Diagnosis of covid-19 pneumonia based on graph convolutional network

Introduction

The first case of COVID-19 was described in China in December 2019, and then COVID-19 has spread all over the world rapidly. So far, it has infected over 31 million people and has resulted in over 0. 9 million deaths as of September 23, 2020. With the numerous cases needed to be tested, most countries and regions face a shortage of testing kits and medical resources. For this issue, a series of automatic diagnosis methods based on deep learning models are proposed to relieve the medical burden ([1]). Unlike the reverse-transcription polymerase chain reaction (RT-PCR) based on a patient's respiratory samples, automatic diagnosis methods based on deep learning models usually accomplish the diagnosis task by using chest radiography images ([2]). Various published research articles indicate that chest scans are useful in detecting COVID-19 ([3] , [4]). The lungs of the infected cases have visual marks like ground-glass opacity or hazy darkened spots, which help to differentiate infected cases from normal controls ([5]). With good sensitivity (SEN) and speed, chest computed tomography (CT) has been widely used in automatic diagnosis methods ([6] – [9]).

For the existing automatic diagnosis methods of COVID-19 based on medical images, structures based on a convolutional neural network (CNN) are widely used. For example, the DarkNet model ([10]) with 17 convolutional layers is used as a classifier for COVID-19 diagnosis based on X-ray images, and its accuracy (ACC) reaches 98. 08%. A transfer learning neural network ([11]) based on the inception network is proposed to fit in few-shot CT datasets, and it achieved a total ACC of 89. 5%. A deep three-dimensional (3D)-CNN (

12) is applied to detect COVID-19 from CT volumes reaching 90. 8% ACC. Additionally, ResNet50 (13), ResNet152 (14), LSTM (15), GAN (16 , 17), and some other structures are successively used for COVID-19 diagnosis. Limited by the insufficient training samples and the great number of parameters in deep learning structures, the ACC of the above methods based on 3D-CT images is not satisfied. Besides, in most of the existing automatic diagnosis methods, the differences between image standards from different equipment types and hospitals are ignored, which further deteriorates the final diagnosis performance. As COVID-19 widely spreads, it is of great significance to exploit a robust diagnosis method to adapt to different acquired equipment types all over the world.

A graph convolutional network (GCN), which integrates phenotypic information into a graph to establish interactions between individuals and populations, can achieve an excellent filtering effect by graph theory. Nevertheless, there are no related works to study its application for COVID-19 diagnosis. Therefore, we propose a novel COVID-19 graph in GCN, which considers the image differences between different equipment types and hospitals. Specifically, we first employ the popular 3D-CNN structure to extract image features from 3D-CT images. In this process, a transfer learning method based on predicting equipment type is used to initialize the parameters of 3D-CNN. After this process, every subject on the graph is represented by a feature vector. Besides, every subject gets an initial diagnosis score from 3D-CNN. Second, we design a COVID-19 graph in GCN to consider the differences between different equipment types, hospital information, and disease statuses of those training samples. We also

combine the extracted features and the scores from 3D-CNN to construct edge weights. Third, we input our COVID-19 graph into a GCN model for the final diagnosis.

Overall, we apply a 3D-CNN to extract image features from CT images and then design a COVID-19 graph in GCN to complete the diagnosis. The main contributions are described as follows:

(1) We propose a transfer learning method by predicting equipment type to initialize the parameters of the 3D-CNN structure for extracting features from CT images.

(2) We design a COVID-19 graph in GCN, which considers the information of equipment type, hospital, and disease status. We compute edge weights based on the correlation distance of extracted features and the score differences of subjects from the 3D-CNN structure.

(3) We analyze the feature differences between different equipment types. Experimental results show that our method has a good diagnosis performance.

The rest of the paper is organized as follows. The related works of 3D-CNN and GCN are introduced in section Related Works. In section Methodology, our methodology is presented. The results of our experiments and analysis of feature differences between different equipment types are given in section Experiments and Results. Finally, this paper is concluded.

Related Works

3D-CNN for Feature Extraction

As deep CNN can filter noise and reduce parameters, it is widely studied. The most popular CNN methods include LeNet-5, AlexNet, VGG-16, Inception-v1, ResNet-50, Inception-ResNet, and so on, which are successfully applied in semantic segmentation (18), object detection (19 , 20), and image recognition and segmentation (21).

As CNNs usually contain numerous parameters to achieve good ACC, it is necessary to explore simple and efficient network architectures, especially for our time series CT scan images, which can be regarded as 3D images. Multitask learning incorporates the benefits from several related tasks to excavate features better. It can take the underlying common information that may be ignored by single-task learning. Eventually, it improves the performance, the robustness and stability of disease detection, or image segmentation (22 , 23). Transfer learning can be used to improve a learner from one domain by transferring information from a related domain (24) and is widely used to initialize the parameters of a system (25 – 27). As the equipment information is usually acquired and is an essential feature to images, we propose to utilize the task of predicting equipment type to initialize the parameters of the 3D-CNN structure and finally employ it to improve the extracted features.

GCN

A graph neural network (GNN) was proposed in 2009 (28), which is based on the graph theory (29), building the foundation of all kinds of graph networks (30 – 33). As one of the most famous graph networks, GCN mainly

applies the convolution of Fourier transform and Taylor's expansion formula to improve filtering performance (34). With its excellent performance, GCN has been widely used in disease classification (34 – 38).

