

# NAGNN: Classification of COVID-19 based on neighboring aware representation from deep graph neural network

Siyuan Lu<sup>1</sup>  | Ziquan Zhu<sup>2</sup>  | Juan Manuel Gorriz<sup>3</sup>  |  
Shui-Hua Wang<sup>4</sup>  | Yu-Dong Zhang<sup>1</sup> 

<sup>1</sup>School of Informatics, University of Leicester, Leicester, UK

<sup>2</sup>Science in Civil Engineering, University of Florida, Gainesville, FL, USA

<sup>3</sup>Department of Signal Theory, Networking and Communications, University of Granada, Granada, Spain

<sup>4</sup>School of Mathematics and Actuarial Science, University of Leicester, Leicester, UK

## Correspondence

Shui-Hua Wang, School of Mathematics and Actuarial Science, University of Leicester, Leicester LE1 7RH, UK.

Email: [shuihuawang@ieee.org](mailto:shuihuawang@ieee.org)

Yu-Dong Zhang, School of Informatics, University of Leicester, Leicester LE1 7RH, UK.

Email: [yudongzhang@ieee.org](mailto:yudongzhang@ieee.org)

## Funding information

Royal Society International Exchanges Cost Share Award (GB), Grant/Award Number: RP202G0230; Hope Foundation for Cancer Research, Grant/Award Number: RM60G0680; Medical Research Council Confidence in Concept Award, UK, Grant/Award Number: MC\_PC\_17171; Global Challenges Research Fund (GCRF), UK, Grant/Award Number: P202PF11

## Abstract

COVID-19 pneumonia started in December 2019 and caused large casualties and huge economic losses. In this study, we intended to develop a computer-aided diagnosis system based on artificial intelligence to automatically identify the COVID-19 in chest computed tomography images. We utilized transfer learning to obtain the image-level representation (ILR) based on the backbone deep convolutional neural network. Then, a novel neighboring aware representation (NAR) was proposed to exploit the neighboring relationships between the ILR vectors. To obtain the neighboring information in the feature space of the ILRs, an ILR graph was generated based on the  $k$ -nearest neighbors algorithm, in which the ILRs were linked with their  $k$ -nearest neighboring ILRs. Afterward, the NARs were computed by the fusion of the ILRs and the graph. On the basis of this representation, a novel end-to-end COVID-19 classification architecture called neighboring aware graph neural network (NAGNN) was proposed. The private and public data sets were used for evaluation in the experiments. Results revealed that our NAGNN outperformed all the 10 state-of-the-art methods in terms of generalization ability. Therefore, the proposed NAGNN is effective in detecting COVID-19, which can be used in clinical diagnosis.

**KEYWORDS**

convolutional neural network, graph convolutional network, random vector functional link net, transfer learning

## 1 | INTRODUCTION

COVID-19, named by World Health Organization, is a highly infectious pandemic that started in December 2019.<sup>1</sup> Caused by a new coronavirus, COVID-19 is can be transmitted from human to human.<sup>2</sup> What's worse is that this coronavirus evolved into new variants in 2021, such as the delta variant and lambda variant, which can be more infectious. Accurate diagnosis is an indispensable step to control this disaster by isolating the patients.

There are generally two types of diagnosis methods for COVID-19.<sup>3</sup> The most prevailing one is the real-time reverse transcription-polymerase chain reaction (rRT-PCR) test on the samples from the nasopharyngeal swab.<sup>4</sup> The time to obtain the diagnosis result ranges from hours to two days. The other type of diagnosis is based on the analysis of chest computed tomography (CT). The main difference between COVID-19 cases and normal controls is the ground-glass opacity, which can be revealed in CT images.<sup>5</sup> Compared with rRT-PCR testing, CT offers good visualization of the disease.

However, the rRT-PCR is in short supply, and the CT analysis suffers from high inter-observer and intraobserver variance; neither is optimal. Computer-aided diagnosis (CAD) for medical image analysis can assist radiologists in diagnosis based on artificial intelligence.<sup>6</sup> The trained CAD systems can generate predictions from medical images, and the diagnosis process is high in reproducibility compared with manual analysis.<sup>7</sup> The predictions by CAD systems can give guidance and verification for final predictions. The recent breakthrough in deep learning enables CAD systems to achieve better diagnosis accuracy.<sup>8,9</sup> For example, Yu et al.<sup>10</sup> used four pretrained deep models, including InceptionV3, ResNet-50, ResNet-101, and DenseNet-201 for feature extraction from chest CT images,<sup>11,12</sup> and trained five classical machine learning models for classification and comparison, including linear discriminant, linear support vector machine (SVM), cubic SVM,  $k$ -nearest neighbors ( $k$ -NN), and Adaboost decision tree. From the experiment, they discovered that the combination of DenseNet-201 + cubic SVM outperformed others with the accuracy of 95.20% by 10-fold cross-validation. Transfer learning-based feature extraction can generate high-level image representations, which can be helpful for classification,<sup>13,14</sup> but backbone model selection and the tedious parameter tuning pose a challenge in real-world applications. Also, the distribution of these features and their relationship in the latent space can be exploited to generate more robust image representations because the feature vector is likely to share some common characteristics with its neighbors. To address these problems, in this study, we present a new CAD method for COVID-19. The *contributions* of this paper are

- i. A novel end-to-end COVID-19 classification architecture called neighboring aware graph neural network (NAGNN) was proposed.
- ii. A novel image-level representation (ILR) learning algorithm was proposed, which is capable of adaptively finding the optimal backbone model as well.
- iii. A novel universal neighboring-aware representation (NAR) learning framework that can be used for any image recognition task was proposed.

- iv. A novel graph random vector functional link (GRVFL) was proposed.
- v. Our NAGNN outperformed state-of-the-art approaches based on experiment results from both our private data set and the public SARS-COV-2 Ct-Scan Data set.

The rest of this study is organized as follows. Section 2 presents related work. Section 3 provides the detailed methodology, including transfer learning, neighboring aware representation (NAR), and GRVFL. The experiment design is provided in Section 4. Results and discussion are presented in Section 5. Finally, the conclusion is given in Section 6.

## 2 | RELATED WORK

In this section, we will discuss the recently published COVID-19 detection methods based on medical images and machine learning algorithms, as well as the feature extraction methods in computer vision.

### 2.1 | COVID-19 detection

So far, researchers and practitioners have developed a bunch of CAD systems for COVID-19 detection based on classical machine learning as well as deep learning models and have made significant progress. Ucar and Korkmaz<sup>15</sup> proposed their diagnosis scheme for COVID-19. The pretrained SqueezeNet was employed as the backbone model. They tuned the SqueezeNet with Bayesian optimization on the X-ray images. Their Bayes-SqueezeNet achieved 98.26% accuracy in the experiment. Ozturk et al.<sup>16</sup> implemented automated COVID-19 classification by transfer learning of DarkNet and produced 98.08% accuracy for binary identification. Apostolopoulos et al.<sup>17</sup> proposed to train the MobileNetV2 from scratch and achieved an accuracy of 99.18%. Later, Apostolopoulos and Mpesiana<sup>18</sup> used transfer learning to detect COVID-19 and tested several famous models, including ResNet, Visual Geometry Group (VGG), MobileNet, Inception, and Xception. Zhou et al.<sup>19</sup> proposed a standard three-dimensional (3D) embedding method to transform a 3D lung CT image into the embedding space. They implemented 3D image segmentation by three 2D image segmentations. Yasar and Ceylan<sup>20</sup> developed a 23 layered convolutional neural network (CNN) and trained it for the classification of COVID-19 and normal samples. Wang et al.<sup>5</sup> made an improvement on the COVID-Net model in mainly two ways. They proposed to leverage separate feature normalization on the top of the model to solve the domain shifting problem. Additionally, a contrastive training objective was formed to improve the domain invariance performance. Experiment results from two public COVID-19 data sets revealed that there is an obvious increase for the redesigned model compared with the original COVID-Net. Wang et al.<sup>2</sup> first used a pretrained UNet to generate lung regions from 3D chest CT images. Then, a 3D CNN was trained to predict the probabilities of infected regions. The predictions were refined by connected components analysis. Waheed et al.<sup>21</sup> suggested using a generative adversarial network (GAN) to obtain more chest X-ray images for training CNN. The proposed model named COVIDGAN can improve the CNN in the classification of COVID-19 from 85% to 95% accuracy in comparison experiments. Togacar et al.<sup>22</sup> developed a hybrid COVID-19 classification system. They proposed to use a fuzzy color algorithm to generate structured images from the chest X-ray images and stacked them with the original images. Then, MobileNetV2 and SqueezeNet were employed to obtain features. The two feature sets

were fused by social mimic optimization. Finally, an SVM was trained for classification. Sun et al.<sup>23</sup> proposed to utilize deep forest to get representations from chest CT images and implement adaptive feature selection. Sakib et al.<sup>24</sup> employed both GAN and conventional image augmentation methods to generate the augmented data set and trained the customized CNN model. Roy et al.<sup>4</sup> put forward a COVID-19 detection approach for chest ultrasound images. Their classification model was based on spatial transformer networks, trained by a weakly supervised method. Their system can predict segmentation results, image labels as well as video labels. Ouyang et al.<sup>25</sup> used an online attention module to improve the classification performance of 3D CNN in detecting COVID-19 from community-acquired pneumonia. To solve the distribution imbalance problem in the data set, a dual sampling (DS) algorithm was employed. Oh et al.<sup>26</sup> proposed a patch-based CNN for COVID-19 detection with insufficient training data. They first segmented the chest X-ray images to obtain lung contours. Then, a set of random patches was extracted to fine-tune a set of pretrained ResNet-18 models. Finally, the predictions were determined by majority voting of the outputs from the ResNet-18 models. Li et al.<sup>27</sup> suggested to use self-supervised strategy to train the classifier, and the soft labels were obtained based on the distances between the sample features. Horry et al.<sup>7</sup> studied the COVID-19 detection performance using different medical imaging modalities, including CT, ultrasound, and X-ray images. They chose VGG as the backbone model and trained it with the three data sets. Testing results revealed that ultrasound images provided better precision than CT and X-ray images. The shortcomings of the abovementioned methods are presented in Table 1.

