

Received February 9, 2021, accepted February 13, 2021, date of publication February 16, 2021, date of current version March 3, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3059853

Hierarchical Graph Attention Based Multi-View Convolutional Neural Network for 3D Object Recognition

HUI ZENG^{1,2}, TIANGENG ZHAO¹, RUTING CHENG¹, FUZHOU WANG¹, AND JIWEI LIU^{1,2}

¹Beijing Engineering Research Center of Industrial Spectrum Imaging, School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China

²Shunde Graduate School, University of Science and Technology Beijing, Foshan 528399, China

Corresponding author: Jiwei Liu (liujiwei@ustb.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61973029 and Grant 61375010, and in part by the Fundamental Research Funds for the Central Universities under Grant FRF-BD-19-002A.

ABSTRACT For multi-view convolutional neural network based 3D object recognition, how to fuse the information of multiple views is a key factor affecting the recognition performance. Most traditional methods use max-pooling algorithm to obtain the final 3D object feature, which does not take into account the correlative information between different views. To make full use of the effective information of multiple views, this paper introduces the hierarchical graph attention based multi-view convolutional neural network for 3D object recognition. At first, the view selection module is proposed to reduce redundant view information in multiple views, which can select the projective views with more effective information. Then, the correlation weighted feature aggregation module is proposed to better fuse multiple view features. Finally, the hierarchical feature aggregation network structure is designed to further to make full use of the correlation information of multiple views. Extensive experimental results have validated the effectiveness of the proposed method.

INDEX TERMS 3D object recognition, multi-view convolutional neural network, graph attention network, feature aggregation.

I. INTRODUCTION

With the rapid development of 3D data acquisition technologies, the number of the 3D models is growing explosively. How to analyse the 3D model has attracts more and more researchers' attentions. The 3D object recognition is an important research direction in the field of 3D model analysis, which can be widely used in virtual reality, medical diagnosis, intelligent robot, remote sensing, autonomous driving, etc [1]–[3]. The 3D Compared with 2D images, 3D models contain more geometric, shape and scale information, which can be represented with different formats, such as point clouds, meshes, volumetric grids and so on [4]–[8]. The relatively mature 2D object recognition methods can't be directly applied to 3D object recognition. So it is necessary for us to make an in-deep study of 3D object recognition.

The associate editor coordinating the review of this manuscript and approving it for publication was Gustavo Olague.

In recent years, deep learning has become the most popular technique in the field of 2D image analysis and understanding. Researchers have proposed a large number of deep neural network models for image analysis and understanding tasks such as image classification, image segmentation, image retrieval and so on [9]. These deep learning based methods have obtained better performance than traditional methods. Because the 3D point clouds and 2D images have different data structure, the deep neural network models designed for the 2D images can't be directly applied in the 3D point cloud analysis. Compared with the 2D image based deep learning method, the deep learning method for the 3D point cloud analysis is still in its infancy, and there are still many key technical problems to be solved, such as the high dimensionality and the unstructured nature of 3D point clouds [1].

In order to use the relatively mature 2D deep learning technique for 3D point cloud analysis, the researchers have proposed 3D deep learning methods based on multi-view representation. The basic idea of this kind of method is

similar to the principle of people's object classification and recognition mechanism from multiple views. The most representative work is the 3D shape recognition method based on multi-view convolutional neural networks (MVCNN) [4]. Firstly, the multiple projective images of the 3D model are sent to the convolution neural network for feature learning. Then the view pooling layer is used to fuse these features together for learning the feature of the 3D model. One of the key problems of the multi-view based 3D object recognition algorithms is how to effectively aggregate the features extracted from multiple views to generate the 3D object feature. Most traditional methods [4], [7] fuse the multi-view features through the max-pooling layer. The max-pooling layer has permutation invariance, so it is stable for some data disturbance. But the correlation information between multiple views is ignored. As the recurrent neural network [9] has achieved a very prominent performance in processing sequence information, many researchers [10]–[12] using it in multi-view object recognition by arranging multiple views into a sequence according to their projective positions. For the recurrent neural network based multi-view 3D object recognition method, the position information can be used to aggregate these views to a 3D object feature. With the deepening of research, researchers found that some views play a more critical role for the final recognition and some views may cause interference for recognition. So we can increase the weights of the related views and reduce the influence of the irrelevant views by adding attention mechanism to further improve the performance of 3D object recognition.

The attention mechanisms have almost become a necessary module in many computer vision tasks. They can focus on the most relevant information of the input to make decision, and then the performance of the model can be improved effectively. So far, the researchers have proposed a variety of attention mechanisms, such as hard attention, soft attention, global attention, local attention, self-attention, etc. When an attention mechanism is used for a sequence based task, the researchers usually use self-attention. The self-mechanism not only can be used to improve the performance of the recurrent neural network or the convolutional neural network, but also can be used to construct a powerful model on the machine translation task. So following the self-attention strategy, the graph attention network (GAT) has been successfully used in multi-view network framework, which can operate on graph-structured data [13]. The GAT computes the hidden representations of each node in the graph by attending over its neighbors. It is suitable to handle 3D data, and its computation efficiency is high because it is parallelizable across node neighboring pairs. Furthermore, the neighboring graph nodes reveal different importance to the central graph node, which can be achieved by specifying arbitrary weights to the neighbors. Based on the above analysis, we select GAT to design the view-selection module in this paper. The position information of the multiple views can be considered in the process of multi-view feature aggregation, and the degrees of influence of each view are

distinguished in the process of 3D object feature learning. Extensive experimental results have shown that our proposed method has better performance than the state-of-art methods.

The major contributions of this paper can be summarized as follows:

(i) We design the hierarchical Graph Attention based Multi-view Convolutional Neural Network (GA-MVCNN) for 3D object recognition, which is a hierarchical feature aggregation network using graph attention mechanism to make full use of the correlation information of multiple views.

(ii) To reduce the redundant view information in multi-view based 3D object recognition method, the graph attention mechanism based view selection module is proposed to retain the projective views with abundant effective information and discard the projective views with less effective information.

(iii) The correlation weighted feature aggregation module is proposed to enhance the efficiency of information utilization, which can assign different weights to multiple views by measuring the information of them.

The rest of this paper is organized as follows. Section 2 gives the related works about 3D object recognition and graph neural networks. Section 3 introduces the proposed 3D object recognition method in detail including view feature extraction, GAT based view selection, hierarchical feature aggregation and network training. Section 4 reports the experimental results and the detailed analysis. Finally, conclusions are provided in Section 5.