For graph theory, a node on the graph represents a subject's imaging data, and the edges establish interactions between each pair of nodes. Sarah et al. (35) integrated similarities between imaging information and distances between phenotypic information (e. g., sex, equipment type, and age) into edges for the prediction of autism spectrum disorder and the conversion to Alzheimer's disease (AD). Zhang et al. (36) combined an adaptive pooling scheme and a multimodal mechanism to classify Parkinson's disease (PD) status. Kazi et al. (37) designed different kernel sizes in spectral convolution to learn cluster-specific features for predicting mild cognitive impairment and AD. All these studies validate the effectiveness of GCN and show that the convolution operation is the key to prediction performance. On a graph, edges and edge weights determine the convolution operation. According to the characteristic of COVID-19 datasets, we design a COVID-19 graph to establish edges by considering equipment types, hospitals, and disease statuses. Current edge weights are roughly computed based on the correlation distance between image feature vectors, which is inaccurate and may affect the convolution performance. Therefore, we propose a combination mechanism, which combines the correlation distance of extracted features and the scores from 3D-CNN, to better compute the edge weights.

Methodology

The proposed method in this paper consists of two key parts. Using 3D-CNN, we extract image features from 3D-CT images and get an initial score for every subject. By designing our COVID-19 graph in GCN, we accomplish the COVID-19 diagnosis task. We first introduce the proposed 3D-CNN framework for feature extraction. Then we present the proposed COVID-19 graph and GCN. The overview of the proposed diagnosis mechanism is shown in Figure 1 .

FIGURE 1

Overview of the proposed coronavirus disease 2019 (COVID-19) diagnosis framework.(A)Feature extraction.(B)GCN structure for final diagnosis.

Feature Extraction by Using 3D-CNN

The paper applies a 3D-CNN network (39 , 40) to extract features from 3D-CT images. We first use z-score standardization to process the initial CT scans. Since the acquired datasets are not uniform and the 3D-CT images are of different sizes, we converted all the 3D-CT images into the same size of 64 × 64 × 32. Specifically, the 3D-CNN model has six convolutional layers and six max-pooling layers with a rectified linear unit (ReLU) as its activation function. The details of the 3D-CNN structure are shown in Figure 2 . In the first layer, C30@3 × 3 × 3 denotes there are 30 convolution kernels and the kernel size is 3 × 3 × 3. P2 × 2 × 2 denotes the size of the pooling layer is 2 × 2 × 2.

FIGURE 2

Details of the three-dimensional convolutional neural network (3D-CNN) structure for feature extraction.

Transfer learning is a widely used machine learning technique especially for comparatively little data in many fields (41 , 42), which enables scientists to benefit from the knowledge gained from a previously trained model for a related task. Specifically, by applying transfer learning, we can exploit the valuable information that has been learned in one task to improve generalization in another. The popularly used machine learning transfers the weights that a network has learned at " task A" to a new " task B."

As there are more than millions of parameters in our 3D-CNN and <1, 000 samples for training 3D-CNN, it has great significance to applying transfer learning on our COVID-19 diagnosis task. In view of the fact the equipment type is an important factor that affects acquired images and is easy to acquire, we design a transfer learning method to transfer the weights of 3D-CNN based on the known equipment type, as shown in Figure 1 . There are two tasks, including predicting equipment type and diagnosing COVID-19. We first employ 3D-CT images and their corresponding equipment type labels to train the first system of predicting equipment type. Then we transfer the weights of the 3D-CNN in the first system to the second system of diagnosing COVID-19. Finally, we adopt 3D-CT images and their corresponding COVID-19 labels in a training set to train the COVID-19 diagnosis system.

After the above processes, we get a well-trained COVID-19 diagnosis system by using transfer learning. Further, we utilize the trained COVID-19 diagnosis system to score every subject and extract all samples' features from the 3D-CNN structure. As shown in Figure 1 , we finally get a 150 × 1 feature vector and a score value for every subject. Every extracted feature vector composes a node on the graph in GCN, and score values are used to establish edges between nodes. High-dimensional feature vectors will increase the burden on the following GCN, and we use recursive feature elimination (RFE) (43) to select features from the 150 × 1 feature vector to reduce the feature vector's dimensions. In detail, given an estimator (e. g., ridge classifier) that assigns weights to features, RFE selects features by recursively considering smaller and smaller sets of features. First, the estimator is trained on the initial set of features, and the importance of each feature is obtained. Then the least important features are pruned from the current set of features. This procedure is recursively repeated on the pruned set until the desired number of features to be selected is eventually reached (44 – 46).

GCN

Compared with traditional neural networks, GCN makes use of graph theory to improve performance, and the graph in GCN plays the role of filtering noise. On a graph, a node represents the feature vector of a subject, and an edge denotes the interaction between corresponding pair nodes. Graph theory takes all nodes on the graph to perform convolution, and edge weights are the key to the performance as they are the corresponding convolution coefficients. Thereby, they attract much attention (47 , 48). The

description of graph theory is shown in Figure 3 . As shown, a node represents a subject, and there are total of n subjects, and everyone is represented by a $1 \times m$ feature vector. In mathematical form, a graph with n nodes can be described as an $n \times m$ feature matrix pre-multiplied with an $n \times n$ adjacency matrix, where the adjacency matrix is composed of all edge weights. The $n \times n$ adjacency matrix plays the role of filtering noise. For example, for subject 1, the filtered feature 2 is computed as $x^{12} = \sum_{i=1}^{i=n} a_{1i} \times x_{i2}$. The convolution coefficients determine the filtering effect, and improving them is our main contribution in this paper. There are usually millions of parameters in 3D-CNN, whereas only thousands of COVID-19 infected and non-infected subjects in the training process, and the insufficiency of samples introduces noise in extracted features, which further deteriorates the final diagnosis performance. The existing noise is given in Figure 6 .

FIGURE 3

The description of graph theory.

In this subsection, we propose a COVID-19 graph in GCN to establish edges to fit in the characteristics of the diagnosis task, which finally plays the role of improving the adjacency matrix.

