya, U. N. Wisesty, E. Lisnawati, A. Aditsania, and D. S. Kusumo, "Dimensionality Reduction using Principal Component Analysis for Cancer Detection based on Microarray Data Classification Dimensionality Reduction using Principal Component Analysis for Cancer Detection based on Microarray Data Classification," vol. 14, no. 11, pp. 1521–1530, 2018, doi: 10.3844/jcssp.2018.1521.1530.

[12] M. Montalvo, M. Guijarro, J. M. Guerrero, and Á. Ribeiro, "Identification of Plant Textures in Agricultural Images," vol. 3, pp. 391–401, 2016, doi: 10.1007/978-3-319-32034-2.

[13] B. T. M. T., N. Mhamadian, M. Ghorbani, F. Khorshidi, F. Akbari, and C. Knaup, "Optimization of Brain Tumor MR Image Classification Accuracy Using Optimal Threshold, PCA and Training ANFIS with Different Repetitions," *J. Biomed. Phys. Eng.*, vol. 9, no. 2, pp. 189–198, 2019.

[14] J. D. Novaković, A. Veljović, S. S. Ilić, Ž. Papić, and T. Milica, "Evaluation of Classification Models in Machine Learning," *Theory Appl. Math. Comput. Sci.*, vol. 7, no. 1, p. Pages: 39 – 46, 2017.

[15] S. K. Sharma, U. K. Lilhore, S. Simaiya, and N. K. Trivedi, "An improved random forest algorithm for predicting the COVID-19 pandemic patient health," *Ann. Rom. Soc. Cell Biol.*, vol. 25, no. 1, pp. 67–75, 2021.

[16] B. A. Akinnuwesi *et al.*, "Application of intelligence-based computational techniques for classification and early differential diagnosis of COVID-19 disease," *Data Sci. Manag.*, vol. 4, no. December, pp. 10–18, 2021, doi: 10.1016/j.dsm.2021.12.001.

[17] A. A. Reshi *et al.*, "An Efficient CNN Model for COVID-19 Disease Detection Based on X-Ray Image Classification," *Complexity*, vol. 2021, 2021, doi: 10.1155/2021/6621607.

[18] M. Zhang, W. Li, and Q. Du, "Diverse region-based CNN for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2623–2634, 2018, doi: 10.1109/TIP.2018.2809606.

[19] J. Qin, W. Pan, X. Xiang, Y. Tan, and G. Hou, "A biological image classification method based on improved CNN," *Ecol. Inform.*, vol. 58, p. 101093, 2020, doi: 10.1016/j.ecoinf.2020.101093.

[20] M. M. Jadoon, Q. Zhang, I. U. Haq, S. Butt, and A. Jadoon, "Three-Class Mammogram Classification Based on Descriptive CNN Features," *Biomed Res. Int.*, vol. 2017, 2017, doi: 10.1155/2017/3640901.

[21] M. Sajjad, S. Khan, K. Muhammad, W. Wu, A. Ullah, and S. W. Baik, "Multi-grade brain tumor classification using deep CNN with extensive data augmentation," *J. Comput. Sci.*, vol. 30, pp. 174–182, 2019, doi: 10.1016/j.jocs.2018.12.003.

[22] R. Ghosh, "Automatic Detection and Classification of Diabetic Retinopathy stages using CNN," *2017 4th Int. Conf. Signal Process. Integr. Networks*, 2017, doi: 10.1109/SPIN.2017.8050011.

[23] M. Zhao, C. H. Chang, W. Xie, Z. Xie, and J. Hu, "Cloud Shape Classification System Based on Multi-Channel CNN and Improved FDM,"

IEEE Access, vol. 8, pp. 44111–44124, 2020, doi: 10.1109/ACCESS.2020.2978090.

[24] D. Sarwinda, R. H. Paradisa, A. Bustamam, and P. Anggia, "Deep Learning in Image Classification using Residual Network (ResNet) Variants for Detection of Colorectal Cancer," *Procedia Comput. Sci.*, vol. 179, no. 2019, pp. 423–431, 2021, doi: 10.1016/j.procs.2021.01.025.

[25] A. Mustafa and K. Meehan, "Gender Classification and Age Prediction using CNN and ResNet in Real-Time," *2020 Int. Conf. Data Anal. Bus. Ind. W. Towar. a Sustain. Econ. ICDABI 2020*, 2020, doi: 10.1109/ICDABI51230.2020.9325696.

[26] J. Zhu, Y. Zhang, and Q. Zhao, "Atrial Fibrillation Detection using Different Duration ECG Signals with SE-ResNet," *IEEE 21st Int. Work. Multimed. Signal Process. MMSP 2019*, pp. 1–5, 2019, doi: 10.1109/MMSP.2019.8901729.

[27] S. Guo and Z. Yang, "Multi-Channel-ResNet: An integration framework towards skin lesion analysis," *Informatics Med. Unlocked*, vol. 12, no. June, pp. 67–74, 2018, doi: 10.1016/j.imu.2018.06.006.