II. RELATED WORKS

A. 3D OBJECT RECOGNITION

In recent years, great progress has been made in the field of 3D object recognition. According to the representation formats of 3D object data, existing 3D object recognition methods can be divided into three categories: voxel based methods, point cloud based methods, and multi-view based methods.

The voxel based 3D object recognition methods use the voxels to represent the 3D object, which can be sent to the 3D deep neural network for learning and recognition. The voxels can be considered as 3D extensions of the 2D pixels, and they have regular structures in 3D space. The deep neural network designed for 2D image pixels [14] can be easily extended to process voxels data. Wu *et al.* proposed 3D ShapeNets [15] that first applied voxels to deep neural networks. Maturana *et al.* proposed VoxNet [16] that used 3D convolutional network to process voxels for 3D object recognition. Choy *et al.* proposed 3D-R2N2 [17] that took single or multiple images as input and used voxels to reconstruct objects in the mesh. Sedaghat *et al.* proposed ORION [18] that could simultaneously predict object class labels and orientation labels to reduce the influence of object orientation on recognition accuracy. Li *et al.* [19] proposed FPNN that employs field probing filters to efficiently extract

features from them. Generally speaking, the 3D object recognition methods based on voxel representation can be directly extended from the 2D deep neural network, but some 3D structural information will be lost in the process of transforming from point clouds to voxels. This is because that some close 3D points may be mapped into the same voxel. In addition, the low resolution of voxels will cause the loss of local information, and the high resolution of voxels will increase the memory consumption and calculation time of the algorithm, which greatly limits the application scope of this kind of methods.

The point cloud based 3D object recognition methods directly take the 3D point clouds as the input of the deep neural network to realize end-to-end learning. The point clouds have the characteristics of irregular spatial relationship, so the existing 2D image recognition methods based on deep neural network cannot be directly applied or extended to point cloud based 3D object recognition. The most representative algorithm is PointNet [20], which use unordered point clouds as input and can avoid partial data structure information loss caused by point cloud processing. PointNet has been successfully used in 3D point cloud classification and segmentation. In order to enhance the analysis ability of PointNet for complex scenes, Charles *et al.* [21] proposed a hierarchical neural network named PointNet++, which applies PointNet recursively on a nested partitioning of the input point set. Inspired by PointNet and PointNet++, Wang *et al.* [22] proposed a dynamic graph convolutional neural network (DGCNN), which uses EdgeConv layer to obtain local features. It can not only ensure permutation invariance, but also capture local geometric features. By stacking or recycling EdgeConv modules, it could extract the global shape information and achieved good results. Different from the classical feature learning methods proposed by PointNet and PointNet++, Su *et al.* [23] designed a new point cloud processing method named SplatNet, which could directly operate on the point clouds and transform the concept of receptive field into irregular point clouds. This kind of methods need not convert 3D points to voxel or multi view representation, which can avoid the loss of information in the process of format transformation. However, the unstructured disordered and large amount of 3D point clouds also bring many technical challenges to the design of 3D deep neural network model.

The multi-view based 3D object recognition methods use a series of 2D projective images of 3D objects to learn 3D features. At first, this kind of methods use 2D convolutional neural network to learn the feature of each projective image. Then the extracted features are fused to obtain 3D shape feature. The pioneering work is MVCNN [4], which uses the convolutional neural network (VGG-M) [24] to learn the features of different projective images and uses the view pooling layer to fuse these features together to generate a feature vector of the 3D object. Up to now, most multi-view representation based 3D object recognition methods are designed based on the structure of MVCNN. The disadvantage of MVCNN was that the maximum pooling operation of the view pooling

layer ignored the correlation information between various views. Feng *et al.* [25] proposed a group-view convolutional neural network named GVCNN, which introduced a grouping module for the neglect of correlation between views. Charles *et al.* [26] improved the performance of MVCNN by using different azimuth and elevation angle rotation to increase the training sample, and introducing multi-resolution 3D filter to obtain different scales information of objects. Kanezaki *et al.* [27] proposed a new neural network named RotationNet, which could not only classify objects but also estimate their pose. Unlike the previously mentioned methods whose perspective tags were known during training, RotationNet could learn in an unsupervised manner from unaligned object data sets. In addition, RotationNet could perform well in the case of multiple views with fewer perspectives. He *et al.* [28] proposed a new loss function named triplet-center loss (TCL) for multi-view convolutional neural network, which solved the problem that the softmax loss function didn't consider the relationships within and between classes. Jiang *et al.* [29] proposed MLVCNN that adopted a novel view input strategy. Wei *et al.* [30] proposed a novel view-based Graph Convolutional Neural Network, dubbed as view-GCN, to recognize 3D shape based on graph representation of multiple views. As the 2D image based deep learning algorithms are mature, the multi-view based 3D object recognition methods have achieved the most excellent recognition performance. However, these methods are easy to be affected by mutual occlusion between objects, which will lose some structural information in the process of view projection, and has certain limitations in the selection of projective views. In addition, the view pooling operation used for multi-view feature fusion will also lose some effective information. Therefore, how to make full use of the correlation between different view features and mine the effective discriminative information contained in multi-view features is still worthy of further research.

B. GRAPH NEURAL NETWORKS

One of the reasons why deep learning methods can achieve outstanding performance in the fields of image classification, semantic segmentation and machine translation is that the data used in these fields belongs to Euclid space structure. However, for the graph structure data in non-Euclid space structure, such as social network, telephone communication network and biological network, their performances are not satisfactory [31]–[33]. Driven by the actual application requirements, how to deal with graph structure data has attracted more and more researchers' attentions.