COVID-19 Graph

We first consider the differences in the equipment types in our datasets, so we divide all nodes on the COVID-19 graph into several clusters. Those subjects with the same equipment type compose a cluster, which means the

number of clusters matches the number of equipment types in our datasets. There are six clusters, including SIEMENS cluster, Philips cluster, NMS cluster, Minfound cluster, FMI cluster, and GE cluster. Every cluster corresponds to one equipment type. We do not establish the connections between nodes in different clusters, which can also be regarded as zero-weight edges. For nodes in the same cluster, we propose a novel method to establish their edges, which considers the hospital information and disease status of those training samples. On our COVID-19 graph, the proposed edge weights between two subjects in the same cluster are calculated as follows:

$$A(v, u) = Sim(F_v, F_u) \times (1 + r_H(H_v, H_u) + r_s(S_v, S_u)) \quad (1)$$

$$r_H(H_v, H_u) = \begin{cases} 1, & H_v = H_u \\ 0, & H_v \neq H_u \end{cases} \quad (2)$$

$$r_S(S_v, S_u) = \begin{cases} 1, & S_v = S_u \\ 0, & S_v \neq S_u \\ 0, & S_v \text{ or } S_u \text{ is unknown} \end{cases} \quad (3)$$

where all edge weights compose adjacency matrix A. $A(v, u)$ is the edge weight between subject v and subject u, Sim(·) denotes the similarity of imaging information, F_v and F_u represent the feature vectors of subject v and subject u, respectively. r_H represents the distance between hospitals, r_S represents the distance of disease status (the statuses of those training samples on the graph are known), S_v and S_u are the subjects' disease statuses (COVID-19 infected case or healthy case), and H_v and H_u denote their corresponding hospital. For example, if the images of subject v and subject u are acquired from the same hospital, we set $H_v = H_u$. If subject v and subject u are all COVID-19 infected cases (or healthy cases), we set $S_v = S_u$.

Edge Weights Based on Correlation Distance

The popular method for evaluating the similarity of imaging information is based on the correlation distance as follows (35):

$$\text{Sim}(F_v, F_u) = \exp\left(-\frac{[\rho(F_v, F_u)]^2}{2\sigma^2}\right) \quad (4)$$

where $\rho(\cdot)$ is the correlation distance function and σ is the width of the kernel. We compute the correlation distances between each pair of nodes, and the σ is set as the mean value of the correlation distances according to the work in Zhang et al. (36).

By Equations (1)–(4), we can get an adjacency matrix A_f, which represents the adjacency matrix constructed based on the correlation distance of extracted feature vectors.

Edge Weights Based on Scores

Using correlation distance to compute similarities as in Equation (4) is rough and deteriorates convolution performance to some extent. We propose a method to compute the similarities in view that the 3D-CNN has good capability to excavate in-depth features. Employing 3D-CNN to extract features from CT images, we also get a diagnosis score for every subject, as shown in Figure 1 . Based on these scores, the proposed similarities are calculated as follows:

$$\text{Sim}(F_v, F_u) = \exp\left(-\frac{[Score_v - Score_u]^2}{2\sigma^2}\right) \quad (5)$$

where $Score_v$ and $Score_u$ denote the scores of subject v and subject u from 3D-CNN diagnosis system, respectively, and σ is also the width of the kernel and is set as the mean value of the correlation distances according to the

work in Zhang et al. ([36]). Based on Equations (1)–(3), (5), we can also get an adjacency matrix A_s.

Combined Adjacency Matrix

After getting an adjacency matrix A_f based on correlation distance and an adjacency matrix A_s based on initial scores, we further design a method to combine the two adjacency matrices to get a robust adjacency matrix. The combined adjacency matrix A_c is calculated as follows:

$$A_c = a \times A_f + b \times A_s \quad (6)$$

where a and b are corresponding coefficients, and we set the two coefficients as 0. 5 in this paper.

Then we can form our COVID-19 graph, which includes the extracted features from 3D-CT images and the edges represented by the combined adjacency matrix.

Spectral Theory and GCN Structure

In GCN series methods, the adjacency matrix is processed to achieve a better filtering effect and computational efficiency ([49]). The spectral convolution can be described as the multiplication of a signal $x \in \mathbb{R}^n$ (a scalar for every node) with a filter $g_\theta = \text{diag}(\theta)$ by

$$g_\theta * x = U g_\theta(\Lambda) U^T x = \sum_{k=0}^{K} \theta_k T_k(\tilde{L}) x \quad (7)$$

where U is the matrix of eigenvectors and is computed from the formula $L = I_N - D^{-(1/2)} A_{ac} D^{-(1/2)} = U \Lambda U^T$. I_N is the identity matrix, and D represents the diagonal degree matrix. $g_\theta(\Lambda)$ is well-approximated by a

truncated expansion in terms of Chebyshev polynomials to the Kth order. θ_k is a vector of Chebyshev coefficients, and T_k is the Chebyshev polynomial function, $\tilde{L} = 2\lambda_{max}\Lambda - I_N$.

After the above spectral convolution is applied, combined adjacency matrix A_c is approximated by $\sum_{k=0}^{K}\theta_k T_k(\tilde{L})$. When polynomial order K is adjusted, a different filtering effect can be obtained. It is worth mentioning that the adjacency matrix is computed according to Equations (1)–(7).