## 2.2 | Feature extraction

Feature extraction from images is an important and necessary procedure for image classification and recognition because the distribution of the features directly determines the complexity of the classification problem. If the interclass variance is high and intraclass variance is low, the classification can be implemented well with simple models, such as linear classification algorithms, vice versa. On the other hand, there is massive information in a digital image, but only a part of the information is useful for recognition and classification, while some can hinder the classification. Therefore, feature extraction or representation generation can retain useful information from images while eliminating the other factors.

Generally, image features fall into two main categories: handcrafted features and deep learning-based features. Handcrafted features are commonly seen in traditional machine learning algorithms, which are based on the calculation of statistics and patterns. These features usually have some advantages, such as gray level invariant and rotation invariant. For instance, the local binary pattern (LBP),<sup>28,29</sup> defined in a  $3 \times 3$  patch, considers the relationships of the eight points and the central pixel. The expression is given as

$$\text{LBP}(x_c, y_c) = \sum_{p=0}^{P-1} 2^p s(i_p - i_c), \quad (1)$$

where  $(x_c, y_c)$  denotes the coordinate of central pixel,  $i_c$  represents the gray level intensity value of the central pixel,  $i_p$  stands for the intensity value of the neighboring pixel, and  $s$  is a step function.

The disadvantage of the original LBP is that it is calculated in a fixed size of the local perceptive field, so it fails when the texture is larger. Later, to overcome this drawback, improved LBP was proposed, such as uniform LBP.

**TABLE 1** Shortcomings in state-of-the-art methods

Methods	Shortcomings
Ucar and Korkmaz <sup>15</sup>	The data set was class-imbalanced with less than 100 COVID-19 samples but over 1000 normal controls
Ozturk et al. <sup>16</sup>	Their data set was class-imbalanced and small to train the deep CNN model for classification
Apostolopoulos et al. <sup>17</sup>	They only trained and tested MobileNetV2 for classification with a class-imbalanced data set
Apostolopoulos and Mpesiana <sup>18</sup>	There are much more pneumonia and normal samples than COVID-19 samples in their data set
Zhou et al. <sup>19</sup>	Their data set was too small to train the deep 3D segmentation model
Yasar and Ceylan <sup>20</sup>	The sensitivity of the proposed method was 94.04%, which was relatively low
Wang et al. <sup>5</sup>	The resolution of their images varied in a wide range
Wang et al. <sup>2</sup>	Their data set was small to train the deep CNN model
Waheed et al. <sup>21</sup>	Their data set was too small and class-imbalanced
Togacar et al. <sup>22</sup>	The data set was class-imbalanced with less than 500 samples in total
Sun et al. <sup>23</sup>	Their method can only classify COVID-19 and community-acquired pneumonia (CAP) because there are no normal samples in the data set
Sakib et al. <sup>24</sup>	The reported accuracy was 93.94%, which was low
Roy et al. <sup>4</sup>	Their data set contained noisy labels
Ouyang et al. <sup>25</sup>	The accuracy of the proposed system was not satisfactory
Oh et al. <sup>26</sup>	Their method was only evaluated with a hold-out validation
Li et al. <sup>27</sup>	They did not compare the performance of different backbone models
Horry et al. <sup>7</sup>	The data sets in their experiments were curated and small, and it can cause overfitting to training VGG with small data sets for classification

Abbreviations: 3D, three-dimensional; CNN, convolutional neural network; VGG, Visual Geometry Group.

On the other hand, deep learning-based features are prevailing with the advent of deep CNN models. Handcrafted features are usually invented by sophisticated formulations, so they are interpretable. However, deep learning-based features are extracted from trained CNN models. Specifically, to extract features, we need to first train the CNN model on the image data set. Then, the output tensor of a certain layer in the CNN is computed as the features. It can be found that deep learning-based features require less manual intervention and are easy to obtain. As they are derived from CNN, these features are usually effective for image classification. For example, in research,<sup>30</sup> deep features were extracted from multiple deep CNN models and fed into multiple SVMs for COVID-19 classification. The best advantage of deep learning-based features is that it is convenient to implement because it requires no prior knowledge or predesigned patterns. However, deep learning-based features are less interpretable than handcrafted features.

Currently, many deep learning-based medical image analysis methods simply transfer the off-the-shelf CNN models and train them with medical image data sets.<sup>7,21</sup> Domain shifting is an important issue for transfer learning, especially for medical image analysis, because the features

in the medical images and the distribution of the medical images are usually intensely different from those in the ImageNet data set, which is used to train deep CNN models. Domain adaptation is an effective tool to handle this shifting, including reweighting the samples from the source domain, search common representation spaces between the source domain and the target domain, and so forth on the other side, the relationship between the features is often ignored, while this relationship is significant for classification. Graph convolutional network (GCN)<sup>31,32</sup> is proposed for addressing graph data, such as social networks. In graph data, the samples are related to each other, so the linked samples share some common interests. Therefore, robust representations can be generated if the relations among the samples can be utilized.

From the above analysis, we discovered that deep CNN models achieved good classification performance, either trained from scratch or by transfer learning. However, most methods simply used single ILRs from a CNN model. In the latent feature space, the distribution of image features is the most significant factor for classification and diagnosis. Generally, borrowing information from the neighboring image representations can be beneficial for classification, because the samples in a neighborhood are likely to have the same label. Therefore, NARs can be more robust and accurate in classification compared with ILR.

### 2.3 | Neighboring aware networks (NANs)

Recently, the NAN has received increasing attention from academia, and it is often applied in recommendation systems.<sup>33–35</sup> Because, in recommendation systems, the input data are graphs, where there are data nodes and connections naturally. However, in the medical image classification, the images are isolated because there are no built-in connections between the images, and the patients are usually strangers in real-world situations. Admittedly, there is also other information about patients, such as gender, age, and so forth. But the gender and age are not indispensably related to each other at the sample level.

Therefore, the challenge lies in the generation of the relationships in the data nodes when applying a neighboring aware mechanism for image classification. It is not suitable to link the images directly based on their similarities since similar medical images do not necessarily share the same label. Because the lesions and focuses of diseases may only account for a small region in the images, and the interpatient variance of the images can be high. To overcome this issue, we propose to generate the graph in latent feature space instead of in image level. Because we observe that the image features from backbone CNN models can be highly discriminant, so they are closely related to the classification results. Hence, the neighboring relationships between the feature nodes can contribute to the final classification performance. On the basis of this observation, we propose the NAGNN for COVID-19 detection.

## 3 | METHODOLOGY

We present a new COVID-19 classification framework called NAGNN for chest CT images in this paper. The overview of the proposed NAGNN is demonstrated in Figure 1. It can be easily observed that there are three main components: ILR learning, NAR, and GRVFL training. First of all, we proposed to modify those pretrained backbone models to implement ILR learning and find the optimal backbone model simultaneously. Then, we propose to embed the neighboring information among the ILRs. The graphs are generated based on the  $k$ -NN algorithm for the

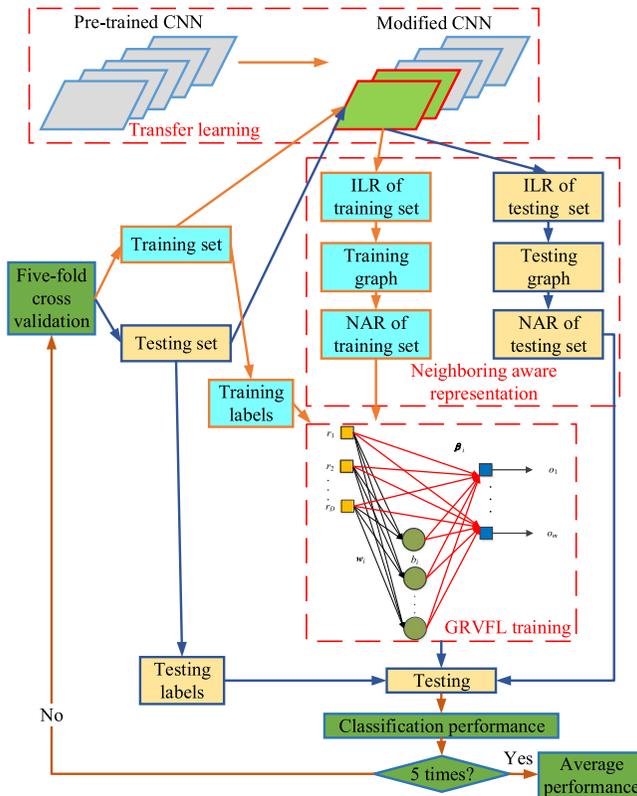


FIGURE 1 An overview of the COVID-19 detection system. CNN, convolutional neural network; ILR, image-level representation; NAR, neighboring aware representation [Color figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

training and testing set, respectively, and the NARs are generated. Finally, the GRVFL is constructed and trained for classification. The fivefold cross-validation is employed to evaluate the generalization performance of our method. Gradient-weighted class activation mapping (Grad-CAM) was leveraged to reveal the heat map of the NAGNN for explanation and interpretation. A detailed discussion of the proposed method is presented in the rest of this section.