The early researchers [34], [35] processed the data represented as directed acyclic graphs in the graph domain through recurrent neural networks. The graph neural network (GNN) was improved from the recurrent neural network [36], [37]. It can directly process more general graphs, such as recurrent graphs, directed graphs, undirected graphs, etc. Generally, the graph neural network consists of two parts: the propagation module and the output module. The propagation module

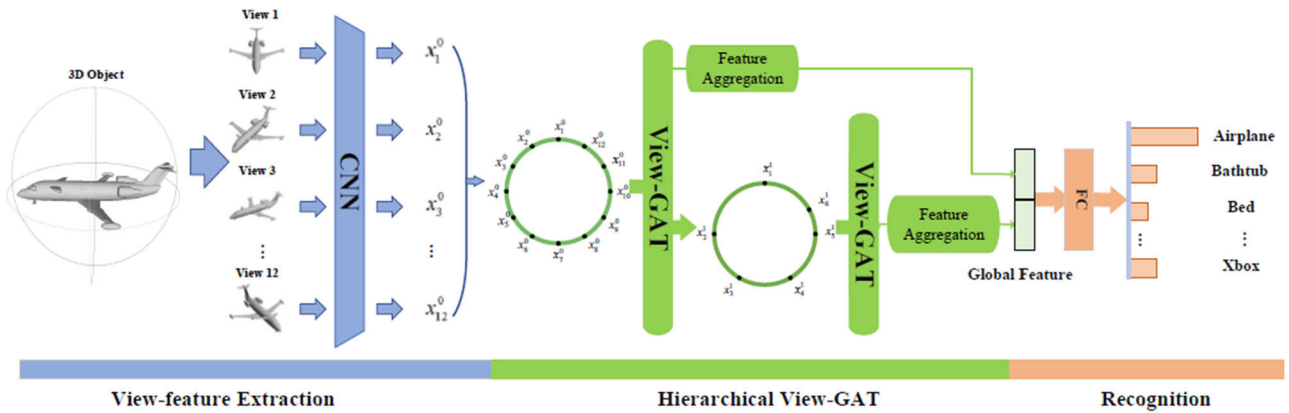


FIGURE 1. Multi-view 3D object recognition algorithm based on graph attention network.

is used to transmit node information and update node status. The output module defines different objective functions according to different task requirements based on the representation of nodes and connection edges. With the development of the graph neural network, the related algorithms are mainly divided into two kinds: spatial domain based methods and frequency domain based methods.

The methods based on frequency domain draw lessons from the idea of signal processing. Firstly, the graph signals are converted from spatial domain to frequency domain for convolution operation. Then, the graph signals are converted from frequency domain back to spatial domain. Bruna *et al.* [38] firstly introduced the idea of convolution into the graph neural network, and realized the transformation of the graph from the spatial domain to the frequency domain by means of the Laplace matrix of the graph. However, all the graph data needed to be loaded at the same time during calculation, which has high time complexity and could not be applied to large-scale graphs. Henaff *et al.* [39] improved Bruna's work to handle large-scale data tasks, and proposed unsupervised and supervised graph estimation method for data without given graph structure. By improving convolution kernel, convolution process and pooling process, Defferrard *et al.* [40] solved the problems of high computational complexity. Kipf and Welling [41] optimized the above approach by restricting the filter to operate in the neighborhoods of the nodes. From the above analysis, we can conclude that all the frequency domain based filter learning methods rely on the Laplace matrix. As the Laplace matrix is easily influenced by graph structure, the model trained on specific structure cannot be directly applied to the graphs with different structures.

The methods based on spatial domain are mainly inspired by the convolution operation in the convolutional neural network. The graph convolution is defined through the spatial relationship of the graph nodes, and the neighboring nodes within a specific range are included in it. It has local convolution invariance in the convolutional neural network.

Atwood and Towsley [42] proposed the diffusion convolutional neural networks (DCNNs) that handled node classification and graph classification tasks using H-hop matrix to represent each node. Niepert *et al.* [43] proposed the learning convolutional neural networks for graphs (PATCHY-SAN). It maps graph nodes and their adjacent nodes to fixed-length vectors, and then the features are extracted from them using convolutional neural networks. Monti *et al.* [44] improved and generalized the structure of the convolutional neural networks so that it could input and process non-Euclid structured data and learn the features of different task requirements. Hamilton *et al.* [45] proposed the Inductive representation learning on large graphs (GraphSAGE) that obtained the features of target nodes by learning an aggregation function to aggregate neighbor nodes, which solved the problem that the previous algorithm could not quickly obtain the features of new nodes. From the above study, we can found that the importance of each neighboring node to the center node of the graph structure data was different. To solve this problem, Veličković *et al.* [46] proposed the graph attention network that learned the weights of neighboring nodes by stacking the attention layers of the graph and realized the weighted aggregation to neighboring node, which could solve inductive problems and transductive problems.

III. THE PROPOSED METHOD

In this section, we introduce our proposed hierarchical Graph Attention based Multi-view Convolutional Neural Network (GA-MVCNN) for 3D object recognition. As shown in FIGURE 1, the proposed method has three main steps: (i) For Each 3D object, it is projected into a series of 2D images, and then the projective images are respectively sent to the convolutional neural network for view feature extraction. (ii) The extracted features of multiple views are aggregated inter-relatedly and selectively by view Graph attention network module and correlation weighted feature aggregation module to form a global feature. (iii) The global feature is used for 3D object recognition by the fully connection layer to

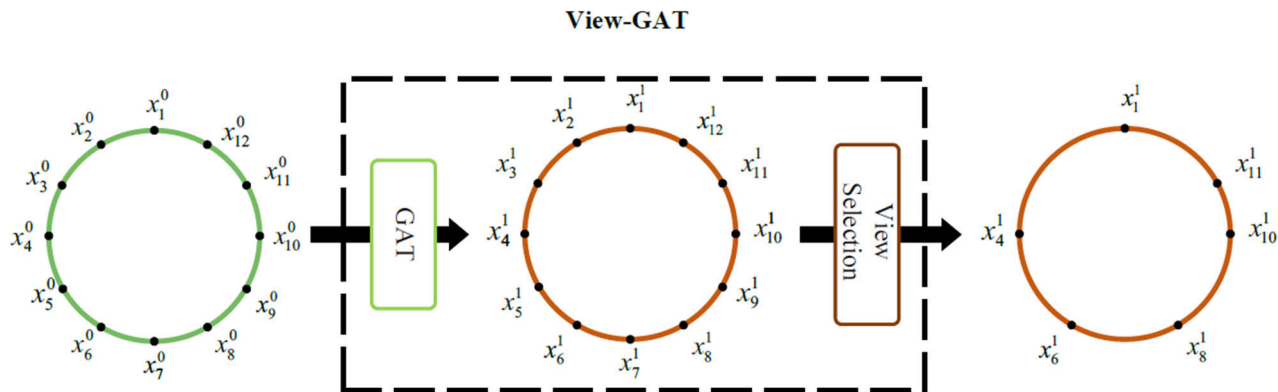


FIGURE 2. The View-GAT based view selection algorithm.

realize the 3D object recognition. In the process of practical application, our proposed method consists of two phase: the training phase and the testing phase. Compared with other methods, the proposed method can make better use of the effective information of multiple views and achieve better recognition performance.