As shown in Figure 1 , there are two graph convolutional layers with a ReLU function as the activation function and one softmax function as the final output layer. Let $\hat{A} = \sum_{k=0}^{K}\theta_k T_k(\tilde{L})$, and the formula of the two-layer GCN is as follows:

$$Z = softmax(\hat{A}\,ReLU(\hat{A}XW^{(0)})W^{(1)}) \quad (8)$$

The GCN model is trained and tested using the whole graph as input. Let F_i and F_j represent the feature vectors of subject i and subject j, respectively. $X = [F_1; F_2; ...; F_n]$ represents the feature vectors of all subjects. In the training and test processes, the feature vector F_i is approximated by $\sum_{j=1}^{j=n}\hat{A}_{i,j}F_j$ in every layer, and this is described by $\hat{A}X$. It is also embodied in Figure 3 . In Equation (8), Z represents the labels of training and test samples. In the training process, the input is X, the feature vector of every training sample is approximated by $\sum_{j=1}^{j=n}\hat{A}_{i,j}F_j$ in every layer, and the labels of test samples in Z are masked. The training samples and their labels are used for training GCN. In the test process, the input is also X, the feature vector of every test sample is approximated by $\sum_{j=1}^{j}$

= n A ^ i , j F j in every layer, and after a two-layer trained GCN, we get their prediction results. In other words, in the training process, the feature vectors of test samples are used to help update the feature vectors of training samples. In the test process, the feature vectors of training samples are also used to help update the feature vectors of test samples. This is the meaning of graph theory for classification, and this is presented in Figure 3 and Equation (8). In adjacency matrix A ^ , every row of elements corresponds to the convolution coefficients (convolution kernel) of a subject. For example, the *i* row of elements in A ^ represents the convolution coefficients of subject *i* as shown in Figure 3 .

Experiments and Results

The proposed methodology is implemented on a database of CT scan images from open sources (50 – 53). The dataset contains 399 COVID-19 infected cases and 400 normal controls with six equipment types. The equipment type information is shown in Table 1 .

TABLE 1

The equipment type information in the dataset.

The parameters of the proposed GCN structure are as follows: learning rate is 0. 005, dropout rate is 0. 1, l_2 regularization is 5×10^{-4} , the number of epochs is 200, the number of neurons per layer is 64, and the number of extracted features is 20. Classification ACC, SEN, specificity (SPE), and area under receive operation curve (AUC) are selected as evaluation metrics.

This section is divided into seven parts. First, we evaluate the performance of the proposed transfer learning method on the 3D-CNN structure. Second, we test the performance of our GCN structure by comparing it with traditional methods. Third, we present the effect of different equipment types on extracted features. Fourth, we analyze the influence of the number of extracted features. Fifth, we analyze the effect of the width of kernel. Sixth, we analyze the performance of the GCN method on other public datasets. Last, we compare our methods with related works.

Performance of the Proposed Transfer Learning Method

In the proposed feature extraction framework, the task of predicting equipment type is utilized to initialize the parameters of the 3D-CNN structure, as shown in Figure 1 . The transfer learning method improves the performance of 3D-CNN, and Table 2 shows the method's effectiveness on the diagnosis performance of the 3D-CNN diagnosis framework.

TABLE 2

Performance of the 3D-CNN, 3D-ResNet18, and PENet diagnosis frameworks with and without our transfer learning method (5-fold cross validation).

As shown in Table 2 , with our transfer learning method, the ACC of the 3D-CNN diagnosis framework increases by 1. 3%, SEN increases by 6. 1%, and AUC increases by 1. 1%, whereas SPE decreases by 3. 6%. The ACC of the 3D-CNN diagnosis framework reaches 95. 6%. These improvements support us getting better extracted features with our transfer learning method. In addition, we also test our transfer learning method on 3D-ResNet18 and

PENet (54). Table 2 shows that our transfer learning method can improve ACC by 1. 8 and 0. 7%. Compared to 3D-CNN, PENet has a little effect on performance improvement whereas 3D-ResNet18 deteriorates performance, and they consume much more time. In our paper, we use the simplest 3D-CNN structure to extract features as its stable performance.

Performance of Our GCN Diagnosis Framework

To integrate equipment type information, hospital information, and disease status information, we design our GCN structure to accomplish the diagnosis task based on the extracted features and initial scores from the 3D-CNN framework. In this subsection, we compare our novel GCN structure with some other classifiers based on the extracted features. The compared classifiers include multiple layer perception (MLP), support vector machine (SVM), random forest (RF), gradient boosting decision tree (GBDT), and traditional GCN (35). The results of the performance comparison are shown in Table 3 . Figure 4 shows the performance comparison by histograms, and it shows our GCN has better performance than others. Figure 5 describes the ROC curves of different methods, and it shows that our GCN has better AUC values.

TABLE 3

Diagnosis performance of different classifiers based on the extracted features (5-fold cross validation).

FIGURE 4

Performance of the different classifiers based on the extracted features.

FIGURE 5

ROC curves of the different classifiers based on the extracted features.

As shown in Table 3 , compared to the diagnosis performance of the 3D-CNN framework with transfer learning in Table 2 , there is virtually no performance improvement by using the 3D-CNN extracted features and then using traditional classifiers (MLP, SVM, RF, and GBDT) for the final diagnosis. The traditional GCN (35) has slight performance improvement, with mean ACC increasing by 1. 7% and mean SEN increasing by 3. 5%. Nonetheless, with our proposed COVID-19 graph in GCN, mean ACC increases by 2. 9%, mean SEN increases by 3. 7%, and mean SPE increases by 1. 9%. These results show that our COVID-19 graph can improve performance significantly. In short, with our COVID-19 graph in GCN, the performance of GCN gets significant improvement, with final mean ACC, SEN, SPE, and AUC reaching 98. 5, 99. 9, 97, and 99. 9%.

Effect of Equipment Type on Extracted Features and the Filtering Effect of Our Graph on Features

As there are no related works to evaluate the filtering effect of GCN series methods for disease prediction, we propose to describe it by comparing X with $A\char`\^ X$ . $X$ represents the feature matrix that is composed by all subjects' feature vectors, $A\char`\^$ is our adjacency matrix, and $A\char`\^ X$ represents the feature matrix after filtering. As there are no real feature values, we propose to use mean values to represent real feature values and use standard

deviation to describe noise level in this subsection. Figure 6 visualizes the filtering effect of the different-equipment-type graph on the extracted features by comparing X with $\hat{A}X$, and the detailed effect on mean and standard deviation is given in Table 4 . Figure 7 shows t-SNE visualization results of feature maps where we compare X with $\hat{A}X$.