### 3.1 | ILR learning

Traditional machine learning systems usually require handcrafted features and classifier training. A major advantage of CNN models is that they provide a means of automated feature learning ability as well as classification. Unfortunately, it is time-consuming to train deep CNN models. Also, dedicated graphics processing units (GPUs) are required for training, which is expensive for users. To mitigate this gap, transfer learning is adopted. A state-of-the-art CNN model pretrained on the ImageNet data set is capable of producing good predictions on 1000 categories of labels. This also indicates that the pretrained CNN model has already gained the ability to generate latent image representations, which are beneficial for classification, and the representation learning ability can be transferred to other image data sets. Specifically, for COVID-19 detection, we can transfer a deep CNN model to our COVID-19 data set which contains two categories: COVID-19 and normal. Transfer learning frees

users from hyperparameter tuning as the CNN models are pretrained. Another advantage of transfer learning is that it enables the pretrained CNN model to have good classification performance within a few epochs, which is time-efficient. However, overfitting is the main problem when transferring deep models to small data sets. Because medical image data sets are usually smaller than the ImageNet data set in terms of both the number of classes and the number of samples in each class. To handle this issue, we first employ the early stopping strategy to set a small value to the max epochs for fine-tuning the pretrained CNN on the COVID-19 data set. In addition, the fine-tuned CNN only serves as the feature extractor instead of the classifier in our CAD system. With the two strategies, we believe that the overfitting problem can be effectively avoided.

To implement transfer learning with a pretrained CNN model, some modifications should be made. First of all, the ImageNet data set contains 1000 categories of samples so the pretrained CNN has 1000 output nodes, but the COVID-19 data set contains only two categories of images, so the number of output nodes is modified as 2. Then, the pretrained CNN has two fully connected layers, and the node numbers are (4096, 1000). We propose to insert a fully connected layer between the original two fully connected layers, and the node dimensions are set as (4096, 256, 2). The extra inserted fully connected layer aims to gradually reduce the feature dimension. Because the original mapping was 4096–1000, but the modified mapping was 4096 to merely 2. The added ‘FC256’ can be a buffer layer to shrink the feature dimension less dramatically.

After tuning the modified CNN model on COVID-19 data, the ILR can be extracted easily by activating the ‘FC256’ layer. Detailed steps for ILR generation by transfer learning are summarized in Proposed Algorithm 1. As the fine-tuned CNN models can provide predictions on the testing set as well, we tested several state-of-the-art CNN models to select the best model based on testing accuracy as the ILR generation model in this study.

### **Proposed Algorithm 1** Adaptive CNN-model selection-based image-level representation learning

---

Phase 1: Transferring a set of pretrained CNN models

---

For each CNN model in the set

Step 1: Load a state-of-the-art CNN model pretrained on ImageNet data set

Step 2: Replace the ‘FC1000’ with ‘FC2’

Step 3: Insert ‘FC256’ and ‘ReLU activation’ between the original two fully connected layers

Step 4: Fine-tune the modified CNN model on the COVID-19 training set

Step 5: Save the CNN model

End for

Phase 2: Adaptive model selection

For each fine-tuned CNN model

Step 6: Load a fine-tuned model

Step 7: Obtain the classification accuracy of the CNN model on the testing set

End for

Step 8: Compare the testing accuracy and get the best-fine-tuned model

Phase 3: ILR generation by the best fine-tuned CNN model

Step 9: Load the best fine-tuned CNN model

Step 10: Feed the COVID-19 training set and testing set into the model

Step 11: Activate the ‘FC256’ layer with the COVID-19 training set and testing set

Step 12: Save the activations as training ILR and testing ILR

---

### 3.2 | NAR learning

We propose a novel NAR to improve the robustness because the information in the neighborhood of an ILR is helpful for the classification. An ILR vector is likely to share some common characteristics with its neighbors. This neighboring relationship between the ILRs in the latent feature space can be fused to get a better representation. However, there are no off-the-shelf methods to generate NAR as the ILRs are extracted from individual images. Hence, we propose to leverage graph theory to implement NAR extraction, where the ILRs are defined as the nodes in a graph and are linked. The algorithm to link the ILR nodes is the  $k$ -NN. Suppose we have the training set ILRs as  $\mathbf{R}_{\text{IL}} = [r_1, r_2, r_3, \dots, r_N]^T \in \mathbb{R}^{N \times D}$ , where  $N$  denotes the number of training samples and  $D$  stands for the dimension of ILR, the graph generation steps are presented in the Proposed Algorithm 2.

#### Proposed Algorithm 2 Generation of graph based on ILRs

---

Step 1: Load the ILRs of the training set

---

For each ILR,

Step 2: Calculate the Euclidean distances between the ILR and all the other ILRs

Step 3: Sort the distances, and find the  $k$ -nearest neighbors of the ILR

Step 4: Save the distance vector of the  $k$ -nearest neighbors for the ILR

End for

---

Now, we obtained both the ILRs of the training set and the graph, the distance matrix  $\mathbf{Dst}$  and the adjacent matrix  $\mathbf{Adt}$  can be computed by

$$\mathbf{Dst}(i, j) = \|r_i - r_j\|, \quad 1 \leq i, j \leq N \quad \text{and} \quad i \neq j, \quad (2)$$

$$\mathbf{Adt}(i, j) = 1, \quad \text{if } r_j \in \text{kNN}(r_i), \quad 1 \leq i, j \leq N \quad \text{and} \quad i \neq j, \quad (3)$$

where both  $\mathbf{Dst}$  and  $\mathbf{Adt}$  are initialized with zeros, and the  $\text{kNN}(r_i)$  denotes the  $k$ -nearest neighbors of  $r_i$  which can be obtained by the  $\mathbf{Dst}$ . To obtain the normalized  $\widehat{\mathbf{Adt}}$ , we have

$$\widehat{\mathbf{Adt}} = \mathbf{Degree}^{-\frac{1}{2}}(\mathbf{Adt} + \mathbf{E})\mathbf{Degree}^{-\frac{1}{2}}, \quad (4)$$

where  $\mathbf{E}$  is the identity and the  $\mathbf{Degree}$  can be calculated by

$$\mathbf{Degree}(i, j) = \begin{cases} k, & \text{if } i = j, \\ 0, & \text{if } i \neq j, \end{cases} \quad 1 \leq i, j \leq N. \quad (5)$$

Finally, the NAR set  $\mathbf{R}_{\text{NA}}$  can be generated by

$$\mathbf{R}_{\text{NA}} = \widehat{\mathbf{Adt}} \cdot \mathbf{R}_{\text{IL}}. \quad (6)$$

A summary of NAR generation is shown in Proposed Algorithm 3 and Figure 2. Similarly, the graph and the NAR of the testing set can be generated by these operations. These NARs are fed into a novel GRVFL for training and classification.

**Proposed Algorithm 3** NAR generation

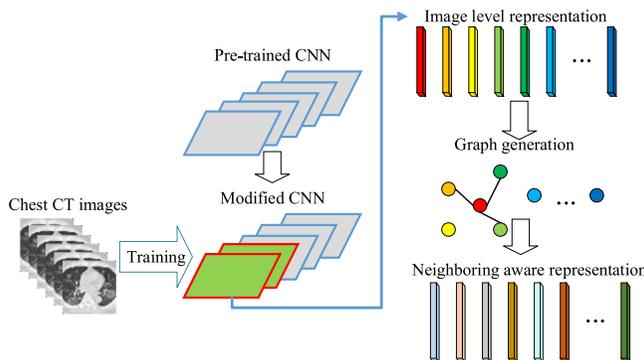
- 
- Step 1: Load the state-of-the-art CNN model pretrained on ImageNet

---

  - Step 2: Modify the top layers in the CNN model according to the COVID-19 data
  - Step 3: Fine-tune the CNN on the COVID-19 training set
  - Step 4: Obtain the ILR set by activating the layer ‘FC256’ in the CNN
  - Step 5: Find the  $k$ -NNs for each ILR node
  - Step 6: Generate the graph for the ILR set based on the  $k$ -NNs obtained
  - Step 7: Compute the distance matrix and adjacent matrix for the graph by Equations (2) and (3)
  - Step 8: Construct the NAR set based on Equation (6)
- 

**3.3 | GRVFL for classification**

As we obtain the NAR, we propose a GRVFL as the classifier for COVID-19 diagnosis. The structure of GRVFL is similar to conventional random vector functional-link (RVFL),<sup>36</sup> which belongs to a kind of randomized neural network. However, the inputs to the GRVFL are the NARs in the graph instead of isolated ILRs. The GRVFL contains three layers. The  $\mathbf{R}_{NA}(i) = [r_1, r_2, r_3, \dots, r_D]^T$  denotes the input to the GRVFL, which is mapped to an enhanced space by random weight  $\mathbf{w}_i$  and bias  $b_i$ . Then, the enhanced representations in the hidden layer are concatenated with the original NARs to form the final integrated representations. Finally, these representations are mapped to output labels by the output weight  $\beta_i$ . The architecture of GRVFL is simple, but the classification performance is promising. As the COVID-19 diagnosis in this study is a binary classification problem and the NARs are obtained from chest CT images, we do not require deep networks for classification. The major advantage of using GRVFL is that the training is time-efficient compared with traditional networks, such as the



**FIGURE 2** Generation of neighboring aware representation. CNN, convolutional neural network; CT, computed tomography [Color figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

backpropagation neural network (BPNN). Because the training algorithm for BPNN is based on gradient descent methods, which need iterations to converge, training can be time-consuming. And, BPNN cannot ensure the training stops at the optimal solution, as gradient descent methods are greedy methods. However, GRVFL is trained differently. The input weight  $w_i$  and bias  $b_i$  are assigned with random values, and the output weight  $\beta_i$  is computed by Moore–Penrose pseudoinverse. There is no solution iteration, which contributes to the fast training. The norm of the weight after training is likely to get smaller values so that the GRVFL can produce higher generalization performance.<sup>37</sup> The training steps of GRVFL are presented in Proposed Algorithm 4.