A. VIEW FEATURE EXTRACTION

In the view feature extraction module, each 3D object is first projected to generate multiple projective views. In this paper, 12 projective views are obtained from 12 perspectives. The virtual cameras are placed every 30° on the horizontal plane 30° from the ground and points to the centroid of the 3D object. Then, the multiple projective views are respectively sent to the convolutional neural networks, which are pre-trained with ImageNet. In this paper, according to a series of experiments, we select ResNet-34 to extract view features. The results of comparative experiments are introduced in Section 4. Furthermore, we use multi-view dataset to adjust the parameters of ResNet-34. The parameter adjustment step not only speeds up the subsequent training process, but also improves the recognition accuracy. We take the vector extracted before the fully connection layer of ResNet-34 as the view feature $\{x_i^0\}_{i=1}^N$.

B. VIEW-GAT BASED VIEW SELECTION

The proposed view-GAT based view selection algorithm can be briefly described as follows. Firstly, we build graph G^0 by making view feature $\{x_i^0\}_{i=1}^N$ as nodes, and adjacent projective positional relationships between views are defined as the edges. The graph G^0 is sent to the graph attention network (GAT) to get the updated node feature $\{x_i^1\}_{i=1}^N$, which can make each node contain the feature of the adjacent nodes. Then the node features are filtered by view selection module to remove some redundant views. The flow chart of the view-GAT based view selection algorithm is shown in FIGURE 2. From FIGURE 2, we can see that the algorithm includes two parts: the GAT module and the view selection

module. The detailed processing steps of the two modules are shown in FIGURE 3 and FIGURE 4.

Graph attention network (GAT): As shown in FIGURE 3, the view feature $\{x_i\}_{i=1}^N$ is first linearly transformed using a shared weight matrix $W \in \mathbb{R}^{F' \times F}$, where F is the number of input features in each node and F' is the number of output features in each node. Then they are joined by a single-layer feedforward network whose parameter is $\bar{a} \in \mathbb{R}^{2F'}$. The output is processed by a nonlinear LeakyReLU function. In order to compare the attention coefficients of different nodes, we use the softmax function to normalize them and eventually get the attention coefficient α_{ij} . In order to retain the graph structural information, we only compute the attention coefficient between nodes i and its adjacent node j : $\alpha_{ij}, j \in N_i$. N_i includes the node i itself and its first-order neighborhood. The attention coefficient is defined as follows:

$$\alpha_{ij} = \frac{\exp(\text{LeakyReLU}(\bar{a}^T [W\bar{x}_i || W\bar{x}_j]))}{\sum_{k \in N_i} \exp(\text{LeakyReLU}(\bar{a}^T [W\bar{x}_i || W\bar{x}_k]))} \quad (1)$$

After that, we make a linear combination of the attention coefficients and the corresponding features. Then we get the output feature of each updated node through a nonlinear activation function. In order to make the learned feature more robust and prevent overfitting, we use the multi-head attention method to aggregate multiple independent attention coefficients and get the final node feature. The computation formula is defined as below:

$$\bar{x}'_i = \sigma\left(\frac{1}{K} \sum_{k=1}^K \sum_{j \in N_i} \alpha_{ij}^k W^k \bar{x}_j\right) \quad (2)$$

After the above steps, we can get the view feature $\{x'_i\}_{i=1}^N$ that has been updated by the graph attention network. The updated feature has included the features of adjacent views.

View selection module: The purpose of this module is to select and remove the redundant views. As shown in FIGURE 4, the nodes $\{v_i^1\}_{i=1}^{N/2}$ is initial sampling results

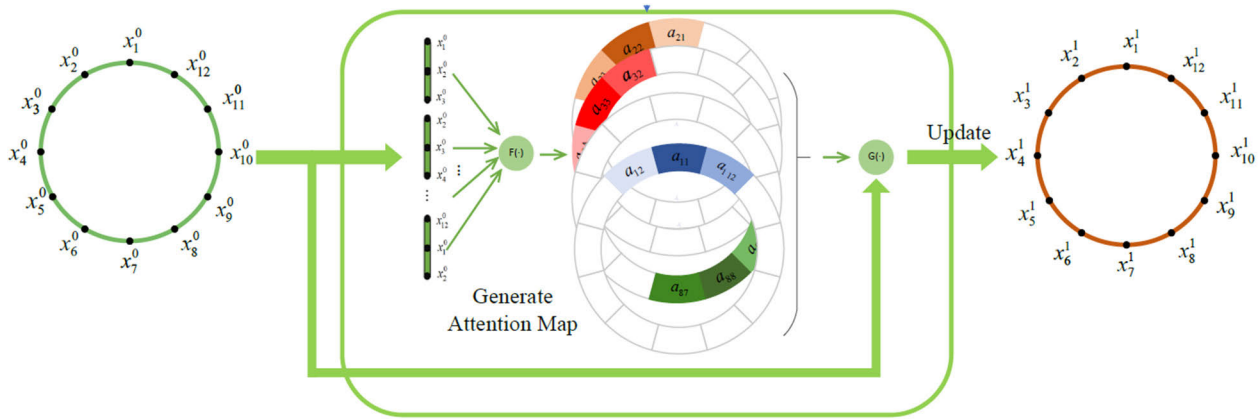


FIGURE 3. The GAT module.