[28] M. Guo and Y. Du, "Classification of Thyroid Ultrasound Standard Plane Images using ResNet-18 Networks," *Proc. Int. Conf. Anti-Counterfeiting, Secur. Identification, ASID*, vol. 2019-Octob, no. i, pp. 324–328, 2019, doi: 10.1109/ICASID.2019.8925267.

[29] A. Tahir *et al.*, "Coronavirus : Comparing COVID-19 , SARS, and MERS in The Eyes of AI," *arXiv*, pp. 1–35, 2021.

[30] K. Padmavathi and K. Thangadurai, "Implementation of RGB and grayscale images in plant leaves disease detection - Comparative study," *Indian J. Sci. Technol.*, vol. 9, no. 6, pp. 4–9, 2016, doi: 10.17485/ijst/2016/v9i6/77739.

[31] A. A. Siddiqi, G. B. Narejo, M. Tariq, and A. Hashmi, "Investigation of Histogram Equalization Filter for CT Scan Image Enhancement," *Biomed. Eng. Appl. Basis Commun.*, vol. 31, no. 5, pp. 1–10, 2019, doi: 10.4015/S1016237219500388.

[32] I. T. Jollife and J. Cadima, "Principal Component Analysis: A Review and Recent Developments," *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.*, vol. 374, no. 2065, pp. 1–16, 2016, doi: 10.1098/rsta.2015.0202.

[33] A. Saygili, "A New Approach for Computer-Aided Detection of Coronavirus (COVID-19) from CT and X-Ray Images Using Machine Learning Methods," *Appl. Soft Comput.*, vol. 105, 2021, doi: 10.1016/j.asoc.2021.107323.

[34] H. Zhao, J. Zheng, J. Xu, and W. Deng, "Fault Diagnosis Method based on Principal Component Analysis and Broad Learning System," *IEEE Access*, vol. 7, pp. 99263–99272, 2019, doi: 10.1109/ACCESS.2019.2929094.

[35] W. Yang, Y. Zhao, D. Wang, H. Wu, A. Lin, and L. He, "Using Principal Components Analysis and IDW Interpolation to Determine Spatial and Temporal Changes of Surface Water Quality of Xin'anjiang River in Huangshan, China," *Int. J. Environ. Res. Public Health*, vol. 17, no. 8, pp. 1–14, 2020, doi: 10.3390/ijerph17082942.

[36] S. Wang, J. Wang, and Y. Han, "Pattern Matching Strategy Based on Optimized Fuzzy Clustering Algorithm for Industrial Internet," in *Proceedings of the 38th Chinese Control Conference July*, 2019, pp. 7005–7009, doi: 10.23919/ChiCC.2019.8866536.

[37] Y. Zhang, B. Zhang, and Z. Wu, "Multi-Model Modeling of CFB Boiler Bed

Temperature System Based on Principal Component Analysis," *IEEE Access*, vol. 8, pp. 389–399, 2020, doi: 10.1109/ACCESS.2019.2961414.

[38] R. Zhang, T. Du, and S. Qu, "A Principal Component Analysis Algorithm Based on Dimension Reduction Window," *IEEE Access*, vol. 6, pp. 63737–63747, 2018, doi: 10.1109/ACCESS.2018.2875270.

[39] M. Alkhayrat, M. Aljnidi, and K. Aljoumaa, "A Comparative Dimensionality Reduction Study in Telecom Customer Segmentation using Deep Learning and PCA," *J. Big Data*, vol. 7, no. 1, 2020, doi: 10.1186/s40537-020-0286-0.

[40] G. Marques, D. Agarwal, and I. de la T. Díez, "Automated medical diagnosis of COVID-19 through EfficientNet convolutional neural network," *Appl. Soft Comput. J.*, vol. 96, p. 106691, 2020, doi: 10.1016/j.asoc.2020.106691.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[42] L. Alzubaidi *et al.*, "Review of Deep Learning: Concepts , CNN Architectures , Challenges , Applications , Future Directions," *J. Big Data*, vol. 8, no. 53, 2021, doi: 10.1186/s40537-021-00444-8.

[43] E. Paladini, E. Vantaggiato, F. Bougourzi, C. Distante, A. Hadid, and A. Taleb-Ahmed, "Two Ensemble-CNN Approaches for Colorectal Cancer Tissue Type Classification," *J. Imaging*, vol. 7, no. 3, p. 51, 2021, doi: 10.3390/jimaging7030051.

[44] T. A. Soomro *et al.*, "Deep Learning Models for Retinal Blood Vessels Segmentation: A Review," *IEEE Access*, vol. 7, pp. 71696–71717, 2019, doi: 10.1109/ACCESS.2019.2920616.