#### Proposed Algorithm 4 GRVFL training algorithm

---

Load the NARs and the labels of the training set

---

Construct the GRVFL architecture

Compute the output of the hidden layer by Equation (8)

Obtain the concatenated representations by Equation (9)

Generate the output weight by Equation (10)

Output the trained GRVFL classifier

---

Suppose the training set is denoted as

$$\mathbf{S} = (\mathbf{R}_{\text{NA}}, T), \quad (7)$$

where  $T$  stands for the ground-truth labels. The output of the hidden nodes can be computed as

$$\mathbf{H} = \sum_{i=1}^{\hat{N}} f(w_i r_j + b_i), \quad j = 1, \dots, N, \quad (8)$$

where  $\hat{N}$  denotes the number of nodes in the hidden layer and  $f$  is the sigmoid activation function. Then, the representations can be obtained by concatenating the  $\mathbf{H}$  and input NARs

$$\mathbf{R} = \text{concatenate}(\mathbf{R}_{\text{NA}}, \mathbf{H}). \quad (9)$$

To suffice that  $\mathbf{R}\beta = T$ , we want the output of the GRVFL equals to the ground-truth labels. Therefore, the output weight can be obtained by pseudoinverse of  $\mathbf{R}$

$$\beta = \mathbf{R}^\dagger T. \quad (10)$$

The only predefined hyperparameter in GRVFL is the number of hidden nodes. With less manual intervention, the training of GRVFL can be easily implemented. For testing, the graph and NARs of the testing set should be computed, and fed into the GRVFL.

## 4 | EXPERIMENT DESIGN

### 4.1 | Data set configuration

We evaluated the proposed NAGNN on our private data set as well as a public data set. The public data set called SARS-COV-2 Ct-Scan Data set<sup>38</sup> is available on Kaggle (<https://www.kaggle.com/plameneduardo/sarscov2-ctscan-dataset>). There are 1252 COVID-19 CT images and 1230 non-COVID-19 CT scans in the SARS-COV-2 Ct-Scan Data set. The resolution of the CT images varies around  $300 \times 300 \times 3$ . More detailed information can be found in the literature.<sup>39</sup>

Our private chest CT images were obtained by a Philips spiral CT machine, and the detailed configurations for image acquisition were presented in Table 2. We obtained a COVID-19 data set of two categories: COVID-19 and normal control, and there are 420 images in the size of  $1024 \times 1024 \times 3$  for both classes.

To get the ground-truth labels of the chest CT images in the private data set, multiple manual labeling was conducted by two junior doctors and one senior doctor, denoted as  $(J_1, J_2, S)$ . The ground-truth target labels can be generated by the following expression:

$$\hat{T}(\text{CCT}) = \begin{cases} T(J_1) & \text{if } T(J_1) = T(J_2), \\ M(J_1, J_2, S) & \text{if } T(J_1) \neq T(J_2), \end{cases} \quad (11)$$

where  $T$  stands for the labels from each doctor, CCT denotes the chest CT image, and  $M$  represents the majority voting mechanism to deal with the situations that the two junior doctors came up with contradictory labeling results.

### 4.2 | Image preprocessing

The raw chest CT images need to be preprocessed before being fed into networks for training and testing because these raw images contain redundant information, such as background, and the image quality can be enhanced to get better classification performance. Additionally, the image size was  $1024 \times 1024 \times 3$ , which is larger for training state-of-the-art CNN models, as most input size of these models is  $227 \times 227 \times 3$ . We denote the raw image data set as

$$C_1 = \{c_1(1), c_1(2), c_1(3), \dots, c_1(i), \dots, c_1(|C_1|)\}, \quad (12)$$

TABLE 2 Image acquisition configurations

Parameter	Value
KV	120
MAS	240
Layer thickness	3 mm
Layer spacing	3 mm
Screw pitch	1.5
Lung window size	[1500, -500]
Mediastinum window size	[350, 60]

where  $|C_1| = 840$  as we got 420 for both classes. Then, we converted the data into grayscale images by

$$C_2 = \text{rgb2gray}(C_2) = \{c_2(1), \dots, c_2(i), \dots, c_2(|C_2|)\}. \quad (13)$$

Afterward, we enhanced the contrast of the images by the histogram normalization algorithm. The maximum and minimum grayscale value of  $c_2(i)$  can be found by

$$\begin{cases} p_{\min}(i) = \min_{1 \leq m, n \leq W} c_2(i | m, n), \\ p_{\max}(i) = \max_{1 \leq m, n \leq W} c_2(i | m, n), \end{cases} \quad (14)$$

where  $W = 1024$ . And the normalized image can be generated by

$$c_3(i | m, n) = \frac{c_2(i | m, n) - p_{\min}(i)}{p_{\max}(i) - p_{\min}(i)}, \quad 1 \leq m, n \leq W. \quad (15)$$

The normalized data set is denoted as

$$C_3 = \{c_3(1), c_3(2), c_3(3), \dots, c_3(i), \dots, c_3(|C_3|)\}. \quad (16)$$

To remove the background along with the text information near the boundaries and corners, the images are cropped as

$$C_4 = \text{crop}(C_3) = \{c_4(1), \dots, c_4(i), \dots, c_4(|C_4|)\}. \quad (17)$$

Finally, these images were resized to  $227 \times 227 \times 3$  to fit in the state-of-the-art CNN models. The raw image size was  $1024 \times 1024 \times 3$ , which was too large for CNN training.

Four samples from both private and public data sets are presented in Figure 3.

### 4.3 | Evaluation metrics

The proposed method was evaluated by fivefold cross-validation. Compared with 10-fold cross-validation, fewer samples serve for training, and more samples are tested in fivefold cross-validation. Meanwhile, 10-fold cross-validation requires approximately twice the time as

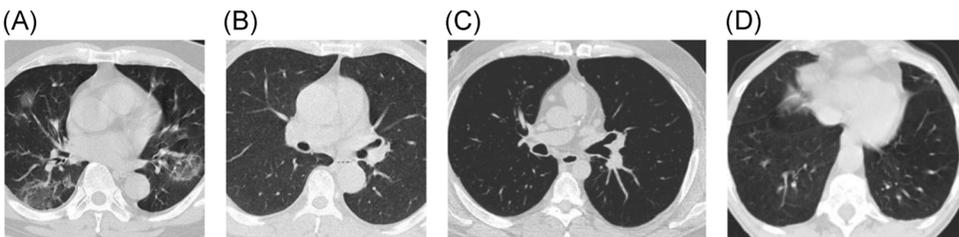


FIGURE 3 Chest computed tomography images in our final data set

that of fivefold cross-validation. We used the following common measurements, including accuracy, sensitivity, specificity, precision, and F1-score. They can be calculated by the four factors in the confusion matrix of binary classification: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). The formulae are presented below: Accuracy =  $(TP + TN) / (TP + TN + FP + FN)$ , Sensitivity =  $TP / (TP + FN)$ , Specificity =  $TN / (TN + FP)$ , Precision =  $TP / (TP + FP)$ , and F1-score =  $2 \times (\text{Precision} \times \text{Sensitivity}) / (\text{Precision} + \text{Sensitivity})$ . It can be revealed that the higher TP and TN are, the better the classification performance is. For the expression simplicity, we abbreviated these measurements as acc, sen, spe, pre, and F1 in the following sections.

#### 4.4 | Hyperparameter settings

The proposed COVID-19 detection approach was implemented and evaluated on a laptop with i7 7700HQ CPU, 16 GB RAM, and GTX 1060 GPU. The platform is MATLAB R2020a using deep learning toolbox and pretrained CNN models, including AlexNet,<sup>40</sup> ResNet-18,<sup>11</sup> ResNet-50,<sup>11</sup> DenseNet-201,<sup>12</sup> and MobileNetV2<sup>41</sup> for ILR extraction.

The hyperparameter settings are illustrated in Table 3. For transfer learning, the batch size was 20, max epochs were defined as 1 and the initial learning rate was  $1e - 4$ . The batch size was set considering the sizes of data sets and the computational capability of our GTX 1060 GPU. We set the max epochs as only 2 to prevent overfitting because the CNN models were pretrained, and COVID-19 data sets are small.  $1e - 4$  was the most often used setting for the learning rate. Small learning rates require more training time to converge, and larger learning rates may not converge in the training. The number of neighbors  $k$  in the graph generation was 3, determined by our experiments. In GRVFL training, the number of hidden nodes  $\hat{N}$  was set to be 400, because the dimension of input NARs was 256. Mapping the NARs into higher dimension spaces can hopefully reduce the classification complexity.

## 5 | RESULTS AND DISCUSSION

In this section, we first presented the experimental results on the private data set. Then, the classification performance of our NAGNN on the public SARS-COV-2 Ct-Scan Data set was discussed.

TABLE 3 Hyperparameter settings

Method	Hyperparameter	Value
ILR by transfer learning	Batch size	20
	Max epochs	2
	Learning rate	$1e - 4$
NAR by the graph of $k$ -NN	$k$	3
GRVFL training	$\hat{N}$	400

Abbreviations: GRVFL, graph random vector functional link; ILR, image-level representation;  $k$ -NN,  $k$ -nearest neighbor; NAR, neighboring aware representation.

## 5.1 | Classification performance of the NAGNN on the private data set

The proposed NAGNN was evaluated on the private data set using fivefold cross-validation. The running time for the fivefold cross-validation was 412.88 s. The classification results of the NAGNN were illustrated in Table 4. It can be observed that the proposed NAGNN achieved the perfect 100% sensitivity on fourfolds, which was excellent. Because it is universally acknowledged that sensitivity is more important in clinical diagnosis. If a potential COVID-19 patient was misdiagnosed as normal, it can cause more infected cases, which can be a disaster for the control of the virus. Meanwhile, the F1-score of the NAGNN was 99.04%, indicating the outstanding classification performance of the proposed model. In all, our NAGNN is accurate in the detection of COVID-19.