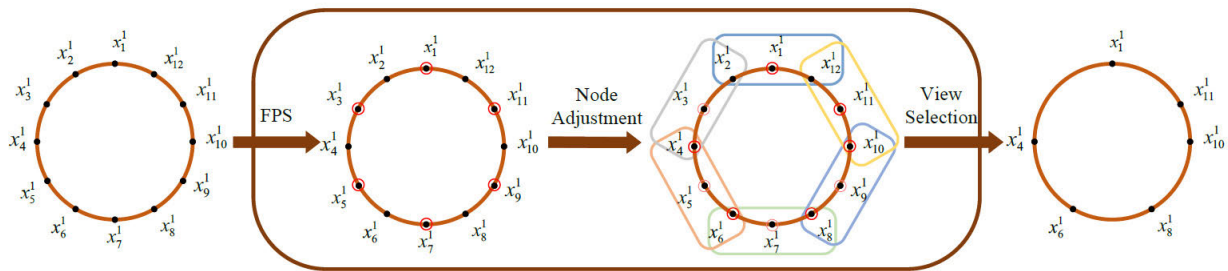


FIGURE 4. The view selection module.

using the farthest point sampling (FPS) method according to the original node position coordinates $\{v_i^0\}_{i=1}^N$. Although the diversity of sampling results can be guaranteed by using the FPS method, the effectiveness of sampled view features cannot be guaranteed for feature recognition. Therefore, it is necessary to adjust these nodes. In this paper, our node adjustment method can be described as follows. Firstly, the sampled node $\{v_i^1\}_{i=1}^{N/2}$ and its adjacent nodes are sent to a two-layer perceptron. The output of the perceptron is the response value about the possibility of which category does the view belong to. Then, the response values of view node and its adjacent view nodes for each class can be obtained. Finally, the node with maximum response value is assigned as the adjusted node. The adjusted node is designed as follows:

$$x_k^1 = \underset{j \in N_i}{\operatorname{argmax}} (\max(V(x_j^1, \theta^1))), \quad k \in N_i \quad (3)$$

where x_k^1 represents the node feature after adjustment, θ^1 represents the parameters of the two-layer perceptron, and N_i represents the node i itself and its adjacent nodes. After view selection, we can obtain the node features $\{x_j^1\}_{j=1}^{N/2}$ and the new graph G^1 , which can be used for the following feature aggregation step.

C. HIERARCHICAL FEATURE AGGREGATION

In this section, we aim to aggregate the extracted multi-view features into a global feature. From the view selection step, we can conclude that the views are not independent from each other but related to each other in terms of position and shape, which can provide more discriminative information for 3D object recognition. So we design the hierarchical feature aggregation to obtain the global feature for recognition, which consists of the correlation weighted feature aggregation module and the hierarchical aggregation scheme.

Correlation Weighted Feature aggregation module:

Because each view has different contribution to the final feature recognition, we can assign different weights to different views to improve the 3D recognition performance. As shown in FIGURE 5, the node features obtained by the GAT view selection module are firstly connected in pairs to concatenate a matrix $M_{N \times N \times 2F}$. Then the matrix is sent to a three-layer perceptron ϕ with LeakyReLU to obtain the similarity matrix $S(i, j)$, where the elements of the matrix $S(i, j)$ indicate the similarity degree of node features between node i and node j . If a view has high similarities with other views, it indicates that the view is more representative and effective. And then, we sum the elements of each row of

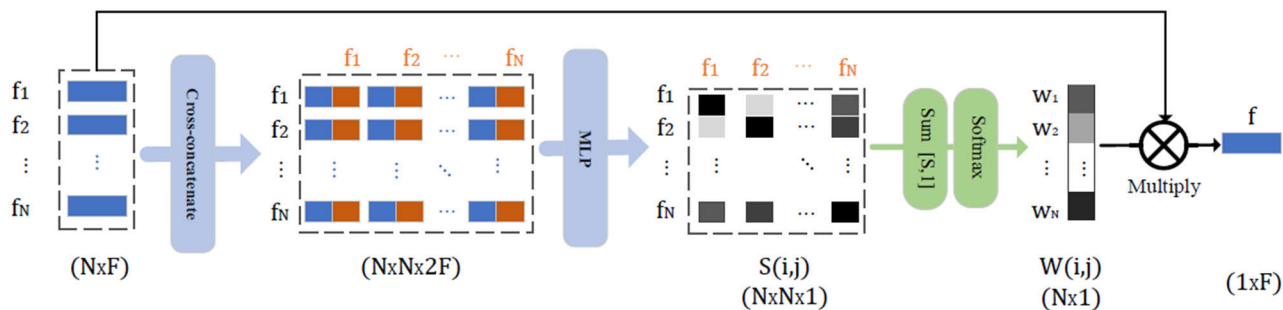


FIGURE 5. The correlation weighted feature aggregation module.

the similarity matrix $S(i, j)$ to obtain $S(i)$, where $S(i)$ is the effectivity score of each view. In order to compare various views, we use the softmax function to normalize $S(i)$ and define it as the weight matrix $W(i)$. The greater the weight is, the higher the representativeness and effectiveness of the view is. Finally, we can obtain the aggregation feature F_{global} by multiplying the weight matrix and the node feature, whose dimension is $1 \times F$.

Hierarchical Aggregation Scheme: As shown in FIGURE 1, we perform the view selection operation and the correlation weighted feature aggregation operation twice to obtain more informative global shape feature. The proposed hierarchical aggregation scheme can be described as follows. After the first view selection and correlation weighted feature aggregation, we can obtain the first aggregated feature F_{global}^1 , as one part of the final global feature. Then for the new graph G^1 obtained by the first view selection operation, we perform the second view selection operation and the second correlation weighted feature aggregation operation, and the second aggregated feature F_{global}^2 can be obtained. Finally, the final global feature F_{global} is computed by concatenating of F_{global}^1 and F_{global}^2 . The final global feature F_{global} consists of two levels of aggregated features, so it contains more discriminative feature for recognition.

For the 3D object recognition task, the final global feature F_{global} is sent to three fully connected layers with LeakyReLU. The dimension of the network output is the number of classes. Finally, we classify the 3D object belongs to the class with the maximum response value.

D. NETWORK TRAINING

The network training has two processes. The first process is training the ResNet-34 network which has been pre-trained by ImageNet, using multi-view dataset. By fine-tuning the network parameters, we can accelerate the subsequent training convergence speed and enhance accuracy of the model. The second process is removing the fully connected layer of the ResNet-34 as the view feature extraction part, and then adding it in the whole algorithm framework to achieve end-to-end learning.

In the step of fine-tuning the network parameters of ReSNet-34, we used the Stochastic Gradient Descent (SGD) algorithm as the optimizer, whose weight attenuation coefficient is set to be 10^{-3} , the momentum is 0.9, the training period is 30, the learning rate is 10^{-2} , the batch size is 400, and the learning rate is reduced by half for every 10 training periods. In the second process of the whole algorithm framework, we also use SGD as the optimizer. The parameter setting is almost the same as that in the first process. The difference is that the learning rate is set to be 10^{-3} . Each batch contained 240 images, which come from 12 projective views of 20 3D objects. The learning rate is set according to the method in [47]. In the first training period, the learning rate is linearly increased from 0 to 10^{-3} . Then in the subsequent periods, the learning rate is changed according to the following formula:

$$lr_t = \frac{1}{2}(1 + \cos(\frac{t\pi}{T}))lr \tag{4}$$

where lr is the initial learning rate.