[45] S. Bharati, P. Podder, M. R. H. Mondal, and V. B. S. Prasath, "CO-ResNet: Optimized ResNet model for COVID-19 diagnosis from X-ray images," *Int. J. Hybrid Intell. Syst.*, vol. 17, no. 1–2, pp. 71–85, 2021, doi: 10.3233/his-210008.

[46] W. Chen, B. Yang, J. Li, and J. Wang, "An Approach to Detecting Diabetic Retinopathy based on Integrated Shallow Convolutional Neural Networks," *IEEE Access*, vol. 8, pp. 178552–178562, 2020, doi: 10.1109/ACCESS.2020.3027794.

[47] H. Gholamalinezhad and H. Khosravi, "Pooling Methods in Deep Neural Networks, a Review," *arXiv*, 2020, [Online]. Available: http://arxiv.org/abs/2009.07485.

[48] Y. Tian, "Artificial Intelligence Image Recognition Method Based on Convolutional Neural Network Algorithm," *IEEE Access*, vol. 8, pp. 125731–125744, 2020, doi: 10.1109/ACCESS.2020.3006097.

[49] D. R. Nayak, N. Padhy, P. K. Mallick, M. Zymbler, and S. Kumar, "Brain Tumor Classification Using Dense Efficient-Net," *Axioms*, vol. 11, no. 1, 2022, doi: 10.3390/axioms11010034.

[50] A. Raza, A. Mehmood, S. Ullah, M. Ahmad, G. S. Choi, and B. W. On, "Heartbeat Sound Signal Classification Using Deep Learning," *Sensors (Switzerland)*, vol. 19, no. 21, pp. 1–15, 2019, doi: 10.3390/s19214819.

[51] X. Deng, Q. Liu, Y. Deng, and S. Mahadevan, "An Improved Method to Construct basic Probability Assignment based on The Confusion Matrix for Classification Problem," *Inf. Sci. (Ny).*, vol. 340–341, pp. 250–261, 2016, doi:

10.1016/j.ins.2016.01.033.

[52] M. Rahman, D. Islam, R. J. Mukti, and I. Saha, "A Deep Learning Approach based on Convolutional LSTM for Detecting Diabetes," *Comput. Biol. Chem.*, vol. 88, p. 107329, 2020, doi: 10.1016/j.compbiolchem.2020.107329.

[53] A. Gupta, Anjum, S. Gupta, and R. Katarya, "InstaCovNet-19: A Deep Learning Classification Model for The Detection of COVID-19 Patients Using Chest X-ray," *Appl. Soft Comput.*, vol. 99, no. xxxx, p. 106859, 2021, doi: 10.1016/j.asoc.2020.106859.

[54] I. R. I. Haque and J. Neubert, "Deep learning approaches to biomedical image segmentation," *Informatics Med. Unlocked*, p. 100297, 2020, doi: 10.1016/j.imu.2020.100297.

[55] D. Chicco, M. J. Warrens, and G. Jurman, "The Matthews Correlation Coefficient (MCC) is More Informative Than Cohen's Kappa and Brier Score in Binary Classification Assessment," *IEEE Access*, vol. 9, pp. 78368–78381, 2021, doi: 10.1109/ACCESS.2021.3084050.

[56] M. Heidari, S. Mirniaharikandehei, and A. Zargari, "Improving The Performance of CNN to Predict The Likelihood of COVID-19 Using Chest X-Ray Images with Preprocessing Algorithms," *Int. J. Med. Inform.*, vol. 144, p. 104284, 2020, doi: 10.1016/j.ijmedinf.2020.104284.

[57] E. D. Carvalho and E. D. Carvalho, "Diagnosis of COVID-19 in CT image using CNN and XGBoost," in *2020 IEEE Symposium on Computers and Communications (ISCC)*, 2020.

[58] N. Science *et al.*, "CoroDet : A deep learning based classification for COVID-19 detection using chest X-ray images," *Chaos, Solitons & Fractals*, vol. 142, p. 110945, 2021.

[59] A. Halder and B. Datta, "COVID-19 detection from lung CT-scan images using transfer learning approach," *Mach. Learn. Sci. Technol.*, vol. 2, no. 4, 2021.

[60] N. Kumar, P. Singh, and S. Dutt, "Automated detection of COVID-19 from CT scan using convolutional neural network," *Biocybern. Biomed. Eng.*, vol. 41, pp. 572–588, 2021.

[61] S. V Kogilavani *et al.*, "COVID-19 Detection Based on Lung Ct Scan Using Deep Learning Techniques," *Comput. Math. Methods Med.*, vol. 2022, p. 7672196, 2022, doi: 10.1155/2022/7672196.

[62] W. Saad, W. A. Shalaby, M. Shokair, F. Abd, E. Samie, and M. Dessouky, "COVID-19 classification using deep feature concatenation technique," *J. Ambient Intell. Humaniz. Comput.*, vol. 13, no. 4, pp. 2025–2043, 2022, doi: 10.1007/s12652-021-02967-7.