FIGURE 6

The filtering effect of the different-equipment-type graph on the extracted features by comparing X with $\hat{A}X$. X is the original feature matrix of all equipment types' subjects, which is represented by the blue line, and $\hat{A}X$ represents the filtered features by pre-multiplying with the adjacency matrix $\hat{A}$, which is represented by the red line.

TABLE 4

The mean values and standard deviations of the top nine most discriminative features in our six clusters.

FIGURE 7

The t-SNE visualization results of feature maps, where we compare X with $\hat{A}X$.

As shown in Figure 6 and Table 4 , there is much difference in the mean values of the same feature between different clusters. For example, the mean values of feature 1 with X in our six clusters are 1. 03, 1. 03, 1. 96, 1.

48, 1. 42, and 0. 93. The mean values of feature 2 with X in our six clusters are 0. 69, 0. 64, 1. 22, 0. 96, 1. 00, and 0. 66. These results validate that there are differences in images between different equipment types. Based on the big differences, we design our COVID-19 graph to divide all samples into several clusters (a cluster includes those samples acquired by one kind of equipment type) and establish edge connections between those samples from the same cluster. By pre-multiplying with adjacency matrix $\hat{A}$, the noise in extracted features is well-suppressed, as shown in Figure 6 , where red lines have a small fluctuation and blue lines have a big fluctuation.

In Table 4 , it is also shown that there are different noise levels between different features in the same cluster and also different noise levels between the same features in different clusters. For example, for feature 1, the standard deviations of our six clusters with X are 0. 32, 0. 32, 0. 86, 0. 70, 0. 12, and 0. 23. For feature 2, the standard deviations of our six clusters with X are 0. 16, 0. 15, 0. 48, 0. 37, 0. 08, and 0. 14. These results show that there is much noise on the extracted features in the NMS cluster and the FMI cluster. Furthermore, the comparison of the mean values and standard deviations of $\hat{A}X$ with those of X shows that mean values were kept stable, whereas standard deviations reduced significantly. For example, for all features, the standard deviations of four clusters (i. e., SIEMENS, NMS, MINFOUND, and GE) are very small. These results show that the GCN has an excellent filtering effect.

Effect of the Number of Extracted Features
The discriminative features are extracted from CT images by using 3D-CNN and initially form a 150 × 1 feature vector for every subject. Further, we

apply RFE to select the principal features. The effect of the number of final selected features on diagnosis ACC is shown in Figure 8 , where the number of extracted features varies from 0 to 150.

FIGURE 8

The effect of the number of extracted features on diagnosis accuracy.

As shown in Figure 8 , the ACC value increases to a stable value rapidly and maintains stability after the number exceeds 10. As a large number can increase the burden on GCN, we set the number of the extracted features as 20 in this paper.

Effect of the Width of Kernel

The width of kernel K in Equation (7) means the filter learned for neighbors K hops away from the node at the center of the receptive field, and it affects classification performance according to a previous study (35). We test the effect of K on performance in this subsection. Here, we test $K \in \{1, 2, 3, 4, 5, 6\}$. By setting different K values in our GCN method, we get their corresponding ACC as {97. 9, 98. 3, 98. 5, 98. 3, 98. 1, 98. 1%}.

As the above results shows, there are a few variations on ACC with different K values. Specifically, with K set as 3, the best ACC is reached, and this result is consistent with the work in Parisot et al. (35) and Ktena et al. (38). Therefore, we set $K = 3$ in our experiments.

Performance of the GCN Method on Other Public Datasets

There are several large public datasets available, but no equipment type information is included. In this subsection, we further test the GCN method by combining other datasets. The experiment includes the dataset from http://ncov-ai. big. ac. cn/download? lang= en (55) and the dataset from https://mosmed. ai/datasets/covid19_1110 (56). Constrained by our computer memory, we selected a total of 1, 560 COVID-19 cases and 1, 560 NC cases randomly from the above datasets. As there is no equipment type information in the above datasets, we use 3D-CNN to extract features, ignoring the transfer learning method, and we treat all samples as one cluster in the COVID-19 graph. The experimental results are listed in Table 5 .

TABLE 5

Diagnosis performance of the GCN method on the large dataset (5-fold cross validation).

As shown in Table 5 , with a large dataset, the performance of COVID-19 diagnosis is satisfactory with an ACC reaching 99. 5% by using the MLP method. Compared to the ACC with the MLP method, the ACC by using our GCN method increases by 0. 3%. Compared to the 1. 7% ACC improvement in Table 3 , the little performance improvement validates that our GCN method relatively adapts to the few-shot learning task.

Comparison With Related Works

Table 6 shows the diagnosis performance of our method and related methods. It is observed that our method achieves the best ACC, with it reaching 98. 5%. Compared to related works, our method improves the ACC by 2. 6–13. 6%. In terms of SEN and SPE, our method also shows the best performance.

TABLE 6

Algorithm comparison with the related works.

Discussion

Diagnosis of COVID-19 utilizing 3D-CT images is a few-shot learning task. Specifically, there are more than millions of parameters in our 3D-CNN and <1, 000 samples for training 3D-CNN. Although we propose a transfer learning method by predicting equipment type to initialize the 3D-CNN parameters, which improves the performance to some extent, there are still much noise existing in the extracted features as shown in Figure 6 . It is also shown that there is much difference on the same features between different clusters. This difference shows that equipment type has a big effect on images and supports the good performance of our transfer learning method. In view of the big difference, we propose our COVID-19 graph, which divides all samples into several clusters and samples with the same equipment type composing a cluster, to suppress the existing noise in extracted features. The purpose of our COVID-19 graph is to improve the filtering effect, and the filtering principle is shown in Figure 3 . With the application of our COVID-19

graph, Figure 6 shows that the noise is well-suppressed, and this is the key for our performance improvement.