IV. EXPERIMENTS

In this section, we perform several experiments to validate the classification and retrieval performance of GA-MVCNN on different 3D object datasets. Firstly, we introduce the dataset and experimental setting. Secondly, we give the details on view feature extraction network structure selection. Thirdly, we discuss the experimental results of the proposed GA-MVCNN method and the representative 3D object recognition methods. And then, we give the comparison results of the proposed GA-MVCNN method and the representative methods on 3D object retrieval task. Finally we conduct ablation studies to analysis the influences of different modules.

A. PLATFORM AND DATASET

In this paper, the codes of all experiments are implemented under pytorch framework. The running configurations are as follows: The CPU is Intel Xeon E5-2630 V4. The memory size is 128GB, and the GPU is GTX2080Ti. The operating system is Ubuntu 16.04.3 LST. In order to evaluate the recognition and retrieval performance of our proposed



FIGURE 6. Some models of ModelNet40 dataset.

GA-MVCNN method, we adopt the ModelNet40 dataset [15] and the ShapeNet Core 55 dataset [48] to carry out experiments.

The ModelNet40 dataset is a subset of the Princeton ModelNet Dataset. It contains 12311 3D object models from 40 classes, including 9483 models for training and 2468 models for testing. Some models of ModelNet40 dataset are shown in FIGURE 6. To conduct our experiments, we randomly selected 100 models from each class, 80 models of which are used as training data and 20 models as testing data.

The ShapeNet Core55 dataset contains 51190 3D object models of 55 categories, which are further divided into 204 sub-categories. Some models of ModelNet40 dataset are shown in FIGURE 7. We follow the official splitting separation method to divide the dataset into train, validation and test subsets with the proportion of 70%, 10% and 20% respectively. The data set has two versions: the normal version and the disturbed version, whose all images are randomly rotated. In this paper, we use the normal version to carry out experiments.

B. VIEW FEATURE EXTRACTION NETWORK SELECTION AND CONVERGENCE ANALYSIS

In order to find the optimal network structure for view feature extracting, we testify the proposed 3D object recognition method using different network structures. Here, we respectively use the network structures of AlexNet, VGG-16, ResNet-18, ResNet-34 and ResNet-50 to perform 3D object recognition experiments. From TABLE 1 we can see that the ResNet-34 network structure performs best in term of the Per Class Accuracy and the Per Instance Accuracy. Furthermore, the training time is within the acceptable range. Therefore,

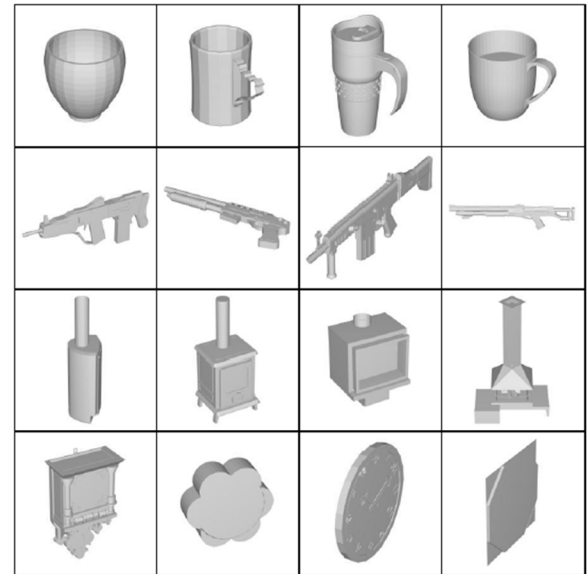


FIGURE 7. Some models of ShapeNet Core55 dataset.

TABLE 1. The algorithm performance under different network structures.

Network	Per Class Acc. (%)	Per Ins Acc. (%)	Time of 30 epochs
AlexNet	89.1	92.0	1h 45min
VGG-16	91.2	93.6	8h 54min
ResNet-18	94.0	95.6	2h 16min
ResNet-34	94.3	96.2	3h 29min
ResNet-50	92.4	95.0	6h 15min

the ResNet-34 network is adopted as the base network of our proposed GA-MVCNN method.

FIGURE 8 presents the changes of the Per Class Accuracy with respect to training epochs, and FIGURE 9 presents the changes of the Per Class Accuracy with respect to training epochs. From FIGURE 8 and FIGURE 9, we can conclude that no matter which network structure is used, the proposed GA-MVCNN method is convergent. The ResNet-34 network based GA-MVCNN method converges better than other base network based GA-MVCNN methods.

C. CLASSIFICATION EXPERIMENTS ON MODELNET40

In this section, the proposed GA-MVCNN base 3D object recognition method is used to perform 3D object classification experiments on ModelNet40 dataset. Then, we compared our proposed method with other state-of-the-art methods, including the voxel based methods, the point cloud based methods, the multi-view based methods and the panoramic-view based methods. In order to evaluate the robustness of the proposed methods, we randomly selected the training and testing models 10 times to obtain the same 10 cross-validation splits. We average 10 classification accuracies as the final results. TABLE 2 presents the quantitative experimental results on ModelNet40. We can see that the performances of

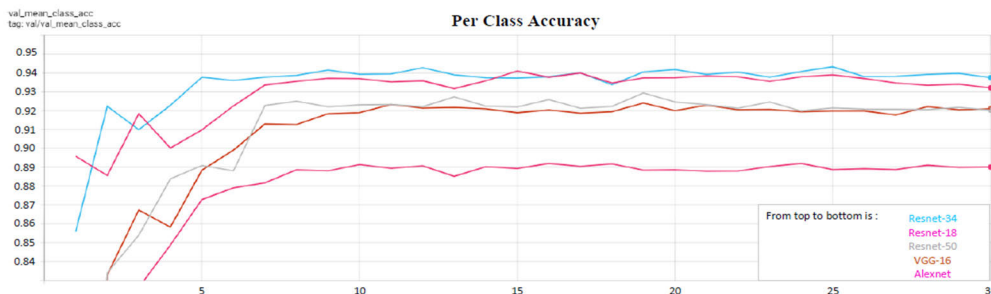


FIGURE 8. The Per Class Accuracy vs training epoches.