## 5.2 | Results of ILR generation on the private data set

To get the best ILRs, we conducted an experiment to transfer several state-of-the-art CNN models on our COVID-19 data set, including AlexNet,<sup>40</sup> ResNet-18,<sup>11</sup> ResNet-50,<sup>11</sup> DenseNet-201,<sup>12</sup> and MobileNetV2,<sup>41</sup> which are all widely used models. The testing results are shown in Table 5 and Figure 4 based on fivefold cross-validation.

It can be found that all the transferred models achieved good performance except AlexNet, of which the accuracy was only 50.00%. In the experiment of AlexNet, the training cannot effectively proceed, and as a result, the transferred AlexNets either classified all the testing images as COVID-19 or recognize all the testing images as normal. Therefore, some measurements were incalculable (NaN). The poor performance of AlexNet may result from these two reasons: the structure of AlexNet is simple, and there are no shortcut connections; batch normalization is not included. However, both shortcut connections and batch normalization are crucial for convergence during training. On the other hand, all the other four backbone models produced promising results, and ResNet-50 achieved the best sensitivity, but the specificity was only 94.68%. Both ResNet-18 and DenseNet-201 produced high accuracy and achieved a good balance among these five measurements, but the size of ResNet-18 is much simpler than DenseNet-201, which is 44 MB versus 77 MB, and the number of parameters in ResNet-18 is only half of that in DenseNet-201. Therefore, we chose ResNet-18 as our backbone model for ILR generation.

**TABLE 4** Classification performance of the NAGNN on the private data set (unit, %; F, fold; A, average)

	Acc	Sen	Spe	Pre	F1
F1	99.40	100.00	98.82	98.81	99.40
F2	99.40	100.00	98.82	98.81	99.40
F3	98.21	100.00	96.55	96.43	98.18
F4	99.40	100.00	98.82	98.81	99.40
F5	98.81	98.81	98.81	98.81	98.81
A	99.05	99.76	98.37	98.33	99.04

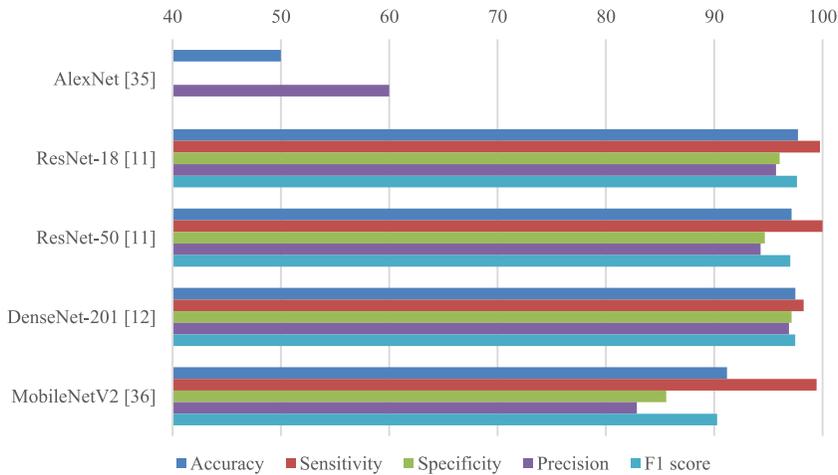
Abbreviations: Acc, accuracy; F1, F1-score; NAGNN, neighboring aware graph neural network; Pre, precision; Sen, sensitivity; Spe, specificity.

**TABLE 5** Transfer learning of CNN models on the private data set (unit, %; F, fold; A, average)

Model	Acc	Sen	Spe	Pre	F1
<i>AlexNet</i> <sup>40</sup>					
F1	50.00	50.00	NaN	100.00	66.67
F2	50.00	50.00	NaN	100.00	66.67
F3	50.00	NaN	50.00	0	NaN
F4	50.00	NaN	50.00	0	NaN
F5	50.00	50.00	NaN	100.00	66.67
A	50.00	NaN	NaN	60.00	NaN
<i>ResNet-18</i> <sup>11</sup>					
F1	97.62	100.00	95.45	95.24	97.56
F2	98.21	100.00	96.55	96.43	98.18
F3	94.05	100.00	89.36	88.10	93.67
F4	99.40	100.00	98.82	98.81	99.40
F5	99.40	98.82	100.00	100.00	99.41
A	<b>97.74</b>	99.76	96.04	95.71	<b>97.64</b>
<i>ResNet-50</i> <sup>11</sup>					
F1	98.21	100.00	96.55	96.43	98.18
F2	98.81	100.00	97.67	97.62	98.80
F3	97.62	100.00	95.45	95.24	97.56
F4	94.05	100.00	89.36	88.10	93.67
F5	97.02	100.00	94.38	94.05	96.93
A	97.14	<b>100.00</b>	94.68	94.29	97.03
<i>DenseNet-201</i> <sup>12</sup>					
F1	95.24	91.30	100.00	100.00	95.45
F2	100.00	100.00	100.00	100.00	100.00
F3	97.02	100.00	94.38	94.05	96.93
F4	95.24	100.00	91.30	90.48	95.00
F5	100.00	100.00	100.00	100.00	100.00
A	97.50	98.26	<b>97.14</b>	<b>96.90</b>	97.48
<i>MobileNetV2</i> <sup>41</sup>					
F1	96.43	98.75	94.32	94.05	96.34
F2	87.50	100.00	80.00	75.00	85.71
F3	91.07	100.00	84.85	82.14	90.20
F4	89.88	98.55	83.84	80.95	88.89
F5	91.07	100.00	84.85	82.14	90.20
A	91.19	99.46	85.57	82.86	90.27

Note: Bold values mean the best average values.

Abbreviations: Acc, accuracy; CNN, convolutional neural network; F1, F1-score; Pre, precision; Sen, sensitivity; Spe, specificity.



**FIGURE 4** Comparison of CNN models based on fivefold cross-validation (unit, %). CNN, convolutional neural network [Color figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

### 5.3 | Effects of graph generation on the private data set

The graph of the ILRs in this study was constructed by the  $k$ -NN algorithm, which was closely related to the NARs. And the structure of the graph is dependent on the number of neighbors,  $k$ , in  $k$ -NN. To uncover the effects of this parameter, we proposed to get the optimal value of  $k$  by grid search. The performance of the proposed COVID-19 detection method, abbreviated as NAGNN, based on fivefold cross-validation with different values of  $k$  is provided in Table 6 and Figure 5. It can be observed that the accuracy of the proposed NAGNN with  $k$  ranging from 2 to 6 was around 99.00%, and the sensitivity fluctuated between 99.00% and 100.00%. The effect of different  $k$  values was more obvious in terms of precision, ranging from 96.90% to 98.33%. In conclusion, the NAGNN produced the best classification results with  $k = 3$ . Hence, the optimal value for the number of neighbors was set to be 3.

### 5.4 | Effects of GRVFL on the private data set

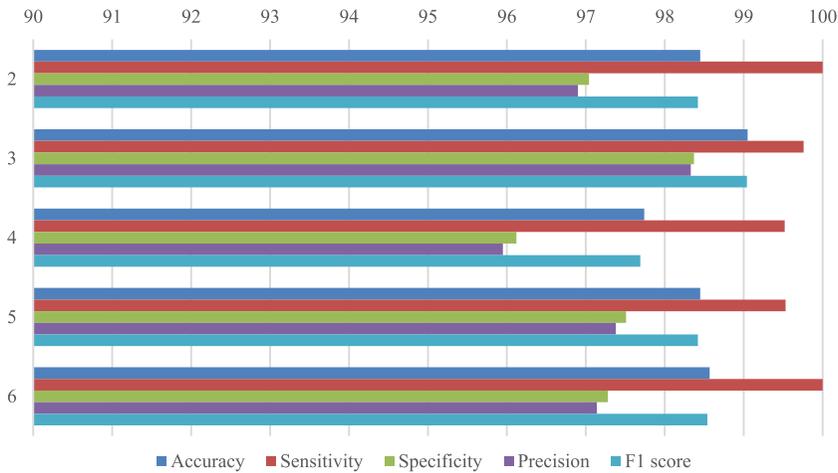
We compared the classification performance of the proposed GRVFL with conventional RVFL based on fivefold cross-validation. The structure of RVFL and GRVFL was the same which both contained 400 nodes in the hidden layer. The difference between the two is that the input of GRVFL is NAR while that of RVFL is ILR. The comparison results are reported in Table 7 and Figure 6. The results suggested that the average performance of GRVFL was better than RVFL for all five metrics, which means that the classification performance can be improved by the spatial relationship among the representations in the latent space. Moreover, the fluctuation range was reduced, that is, in terms of accuracy, the fluctuation range was just a little over 1% while that of RVFL was nearly 3%. Hence, GRVFL was more robust than RVFL. The improvement of the classification performance is contributed by the neighboring information in the latent feature space. The NARs were obtained by the fusion of the ILRs and their neighboring relationships so that the GRVFL can outperform the conventional RVFL.