Our main contribution is analyzing the effect of equipment type on images. By analyzing its effect on the extracted features, we proposed a transfer learning method and a GCN classifier for COVID-19 diagnosis. It is worth mentioning that there are still some limitations. Our GCN method is limited to a binary classification task, where the more important and difficult challenge is discriminating COVID-19 from other abnormal cases (e. g., pneumonia) and normal controls, and we will study this issue in our future work. Our work relatively adapts to a few-shot learning task as the advantage of GCN lies in its filtering effect.

Conclusions

In this study, we proposed a novel method based on 3D-CNN and GCN for COVID-19 diagnosis. The proposed method considers three pieces of information: the equipment type, hospital information, and the disease status of the training samples. Comparing the diagnosis results of our method with the results of using 3D-CNN for diagnosis shows that using GCN with the three pieces of information can result in a 4. 2% improvement on ACC. The comparison result validates that the three pieces of information are essential to CT images and effective for ACC improvement. The excellent performance of using the task of predicting equipment type to initialize the parameters of 3D-CNN also validates that the equipment type is a key information of CT images. The analysis of the extracted features from 3D-CNN in different clusters shows diversified noise levels across different

clusters. We can conclude that there exists disparate imaging quality with the use of different CT equipment. Finally, our method achieves excellent performance, with a diagnosis ACC reaching 98. 5%.

Data Availability Statement

The dataset used in this study is from the open sources and can be downloaded from the following links: https://github. com/ahmadhassan7/Covid-19-Datasets , https://www. medrxiv. org/ , https://www. biorxiv. org/ , https://www. kaggle. com/mohammadrahimzadeh/covidctset-a-large-covid19-ct-scans-dataset , and http://ncov-ai. big. ac. cn/download? lang= en , https://mosmed. ai/datasets/covid19_1110 .

Author Contributions

Conception, design, and statistical analysis were performed by XL and YZ. Drafting the manuscript and editing were performed by JW, QY, YL, and JT. All authors read and approved the final manuscript.

Funding

This work was supported by the Jiangsu Biobank of Clinical Resources (BM2015004).

Conflict of Interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Elgendi M, Fletcher RR, Howard N, Menon C, Ward R. The performance of deep neural networks in differentiating chest X-rays of COVID-19 patients from other bacterial and viral pneumonias. *Front Med.* (2020) 7: 550. doi: 10. 3389/fmed. 2020. 00550

2. Yoo SH, Geng H, Chiu TL, Yu SK, Cho DC, Heo J, et al. Deep learning-based decision-tree classifier for COVID-19 diagnosis from chest X-ray imaging. *Front Med.* (2020) 7: 427. doi: 10. 3389/fmed. 2020. 00427

3. Bernheim A, Mei X, Huang M, Yang Y, Fayad ZA, Zhang N, et al. Chest CT findings in coronavirus disease-19 (COVID-19): relationship to duration of infection. *Radiology.* (2020) 295: 685–91. doi: 10. 1148/radiol. 2020200463

4. Xie X, Zhong Z, Zhao W, Zheng C, Wang F, Liu J. Chest CT for typical 2019-nCoV pneumonia: relationship to negative RT-PCR testing. *Radiology.* (2020) 296: E41–5. doi: 10. 1148/radiol. 2020200343

5. Fang Y, Zhang H, Xie J, Lin M, Ying L, Pang P, et al. Sensitivity of chest CT for COVID-19: Comparison to RT-PCR. *Radiology.* (2020) 296: E115–7. doi: 10. 1148/radiol. 2020200432

6. Zu Z, Jiang M, Xu P, Chen W, Ni Q, Lu G, et al. Coronavirus disease 2019 (COVID-19): a perspective from China. *Radiology.* (2020) 296: E15–25. doi: 10. 1148/radiol. 2020200490

7. Chung M, Bernheim A, Mei X, Zhang N, Huang M, Zeng X, et al. CT imaging features of 2019 novel coronavirus (2019-nCoV). *Radiology.* (2020) 295: 202–7. doi: 10. 1148/radiol. 2020200230

8. Pan Y, Guan H. Imaging changes in patients with 2019-nCov. *Eur Radiol.* (2020) 30: 3612–3. doi: 10. 1007/s00330-020-06713-z

9. Sun L, Mo Z, Yan F, Xia L, Shan F, Ding Z, et al. Adaptive feature selection guideddeep forest for COVID-19 classification with chest CT. *arXiv.* (2020) 2005. 03264. doi: 10. 1109/JBHI. 2020. 3019505

10. Mahmud T, Rahman MA, Fattah SA. CovXNet: a multi-dilation convolutional neural network for automatic COVID-19 and other pneumonia detection from chest X-ray images with transferable multi-receptive feature optimization. *Comput Biol Med.* (2020) 122: 103869. doi: 10. 1016/j. compbiomed. 2020. 103869

11. Wang S, Kang B, Ma J, Zeng X, Xiao M, Guo J, et al. A deep learning algorithm using CT images to screen for corona virus disease (COVID-19). *medRxiv.* (2020). doi: 10. 1101/2020. 02. 14. 20023028

12. Zheng C, Deng X, Fu Q, Zhou Q, Feng J, Ma H, et al. Deep learning-based detection for COVID-19 from chest CT using weak label. *medRxiv.* (2020). doi: 10. 1101/2020. 03. 12. 20027185

13. Hall L, Paul R, Goldgof DB, Goldgof GM. Finding covid-19 from chest x-rays using deep learning on a small dataset. *arXiv.* (2020) 2004. 02060. doi: 10. 36227/techrxiv. 12083964. v4

14. Kumar R, Arora R, Bansal V, Sahayasheela VJ, Buckchash H, Imran J, et al. Accurate prediction of COVID-19 using chest X-Ray images through deep feature learning model with SMOTE and machine learning classifiers. *medRxiv.* (2020). doi: 10. 1101/2020. 04. 13. 20063461

15. Kolozsvari LR, Berczes T, Hajdu A, Gesztelyi R, Tlba A, Varga I, et al. Predicting the epidemic curve of the coronavirus (SARS-CoV-2) disease (COVID-19) using artificial intelligence. *medRxiv.* (2020). doi: 10. 1101/2020. 04. 17. 20069666

16. Khalifa NEM, Taha MHN, Hassanien AE, Elghamrawy S. Detection of coronavirus (COVID-19) associated pneumonia based on generative adversarial networks and a fine-tuned deep transfer learning model using chest X-ray dataset. *arXiv.* (2020) 2004. 01184.