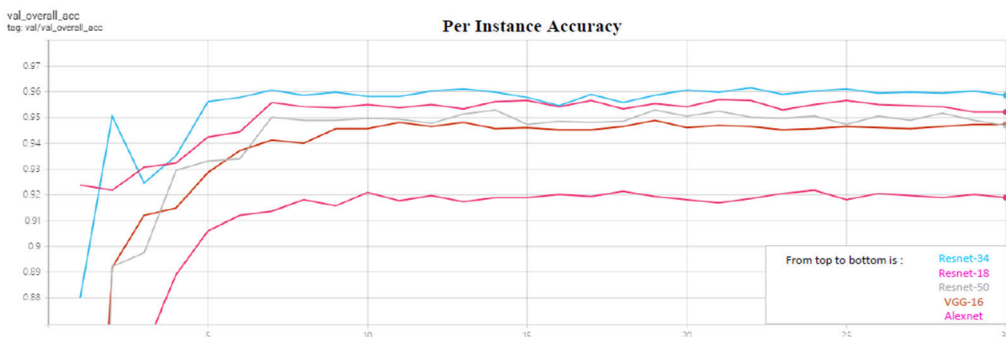


FIGURE 9. The Per Instance Accuracy vs training epoches.

TABLE 2. The classification accuracy of each method on ModelNet40 data set.

Methods	Modality	Per Class Acc. (%)	Per Ins Acc. (%)
3DShapeNets [15]	Voxel	77.3	-
VoxNet [16]	Voxel	83.0	-
VRN Ensemble [49]	Voxel	-	95.5
NormalNet [50]	Voxel	-	88.8
MVCNN-MultiRes [26]	Voxel	91.4	93.8
PointNet++ [21]	Point cloud	-	91.9
Kd-Networks [51]	Point cloud	88.5	91.8
MVCNN [4]	Multi-view	90.1	90.1
MVCNN-new [7]	Multi-view	92.4	95.0
MLVCNN [29]	Multi-view	-	94.2
GVCNN [25]	Multi-view	90.7	93.1
Seqviews2seqlabels [11]	Multi-view	91.1	93.3
SPNet [52]	Multi-view	-	88.6
SPNet_VE [52]	Multi-view	-	92.6
MHBN [53]	Multi-view	93.1	94.7
PANORAMA-NN [54]	Panoramic-view	-	90.7
PANORAMA-NN + GradM [55]	Panoramic-view	-	92.0
PANORAMA-ENN [55]	Panoramic-view	-	95.6
GA-MVCNN (ours)	Multi-view	94.3	96.2

the multi-view based methods are higher than other kinds of methods, and our proposed method outperform other methods in both of the two criteria. The Per Class Accuracy is 1.2% higher than that of the second-highest MHBN [53] method,

and the Per Instance Accuracy is 1.2% higher than that of the second-highest MVCNN-new [7] method. The improvements justify the effectiveness of our proposed GA-MVCNN base 3D object recognition method. Furthermore, the standard deviation of the Per Class Accuracy is $\pm 1.3\%$, and the standard deviation of the Per Instance Accuracy is $\pm 2.1\%$. From experimental results, we can conclude that the proposed method have good reliability.

D. RETRIEVAL EXPERIMENTS ON SHAPENET CORE55

In this section, we evaluate the retrieval performance of our proposed method on ShapeNet Core55 dataset. In the training stage, we first generate multiple views of the 3D object. Then we train our proposed network for classification task. The output vector in the front of the final fully connection layer of the network is used as the 3D object feature in the following retrieval experiments. For the testing sample, we first compute its corresponding global feature using the trained network. Then, we use the L2 distance measure to obtain the retrieval results. Given a 3D object to be queried, the first 1000 retrieved 3D objects are taken as the retrieval result. In the experiments, we use three popular criteria: F-score, mAP and normalized discounted cumulative gain (NDCG) with testing range N equaling to 1000. We calculate micro-averaged values to describe the influence of different classes sizes and macro-averaged values for the entire database.
























































ID	Query	Top10 Retrieved Results									
2691156											
3624134											
2880940											
3636649											
4379243											

FIGURE 10. Randomly select the top 10 retrieval results of 5 categories for visualization.

TABLE 3. Comparison of retrieval performance with different methods on the normalized ShapeNet Core 55 data set.

Method	microALL			macroALL		
	F-score	mAP	NDCG	F-score	mAP	NDCG
ZFDR [54]	28.2	19.9	33.0	19.7	25.5	37.7
DeepVoxNet [55]	25.3	19.2	27.7	25.8	23.2	33.7
DLAN [56]	71.2	66.3	76.2	50.5	47.7	56.3
RotationNet [27]	79.8	77.2	86.5	59.0	58.3	65.6
Improved GIFT [57]	76.7	72.2	82.7	58.1	57.5	65.7
ReVGG [55]	77.2	74.9	82.8	51.9	49.6	55.9
MVFusionNet [55]	69.2	62.2	73.2	48.4	41.8	50.2
CM-VGG5-6DB [55]	47.9	54.0	65.4	16.6	33.9	40.4
GIFT [58]	68.9	64.0	76.5	45.4	44.7	54.8
MVCNN [4]	76.4	73.5	81.5	57.5	56.6	64.0
PANORAMA-ENN [59]	78.9	73.9	84.5	59.1	58.8	65.6
PANORAMA-NN [60]	77.6	72.3	81.5	58.0	55.7	63.0
SPNet_VE [52]	78.9	69.2	89.0	53.5	39.2	69.5
GA-MVCNN (ours)	80.3	77.4	85.7	60.5	58.9	66.7

We compared our proposed GA-MVCNN based 3D object recognition method with the ZFDR method [54], the DeepVoxNet method [55], the DLAN method [56], the RotationNet method [27], the Improved GIFT method [57], the ReVGG method [55], the MVFusionNet method [55], the CM-VGG5-6DB method [55], the GIFT method [58], the MVCNN method [4], the PANORAMA-ENN method [59], the PANORAMA-NN method [60] and the SPNet_VE method [52]. TABLE 3 presents the comparative experimental results. On the micro setting, our proposed method outperforms all the comparative methods by 0.5%-52.1%, 0.2%-58.2% in terms of F-score and mAP. In terms of NDCG criteria, our proposed method is lower than the RotationNet method and the SPNet_VE method, but it is higher than other comparative methods. On the macro setting, our proposed method outperforms all the comparative methods

by 1.4%-43.9%, 0.1-35.7% in terms of F-score and mAP. In terms of NDCG criteria, our proposed method is only lower than the SPNet_VE method. From the above analysis, we can prove the effectiveness of our proposed GA-MVCNN method.