**TABLE 6** Performance of NAGNN with different  $k$  values on the private data set (unit, %; F, fold; A, average)

Value of $k$		Acc	Sen	Spe	Pre	F1
2	F1	97.62	100.00	95.45	95.24	97.56
	F2	98.81	100.00	97.67	97.62	98.80
	F3	97.02	100.00	94.05	94.05	96.93
	F4	100.00	100.00	100.00	100.00	100.00
	F5	98.81	100.00	97.67	97.62	98.80
	A	98.45	<b>100.00</b>	97.04	96.90	98.42
3	F1	99.40	100.00	98.82	98.81	99.40
	F2	99.40	100.00	98.82	98.81	99.40
	F3	98.21	100.00	96.55	96.43	98.18
	F4	99.40	100.00	98.82	98.81	99.40
	F5	98.81	98.81	98.81	98.81	98.81
	A	<b>99.05</b>	99.76	<b>98.37</b>	<b>98.33</b>	<b>99.04</b>
4	F1	98.21	100.00	96.55	96.43	98.18
	F2	98.81	100.00	97.67	97.62	98.80
	F3	96.43	100.00	93.33	92.86	96.30
	F4	98.21	100.00	96.55	96.43	98.18
	F5	97.02	97.59	96.47	96.43	97.01
	A	97.74	99.52	96.12	95.95	97.69
5	F1	99.40	98.82	100.00	100.00	99.41
	F2	97.02	100.00	94.38	94.05	96.93
	F3	97.02	100.00	94.38	94.05	96.93
	F4	98.81	98.81	98.81	98.81	98.81
	F5	100.00	100.00	100.00	100.00	100.00
	A	98.45	99.53	97.51	97.38	98.42
6	F1	100.00	100.00	100.00	100.00	100.00
	F2	97.62	100.00	95.45	95.24	97.56
	F3	97.02	100.00	94.38	94.05	96.93
	F4	98.21	100.00	96.55	96.43	98.18
	F5	100.00	100.00	100.00	100.00	100.00
	A	98.57	<b>100.00</b>	97.28	97.14	98.54

Note: Bold values mean the best average values.

Abbreviations: Acc, accuracy; F1, F1-score; NAGNN, neighboring aware graph neural network; Pre, precision; Sen, sensitivity; Spe, specificity.



**FIGURE 5** Performance of NAGNN with different  $k$  values (unit, %). NAGNN, neighboring aware graph neural network [Color figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

**TABLE 7** Comparison of RVFL with GRVFL on the private data set (unit, %; F, fold; A, average)

Model	Acc	Sen	Spe	Pre	F1
<i>RVFL</i>					
F1	97.62	97.62	97.62	97.62	97.62
F2	98.81	97.67	100.00	100.00	98.82
F3	98.81	100.00	97.67	97.62	98.80
F4	95.83	95.29	96.39	96.43	95.86
F5	97.62	98.78	96.51	96.43	97.59
A	97.74	97.87	97.64	97.62	97.74
<i>GRVFL</i>					
F1	99.40	100.00	98.82	98.81	99.40
F2	99.40	100.00	98.82	98.81	99.40
F3	98.21	100.00	96.55	96.43	98.18
F4	99.40	100.00	98.82	98.81	99.40
F5	98.81	98.81	98.81	98.81	98.81
A	<b>99.05</b>	<b>99.76</b>	<b>98.37</b>	<b>98.33</b>	<b>99.04</b>

Note: Bold values mean the best average values.

Abbreviations: Acc, accuracy; F1, F1-score; GRVFL, graph random vector functional link; Pre, precision; RVFL, random vector functional link; Sen, sensitivity; Spe, specificity.

## 5.5 | Classification results of the NAGNN on the public data set

To further evaluate the COVID-19 detection performance of the proposed NAGNN, we used the public SARS-COV-2 Ct-Scan Data set in experiments. All the hyperparameters were the same as

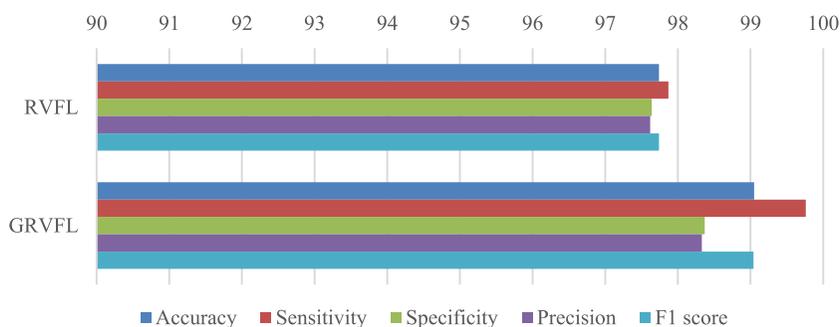


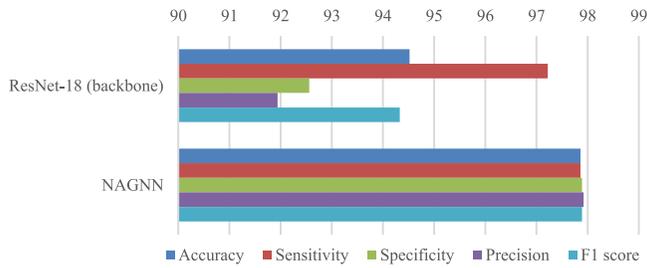
FIGURE 6 Comparison of RVFL with GRVFL (unit, %). GRVFL, graph random vector functional link; RVFL, random vector functional link [Color figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

TABLE 8 Results of the NAGNN on the public data set (unit, %; F, fold; A, average)

Model	Acc	Sen	Spe	Pre	F1
<i>ResNet-18 (backbone)</i>					
F1	96.97	97.96	96.00	96.00	96.97
F2	90.34	100.00	83.67	80.88	89.43
F3	93.75	95.44	92.16	92.00	93.69
F4	95.97	99.15	93.13	92.80	95.87
F5	95.57	93.54	97.86	98.01	95.72
A	94.52	97.22	92.56	91.94	94.33
<i>NAGNN</i>					
F1	99.19	99.20	99.18	99.20	99.20
F2	97.79	97.62	97.96	98.01	97.81
F3	97.58	98.77	96.43	96.40	97.57
F4	96.77	96.06	97.52	97.60	96.83
F5	97.99	97.63	98.36	98.41	98.02
A	97.86	97.86	97.89	97.92	97.89

Abbreviations: Acc, accuracy; F1, F1-score; NAGNN, neighboring aware graph neural network; Pre, precision; Sen, sensitivity; Spe, specificity.

the experiments on our private data set, and the results were obtained based on fivefold cross-validation. The entire running time for the fivefold cross-validation was 1209.37 s. The statistics were listed in Table 8 and Figure 7. It can be observed that all the metrics gained an approximately 3% increase from the backbone ResNet-18 to the proposed NAGNN. All the five metrics of the NAGNN were over 97% and were close. On the other hand, the five metrics of ResNet-18 vary in a range of over 5%. The neighboring information in the latent feature space and the GRVFL in the NAGNN may be the main reasons for the classification improvement. Meanwhile, the NAGNN achieved better robustness. The results from the public SARS-COV-2 Ct-Scan Data set were consistent with those from the private data set, which suggested that the proposed NAGNN was effective in the diagnosis of COVID-19.



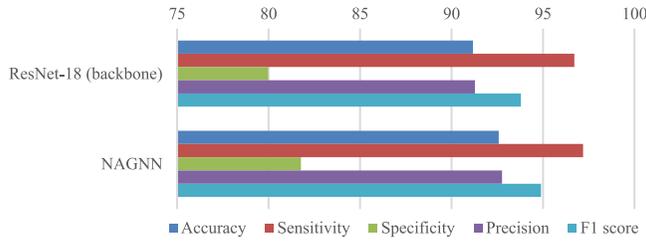
**FIGURE 7** Results on the public data set (unit, %). NAGNN, neighboring aware graph neural network [Color figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

**TABLE 9** Results of the NAGNN on the public variant data set for patient-level evaluation (unit, %; F, fold; A, average)

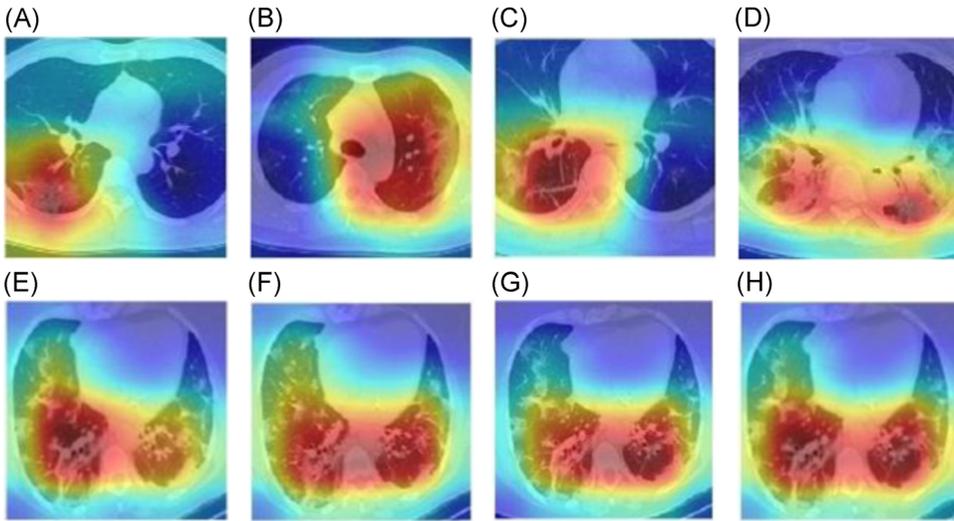
Model	Acc	Sen	Spe	Pre	F1
<i>ResNet-18 (backbone)</i>					
F1	87.79	97.81	69.50	85.41	91.19
F2	91.48	92.52	88.44	95.88	94.17
F3	87.24	98.25	67.27	84.48	90.85
F4	94.32	98.20	85.63	93.86	95.98
F5	95.02	96.77	89.05	96.77	96.77
A	91.17	96.71	79.98	91.28	93.79
<i>NAGNN</i>					
F1	89.73	94.78	77.30	91.15	92.93
F2	94.26	95.24	91.61	96.85	96.04
F3	90.63	97.21	75.66	90.09	93.51
F4	96.27	98.98	89.94	95.82	97.38
F5	92.03	99.76	73.42	89.89	94.57
A	92.58	97.19	81.77	92.76	94.89

Abbreviations: Acc, accuracy; F1, F1-score; NAGNN, neighboring aware graph neural network; Pre, precision; Sen, sensitivity; Spe, specificity.