17. Loey M, Smarandache F, Khalifa NEM. Within the lack of chest COVID-19 X-ray dataset: a novel detection model based on GAN and deep transfer learning. *Symmetry.* (2020) 12: 651. doi: 10. 3390/sym12040651

18. Chen L, Papandreou G, Kokkinos I, Murphy K, Yuille AL. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv.* (2014) 1412. 7062.

19. Lin T, Goyal P, Girshick R, He K, Dollár P. Focal loss for dense object detection. *arXiv.* (2017) 1708. 02002v2. doi: 10. 1109/ICCV. 2017. 324

20. Ren S, He K, Girshick R, Sun J. Faster R-CNN: towards real-time object detection with region proposal networks. *arXiv.* (2016) 1506. 01497v3.

21. Han K, Guo J, Zhang C, Zhu M. Attribute-aware attention model for fine-grained representation learning. In: *Proceedings of the 26th ACM International Conference on Multimedia* , Seoul (2018). p. 2040–8. doi: 10. 1145/3240508. 3240550

22. El-Sappagh S, Abuhmed T, Islam SR, Kwak KS. Multimodal multitask deep learning model for Alzheimer's disease progression detection based on time series data. *Neurocomputing.* (2020) 412: 197–215. doi: 10. 1016/j. neucom. 2020. 05. 087

23. Tam CM, Zhang D, Chen B, Peters T, Li S. Holistic multitask regression network for multiapplication shape regression segmentation. *Med Image Anal.* (2020) 65: 101783. doi: 10. 1016/j. media. 2020. 101783

24. Weiss K, Khoshgoftaar TM, Wang D. A survey of transfer learning. *J Big Data.* (2016) 3: 9. doi: 10. 1186/s40537-016-0043-6

25. Wang Y, Liu Y, Chen W, Ma Z, Liu T. Target transfer Q-learning and its convergence analysis. *Neurocomputing.* (2020) 392: 11–22. doi: 10. 1016/j. neucom. 2020. 02. 117

26. Lin J, Zhao L, Wang Q, Ward R, Wang Z. DT-LET: Deep transfer learning by exploring where to transfer. *Neurocomputing.* (2020) 390: 99–107. doi: 10. 1016/j. neucom. 2020. 01. 042

27. Taherkhani A, Cosma G, McGinnity TM. AdaBoost-CNN: an adaptive boosting algorithm for convolutional neural networks to classify multi-class

imbalanced datasets using transfer learning. *Neurocomputing.* (2020) 404: 351–66. doi: 10. 1016/j. neucom. 2020. 03. 064

28. Scarselli F, Gori M, Tsoi AC, Hagenbuchner M, Monfardini G. The graph neural network model. *IEEE Trans Neural Netw.* (2009) 20: 61–80. doi: 10. 1109/TNN. 2008. 2005605

29. Bapat RB. *Graphs and Matrices* . New York, NY: Springer (2010).

30. Shuman DI, Narang SK, Frossard P, Ortega A, Vandergheynst P. The emerging field of signal processing on graphs: extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Proc Mag.* (2013) 30: 83–98. doi: 10. 1109/MSP. 2012. 2235192

31. Zhao M, Chan RH, Chow TW, Tang P. Compact graph based semi-supervised learning for medical diagnosis in Alzheimer's disease. *IEEE Signal Proc Lett.* (2014) 21: 1192–6. doi: 10. 1109/LSP. 2014. 2329056

32. Defferrard M, Bresson X, Vandergheynst P. Convolutional neural networks on graphs with fast localized spectral filtering. *arXiv.* (2016) 1606. 09375.

33. Duvenaud D, Maclaurin D, Aguilera-Iparraguirre J, Gomez-Bombarell R, Hirzel T, Aspuru-Guzik A, et al. Convolutional networks on graphs for learning molecular fingerprints. *arXiv.* (2015) 1509. 09292v2.

34. Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. *arXiv.* (2016) 1609. 02907.

35. Parisot S, Ktena SI, Ferrante E, Lee M, Guerrero R, Glocker B, et al. Disease prediction using graph convolutional networks: application to autism spectrum disorder and Alzheimer's disease. *Med Image Anal.* (2018) 48: 117–30. doi: 10. 1016/j. media. 2018. 06. 001

36. Zhang Y, Zhan L, Cai W, Thompson P, Huang H. Integrating heterogeneous brain networks for predicting brain disease conditions. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention* . Shenzhen (2019). p. 214–22. doi: 10. 1007/978-3-030-32251-9_24

37. Kazi A, Shekarforoush S, Krishna SA, Burwinkel H, Vivar G, Kortüm K, et al. InceptionGCN: Receptive field aware graph convolutional network for disease prediction. In: *International Conference on Information Processing in Medical Imaging* . Hong Kong (2019). p. 73–85. doi: 10. 1007/978-3-030-20351-1_6

38. Ktena SI, Parisot S, Ferrante E, Rajchl M, Lee M, Glocker B, et al. Metric learning with spectral graph convolutions on brain connectivity networks. *Neuroimage.* (2018) 169: 431–42. doi: 10. 1016/j. neuroimage. 2017. 12. 052