The above experiments were based on slice-level classification, so the data leakage issue was inevitable as the slices from the same patients were in both the training and testing set. To better evaluate the patient-level classification performance of the proposed NAGNN, we employed a variant of the public SARS-COV-2 Ct-Scan Data set<sup>39</sup> on the Kaggle website (<https://www.kaggle.com/plameneduardo/a-covid-multiclass-dataset-of-ct-scans>). In this variant public data set, obtained 758 healthy CT images from 50 patients and 2168 COVID-19 CT scans from 80 patients. The CT slices of the same patients were grouped in folders. Therefore, we experimented with this variant public data set for detecting COVID-19 from healthy controls. Fivefold cross-validation was also utilized, so there were 10 healthy patients and 16 patients in each fold. The overall running time of the fivefold cross-validation on the NAGNN was 803.54 s. The results were shown in Table 9 and Figure 8. The proposed NAGNN gained improvement



**FIGURE 8** Results on the public variant data set for patient-level evaluation (unit, %). NAGNN, neighboring aware graph neural network [Color figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]



**FIGURE 9** Grad-CAMs of COVID-19 samples. Grad-CAM, gradient-weighted class activation mapping [Color figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

for all the five evaluation metrics compared with the backbone model. We found that the differences between folds were big and there was an obvious decrease in performance compared with the slice-level results in Figure 7. We held the view that the potential reasons may include the high interpatient variance and the class-imbalance distribution in the variant data set. Although the variant data set had more slices than the original public data set, the number of COVID-19 slices was over twice more than that of healthy slices, which was imbalanced for training the models and resulted in the high sensitivity but much worse specificity. Moreover, all the slices were generated from merely 80 COVID-19 patients and 50 healthy ones, so the diversity of the variant data set can be poor in terms of patient-level classification. However, the NAGNN still produced an accuracy of 92.58%, a sensitivity of 97.19%, and an F1-score of 94.89%.

### 5.6 | Interpretation of the NAGNN

Grad-CAM<sup>42</sup> is an important tool to interpret the predictions from deep models. Grad-CAM can visualize the heat map of the attention of the CNN models when predicting labels of input images. Eight Grad-CAMs of COVID-19 samples from the proposed NAGNN are presented in

**TABLE 10** Comparison with state-of-the-art approaches (unit, %)

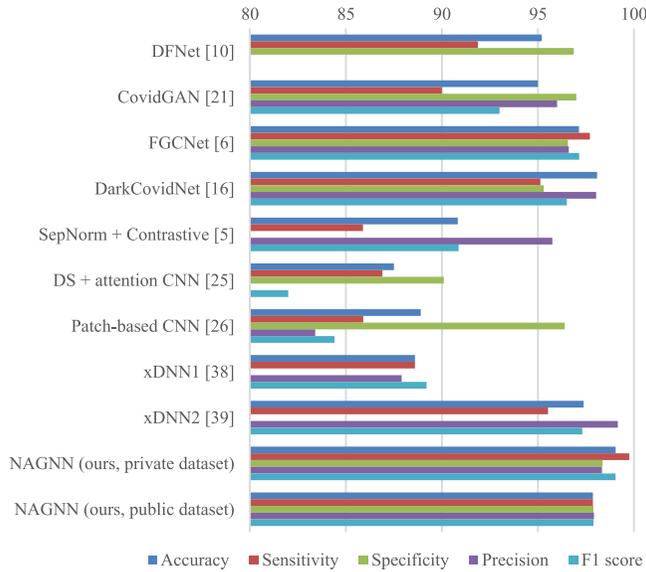
Method	Acc	Sen	Spe	Pre	F1	Data set
DFNet <sup>10</sup>	95.20	91.87	96.87	–	–	Private/imbalanced
COVIDGAN <sup>21</sup>	95.00	90.00	97.00	96.00	93.00	Private/imbalanced
FGCNet <sup>6</sup>	97.14	97.71	96.56	96.61	97.15	Private/balanced
DarkCOVIDNet <sup>16</sup>	98.08	95.13	95.3	98.03	96.51	Public/balanced
Dual-Track Learning <sup>27</sup>	–	86.00	–	89.60	87.80	Private/imbalanced
SepNorm + Contrastive <sup>5</sup>	90.83	85.89	–	95.75	90.87	Public/balanced
DS + attention CNN <sup>25</sup>	87.5	86.9	90.1	–	82.0	Private/imbalanced
Patch-based CNN <sup>26</sup>	88.9	85.9	96.4	83.4	84.4	Public/imbalanced
xDNN1 <sup>38</sup>	88.6	88.6	–	87.9	89.2	Public/balanced
xDNN2 <sup>39</sup>	97.38	95.53	–	99.16	97.31	Public/balanced
NAGNN (ours)	99.05	99.76	98.37	98.33	99.04	Private/balanced
NAGNN (ours)	97.86	97.86	97.89	97.92	97.89	Public/balanced

Abbreviations: Acc, accuracy; CNN, convolutional neural network; F1, F1-score; NAGNN, neighboring aware graph neural network; Pre, precision; Sen, sensitivity; Spe, specificity.

Figure 9, where four of them are from the private data set and the rest four are from the public data set. The colormap is jet. The regions in orange and red colors are where the attention of the NAGNN, while the regions in blue color are ignored by the model. It can be discovered that the NAGNN is capable of getting suspicious regions from chest CT images.

## 5.7 | Comparison with state-of-the-art methods

We compared the proposed NAGNN with other state-of-the-art CAD systems, including DFNet,<sup>10</sup> COVIDGAN,<sup>21</sup> FGCNet,<sup>6</sup> DarkCOVIDNet,<sup>16</sup> Dual-Track Learning,<sup>27</sup> SepNorm + Contrastive,<sup>5</sup> DS + attention CNN,<sup>25</sup> Patch-based CNN,<sup>26</sup> xDNN1,<sup>38</sup> and xDNN2.<sup>39</sup> The detailed information is provided in Table 10 and Figure 10. The ‘Data set’ column indicated whether the data set in the experiments was public or private and whether it was class-balanced or class-imbalanced. It can be discovered that most of the methods achieved over 95.00% accuracy but our NAGNN yielded the best accuracy at 99.05% on the private data set. Sensitivity is a significant indicator in clinical diagnosis and three state-of-the-art algorithms produced over 95.00% sensitivity. Our NAGNN achieved outstanding generalization ability with an F1-score of 99.04% on the private data set. The public SARS-COV-2 Ct-Scan Data set in SepNorm + Contrastive, xDNN1, and xDNN2 was also employed to evaluate our NAGNN. Results showed that our NAGNN outperformed the three in terms of accuracy, sensitivity, specificity, and F1-score. There are three reasons behind the good classification performance. First, the transfer learning from pretrained CNN models can effectively extract ILRs from chest CT images and eliminate excessive information. Second, the proposed NAR based on graph construction can leverage the distribution information between the ILRs in the latent feature space, so the obtained NARs can be more robust. Last but not least, the GRVFL classifier is easy to optimize because of its simple structure compared with deep networks. The running time for training and testing the proposed NAGNN based on fivefold



**FIGURE 10** Comparison with state-of-the-art approaches (unit, %). CNN, convolutional neural network; DS, dual sampling; NAGNN, neighboring aware graph neural network [Color figure can be viewed at [wileyonlinelibrary.com](http://wileyonlinelibrary.com)]

cross-validation was only 412.88 s on private data set and 1209.37 s on public data set, which was intensively fast. In all, the proposed NAGNN offers a good CAD method for COVID-19 detection.

## 6 | CONCLUSION

In this paper, we presented a novel COVID-19 classification method named NAGNN, based on chest CT images. We proposed to first employ transfer learning to obtain ILRs with pretrained state-of-the-art CNN models. Then, to leverage the distribution knowledge of these ILRs, we proposed to generate a graph of the ILRs by *k*-NN algorithm, and the novel NARs can be computed based on the graph. Finally, to classify the NARs, a GRVFL was trained by pseudoinverse. The proposed NAGNN produced an average accuracy of 99.29% on our private data set and an average accuracy of 97.86% on the public SARS-COV-2 Ct-Scan Data set based on fivefold cross-validation, which outperformed 10 state-of-the-art COVID-19 detection methods. The results of Grad-CAM revealed that NAGNN can find the suspicious regions in the chest images automatically and make accurate predictions based on the representations. The good generalization ability suggested that our NAGNN can be used as a verification tool in real-world clinical COVID-19 diagnosis.

However, there are some disadvantages of the proposed method. First of all, the patient-level classification results were worse than slice-level classification performance. We shall develop more advanced models to deal with the high interpatient variance in CT scans. Second, it remains a problem how the CNN model locates the suspicious regions though Grad-CAM can reveal that the predictions from the model are made by these suspicious regions in chest CT images. Another shortcoming is the size of the data set. We only collected a small data set. In the future, we shall continue to collect more images and

include more categories of samples. Finally, our model was only trained and tested by chest CT images. We shall test the model with images of other modalities, such as X-ray images and ultrasound images in our future work. Also, in clinical diagnosis, segmentation of the CT images to get potential focuses is more desired, which is another future research direction.