39. Ji S, Xu W, Yang M, Yu K. 3D convolutional neural networks for human action recognition. *IEEE Trans Pattern Anal.* (2013) 35: 221–31. doi: 10. 1109/TPAMI. 2012. 59

40. Kumawat S, Raman S. LP-3DCNN: unveiling local phase in 3D convolutional neural networks. In: *Proceedings of the IEEE Conference on*

Computer Vision and Pattern Recognition . Long Beach, CA (2019). p. 4903–12. doi: 10. 1109/CVPR. 2019. 00504

41. Li Y, He K, Xu D, Luo D. A transfer learning method using speech data as the source domain for micro-Doppler classification tasks. *Knowl Based Syst.* (2020) 209: 106449. doi: 10. 1016/j. knosys. 2020. 106449

42. Yang X, Zhang Y, Lv W, Wang D. Image recognition of wind turbine blade damage based on a deep learning model with transfer learning and an ensemble learning classifier. *Renew Energy.* (2021) 163: 386–97. doi: 10. 1016/j. renene. 2020. 08. 125

43. Guyon I, Weston J, Barnhill S, Vapnik V. Gene selection for cancer classification using support vector machines. *Mach Learn.* (2002) 46: 389–422. doi: 10. 1023/A: 1012487302797

44. Song X, Elazab A, Zhang Y. Classification of mild cognitive impairment based on a combined high-order network and graph convolutional network. *IEEE Access.* (2020) 8: 42816–27. doi: 10. 1109/ACCESS. 2020. 2974997

45. Shao Z, Yang S, Gao F, Zhou K, Lin, P. A new electricity price prediction strategy using mutual information-based SVM-RFE classification. *Renew Sust Energy Rev.* (2017) 70: 330–41. doi: 10. 1016/j. rser. 2016. 11. 155

46. Sahran S, Albashish D, Abdullah A, Abd Shukor N, Pauzi, SHM. Absolute cosine-based SVM-RFE feature selection method for prostate histopathological grading. *Artif Intell Med.* (2018) 87: 78–90. doi: 10. 1016/j. artmed. 2018. 04. 002

47. Xu K, Li C, Tian Y, Sonobe T, Kawarabayashi KI, Jegelka S. Representation learning on graphs with jumping knowledge networks. *arXiv.* (2018) 1806. 03536.

48. Liu Z, Chen C, Li L, Zhou J, Li X, Song L, et al. Geniepath: graph neural networks with adaptive receptive paths. In: *Proceedings of the AAAI Conference on Artificial Intelligence* . Honolulu, HI (2019). p. 4424–31. doi: 10. 1609/aaai. v33i01. 33014424

49. Defferrard M, Bresson X, Vandergheynst P. Convolutional neural networks on graphs with fast localized spectral filtering. *arXiv* . (2016) 1606. 09375.

50. Github. Available online at: https://github. com/ahmadhassan7/Covid-19-Datasets/tree/master/AHP-covid19-ctscans/Anonymized20200225 (accessed April 18, 2020).

51. Medrxiv. (2020). Available online at: https://www. medrxiv. org/ (accessed September 20, 2020).

52. Biorxiv. (2020). Available online at: https://www. biorxiv. org/ (accessed September 20, 2020).

53. Kaggle. Available online at: https://www. kaggle. com/mohammadrahimzadeh/covidctset-a-large-covid19-ct-scans-dataset (accessed August 27, 2020).

54. Huang SC, Kothari T, Banerjee I, Chute C, Ball RL, Borus N, et al. PENet-a scalable deep-learning model for automated diagnosis of pulmonary

embolism using volumetric CT imaging. *NPJ Digit Med.* (2020) 3: 1–9. doi: 10. 1038/s41746-020-00310-6

55. Zhang K, Liu X, Shen J, Li Z, Sang Y, Wu X, et al. Clinically applicable AI system for accurate diagnosis, quantitative measurements and prognosis of covid-19 pneumonia using computed tomography. *Cell.* (2020) 181: 1423–33. doi: 10. 1016/j. cell. 2020. 04. 045

56. Morozov SP, Andreychenko AE, Pavlov NA, Vladzymyrskyy AV, Ledikhova NV, Gombolevskiy VA, et al. MosMedData: chest CT scans with COVID-19 related findings dataset. *arXiv.* (2020) 2005. 06465. doi: 10. 1101/2020. 05. 20. 20100362

57. Abbas A, Abdelsamea MM, Gaber MM. Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network. *arXiv.* (2020) 2003. 13815. doi: 10. 1101/2020. 03. 30. 20047456

58. Shi F, Xia L, Shan F, Wu D, Wei Y, Yuan H, et al. Large-scale screening of covid-19 from community acquired pneumonia using infection size-aware classification. *arXiv.* (2020) 2003. 09860.

59. Sethy PK, Behera SK. Detection of coronavirus disease (COVID-19) based on deep features. *Preprints.* (2020) 2020030300. doi: 10. 20944/preprints202003. 0300. v1

60. Narin A, Kaya C, Pamuk Z. Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks. *arXiv.* (2020) 2003. 10849.

61. Jin C, Chen W, Cao Y, Xu Z, Zhang X, Deng L, et al. Development and evaluation of an AI system for COVID-19 diagnosis. *medRxiv.* (2020). doi: 10. 1101/2020. 03. 20. 20039834

62. Butt C, Gill J, Chun D, Babu BA. Deep learning system to screen coronavirus disease 2019 pneumonia. *Appl Intell.* (2020). doi: 10. 1007/s10489-020-01714-3. [Epub ahead of print].

63. Bai HX, Wang R, Xiong Z, Hsieh B, Chang K, Halsey K, et al. AI augmentation of radiologist performance in distinguishing COVID-19 from pneumonia of other etiology on chest CT. *Radiology.* (2020) 296: 201491. doi: 10. 1148/radiol. 2020201491