## ACKNOWLEDGMENTS

Siyuan Lu holds a CSC scholarship with the University of Leicester. This paper is partially supported by the Royal Society International Exchanges Cost Share Award, UK (RP20G0230), Medical Research Council Confidence in Concept Award, UK (MC\_PC\_17171), Hope Foundation for Cancer Research, UK (RM60G0680), Sino-UK Industrial Fund, UK (RP20G0289), Global Challenges Research Fund (GCRF), UK (P202PF11), and British Heart Foundation Accelerator Award, UK (AA/18/3/34220).

## CONFLICT OF INTERESTS

The authors declare that there are no conflict of interests.

## AUTHOR CONTRIBUTIONS

*Conceptualization, Methodology, Software, Formal analysis, Writing—Original Draft, Writing—Review and Editing, Visualization:* Siyuan Lu. *Methodology, Software, Data Curation, Writing—Original Draft, Writing—Review and Editing:* Ziquan Zhu. *Methodology, Formal analysis, Writing—Original Draft, Writing—Review and Editing:* Juan Manuel Gorriz. *Validation, Data Curation, Resources, Writing—Review and Editing, Supervision, Project administration, Funding acquisition:* Shui-Hua Wang. *Conceptualization, Validation, Formal analysis, Resources, Writing—Review and Editing, Supervision, Project administration, Funding acquisition:* Yu-Dong Zhang.

## ORCID

Siyuan Lu  <https://orcid.org/0000-0001-6720-1323>

Ziquan Zhu  <https://orcid.org/0000-0001-8792-9354>

Juan Manuel Gorriz  <https://orcid.org/0000-0001-7069-1714>

Shui-Hua Wang  <https://orcid.org/0000-0003-4713-2791>

Yu-Dong Zhang  <https://orcid.org/0000-0002-4870-1493>

## REFERENCES

1. Chowdhury MEH, Rahman T, Khandakar A, et al. Can AI help in screening viral and COVID-19 pneumonia? *IEEE Access*. 2020;8:132665-132676.
2. Wang X, Deng X, Fu Q, et al. A weakly-supervised framework for COVID-19 classification and lesion localization from chest CT. *IEEE Trans Med Imaging*. 2020;39(8):2615-2625.
3. Rajaraman S, Siegelman J, Alderson PO, Folio LS, Folio LR, Antani SK. Iteratively pruned deep learning ensembles for COVID-19 detection in chest X-rays. *IEEE Access*. 2020;8:115041-115050.
4. Roy S, Menapace W, Oei S, et al. Deep learning for classification and localization of COVID-19 markers in point-of-care lung ultrasound. *IEEE Trans Med Imaging*. 2020;39(8):2676-2687.
5. Wang Z, Liu Q, Dou Q. Contrastive cross-site learning with redesigned net for COVID-19 CT classification. *IEEE J Biomed Health Inf*. 2020;24(10):2806-2813.
6. Wang SH, Govindaraj VV, Gorriz JM, Zhang X, Zhang YD. COVID-19 classification by FGCNet with deep feature fusion from graph convolutional network and convolutional neural network. *Inf Fusion*. 2021;67:208-229.

7. Horry MJ, Chakraborty S, Paul M, et al. COVID-19 detection through transfer learning using multimodal imaging data. *IEEE Access*. 2020;8:149808-149824.
8. Huang C, Xie, Y, Lan Y, et al. A new framework for the integrative analytics of intravascular ultrasound and optical coherence tomography images. *IEEE Access*. 2018;6:36408-36419.
9. Górriz JM, Ramirez J, Suckling J. On the computation of distribution-free performance bounds: Application to small sample sizes in neuroimaging. *Pattern Recogn*. 2019;93:1-13.
10. Yu Z, Li X, Sun H, et al. Rapid identification of COVID-19 severity in CT scans through classification of deep features. *Biomed Eng Online*. 2020;19(1):63.
11. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE; 2016:770-778.
12. Huang G, Liu Z, Laurens V, Weinberger K. Densely connected convolutional networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; 2017:2261-2269.
13. Talo M, Baloglu UB, Yildirim Ö, Rajendra Acharya U. Application of deep transfer learning for automated brain abnormality classification using MR images. *Cognit Syst Res*. 2019;54:176-188.
14. Lu S, Lu Z, Zhang Y-D. Pathological brain detection based on AlexNet and transfer learning. *J Comput Sci*. 2019;30:41-47.
15. Ucar F, Korkmaz D. COVIDiagnosis-Net: Deep Bayes-SqueezeNet based diagnosis of the coronavirus disease 2019 (COVID-19) from X-ray images. *Med Hypotheses*. 2020;140:109761.
16. Ozturk T, Talo M, Yildirim EA, Baloglu UB, Yildirim O, Rajendra Acharya U. Automated detection of COVID-19 cases using deep neural networks with X-ray images. *Comput Biol Med*. 2020;121:103792.
17. Apostolopoulos ID, Aznaouridis SI, Tzani MA. Extracting possibly representative COVID-19 biomarkers from X-ray images with deep learning approach and image data related to pulmonary diseases. *J Med Biol Eng*. 2020;40:462-469.
18. Apostolopoulos ID, Mpesiana TA. COVID-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks. *Phys Eng Sci Med*. 2020;43(2):635-640.
19. Zhou L, Li Z, Zhou J, et al. A rapid, accurate and machine-agnostic segmentation and quantification method for CT-based COVID-19 diagnosis. *IEEE Trans Med Imaging*. 2020;39(8):2638-2652.
20. Yasar H, Ceylan M. A novel comparative study for detection of COVID-19 on CT lung images using texture analysis, machine learning, and deep learning methods. *Multimedia Tools Appl*. 2020;80:5423-5447.
21. Waheed A, Goyal M, Gupta D, Khanna A, Al-Turjman F, Pinheiro PR. COVIDGAN: data augmentation using auxiliary classifier GAN for improved COVID-19 detection. *IEEE Access*. 2020;8:91916-91923.
22. Togacar M, Ergen B, Comert Z. COVID-19 detection using deep learning models to exploit social mimic optimization and structured chest X-ray images using fuzzy color and stacking approaches. *Comput Biol Med*. 2020;121:103805.
23. Sun L, Mo Z, Yan F, et al. Adaptive feature selection guided deep forest for COVID-19 classification with chest CT. *IEEE J Biomed Health Inf*. 2020;24(10):2798-2805.
24. Sakib S, Tazrin T, Fouda MM, Fadlullah ZM, Guizani M. DL-CRC: deep learning-based chest radiograph classification for COVID-19 detection: a novel approach. *IEEE Access*. 2020;8:171575-171589.
25. Ouyang X, Huo J, Xia L, et al. Dual-sampling attention network for diagnosis of COVID-19 from community acquired pneumonia. *IEEE Trans Med Imaging*. 2020;39(8):2595-2605.
26. Oh Y, Park S, Ye JC. Deep learning COVID-19 features on CXR using limited training data sets. *IEEE Trans Med Imaging*. 2020;39(8):2688-2700.
27. Li Y, Wei D, Chen J, et al. Efficient and effective training of COVID-19 classification networks with self-supervised dual-track learning to rank. *IEEE J Biomed Health Inf*. 2020;24(10):2787-2797.
28. Ahonen T, Hadid A, Pietikainen M. Face description with local binary patterns: application to face recognition. *IEEE Trans Pattern Anal Mach Intell*. 2006;28(12):2037-2041.
29. Guo Z, Zhang L, Zhang D. A completed modeling of local binary pattern operator for texture classification. *IEEE Trans Image Process*. 2010;19(6):1657-1663.
30. Attallah O, Ragab DA, Sharkas M. MULTI-DEEP: a novel CAD system for coronavirus (COVID-19) diagnosis from CT images using multiple convolution neural networks. *PeerJ*. 2020;8:e10086.
31. Zhao L, Song Y, Zhang C, et al. T-GCN: a temporal graph convolutional network for traffic prediction. *IEEE Trans Intell Transp Syst*. 2020;21(9):3848-3858.

32. Yao L, Mao C, Luo Y. Graph convolutional networks for text classification. In: *The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)*. AAAI Press; 2019:7370-7377.
33. Jiang B, Wang L, Tang J, Luo B. Context-aware graph attention networks. 2019. arXiv:191001736.
34. Song J, Chang C, Sun F, Song X, Jiang P. NGAT4Rec: neighbor-aware graph attention network for recommendation. 2021. arXiv:201012256.
35. Sun J, Zhang Y, Guo W, et al. Neighbor interaction aware graph convolution networks for recommendation. In: *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. Virtual Event, ACM Press; 2020:1289-1298.
36. Pao YH, Park GH, Sobajic DJ. Learning and generalization characteristics of random vector functional-link net. *Neurocomputing*. 1994;6:163-180.
37. Bartlett PL. The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network. *IEEE Trans Inf Theory*. 1998;44(2):525-536.
38. Angelov P, Soares E. Towards explainable deep neural networks (xDNN). *Neural Net*. 2020;130:185-194.
39. Angelov P, Soares E. SARS-CoV-2 CT-scan dataset: a large dataset of real patients CT scans for SARS-CoV-2 identification. 2020. medRxiv.
40. Krizhevsky A, Sutskever I, Hinton G. ImageNet classification with deep convolutional neural networks. In: *International Conference on Neural Information Processing Systems*. Curran Associates Inc.; 2012: 1097-1105.
41. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L-C. MobileNetV2: inverted residuals and linear bottlenecks. In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE; 2018: 4510-4520.
42. Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D. Grad-CAM: visual explanations from deep networks via gradient-based localization. *Int J Comput Vis*. 2017;128:336-359.

**How to cite this article:** Lu S, Zhu Z, Gorriz JM, Wang S-H, Zhang Y-D. NAGNN: Classification of COVID-19 based on neighboring aware representation from deep graph neural network. *Int J Intell Syst*. 2022;37:1572-1598. <https://doi.org/10.1002/int.22686>