Furthermore, some retrieval results that include the given query and the top 10 retrieved 3D models; mistakes are highlighted in red. From FIGURE 10, we can see that the retrieval results of the Bowl class with ID 2880940 and the Table class with ID 3991062 is wrongly retrieved have wrong results, while the retrieval results of the other four classes are correct. Analyzing the poor results of the Bowl class and the Table class, we can find that the projective views of this class are similar. So it is very important to select a good view projection strategy for the multi-view based 3D object analysis methods.

TABLE 4. The ablation studies.

View Number	Method	Per Class Acc. (%)	Per Ins Acc. (%)
12	Benchmark	92.7	95.1
12	Remove GAT module	93.5	95.5
12	Remove view selection module	93.9	95.8
12	Remove feature aggregation module	93.6	95.7
12	Remove the second GAT based feature aggregation module	93.2	95.8
12	Original	94.3	96.2
20	Original	95.5	96.8

E. ABLATION STUDIES

In this part, we perform ablation studies to test the influence of different modules in our proposed GA-MVCNN method. We use the ResNet-34 based experimental results as the benchmark, and the ablation studies includes the validity of the module, the effectiveness of the hierarchy and the influence of the view projection strategy. To evaluate the contributions of each module, we remove the graph attention network, the view selection module and the feature aggregation module respectively from the GA-MVCNN framework. The effectiveness of the hierarchical structure is studied by ablating the second layer of GAT based feature aggregation structure and comparing the corresponding experimental results with the complete algorithm. To investigate the influence of projection strategy, we adopt the regular dodecahedron projection structure, where each vertex is a projection position to generate 20 views for each 3D object. Then we compared the 20-view based GA-MVCNN method with the above 12-view based GA-MVCNN method. The experimental results are shown in TABLE 4.

Compared with the benchmark method, our complete 12-view based GA-MVCNN respectively improved by 1.6% and 1.1% in terms of the Per Class AccuracyAcc. and the Per Instance AccuracyPer Ins Acc. After removing the GAT module, the Per Class Accuracy. and the Per Instance Accuracy Per Ins Acc. decreased by 0.8% and 0.7% respectively. By removing the view selection module, both the Per Class Accuracy and the Per Instance Accuracy dropped by 0.4%. The removing of the correlation weighted feature aggregation module led to 0.7% and 0.5% drops respectively on the Per Class Accuracy and the Per Instance Accuracy, which proved the effectiveness of the module. The above experimental results show that each module has its own contribution to the final GA-MVCNN method. After removing the second layer View-GAT, the Per Class Accuracy and the Per Instance Accuracy decreased by 0.9% and 0.5% respectively, which verified the effectiveness of the hierarchy structure. After we increased the number of views from 12 to 20, the Per Class Accuracy and the Per Instance Accuracy increased by 1.2% and 0.6% respectively. It indicates that a more complicated 3D projection strategy would improve the final performance

because the projective views may have more discriminative information of 3D objects.

V. CONCLUSION

This paper mainly presents a novel hierarchical graph attention based multi-view convolutional neural network for 3D object recognition. It uses the view selection module to reduce redundant view information in multiple projective views, which can retain the projective views with abundant effective information and discard the projective views with less effective information. Then, we use the correlation weighted feature aggregation module to enhance the efficiency of information utilization, which can assign different weights to multiple views by measuring the information of them. Furthermore, we use the hierarchical feature aggregation network structure to make full use of the correlation information of multiple views. To evaluate the effectiveness of the proposed method, we designed the classification and retrieval experiments respectively on datasets ModelNet40 and ShapeNetCore55. Compared with other methods, our proposed method performs better. Through ablation experiments, the effectiveness of the GAT based view selection module, the correlation weighted feature aggregation module and the hierarchy feature aggregation framework have been validated.

REFERENCES

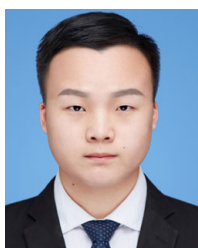
- [1] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun, "Deep learning for 3D point clouds: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jun. 29, 2020, doi: [10.1109/TPAMI.2020.3005434](https://doi.org/10.1109/TPAMI.2020.3005434).
- [2] W. Nie, Y. Zhao, A.-A. Liu, Z. Gao, and Y. Su, "Multi-graph convolutional network for unsupervised 3D shape retrieval," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 3395–3403.
- [3] W.-Z. Nie, A.-A. Liu, S. Zhao, and Y. Gao, "Deep correlated joint network for 2-D image-based 3-D model retrieval," *IEEE Trans. Cybern.*, early access, Jun. 30, 2020, doi: [10.1109/TCYB.2020.2995415](https://doi.org/10.1109/TCYB.2020.2995415).
- [4] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3D shape recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 945–953.
- [5] W. Nie, W. Jia, W. Li, A. Liu, and S. Zhao, "3D pose estimation based on reinforce learning for 2D image-based 3D model retrieval," *IEEE Trans. Multimedia*, early access, Apr. 30, 2020, doi: [10.1109/TMM.2020.2991532](https://doi.org/10.1109/TMM.2020.2991532).
- [6] W. Nie, Q. Liang, A.-A. Liu, Z. Mao, and Y. Li, "MMJN: Multi-modal joint networks for 3D shape recognition," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 908–916.
- [7] J. C. Su, M. Gadelha, R. Wang, and S. Maji, "A deeper look at 3D shape classifiers," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 645–661.
- [8] W. Nie, M. Ren, A. Liu, Z. Mao, and J. Nie, "M-GCN: Multi-branch graph convolution network for 2D image-based on 3D model retrieval," *IEEE Trans. Multimedia*, early access, Jul. 3, 2020, doi: [10.1109/TMM.2020.3006371](https://doi.org/10.1109/TMM.2020.3006371).
- [9] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [10] S. Chen, L. Zheng, Y. Zhang, Z. Sun, and K. Xu, "VERAM: View-enhanced recurrent attention model for 3D shape classification," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 12, pp. 3244–3257, Dec. 2019.
- [11] Z. Han, M. Shang, Z. Liu, C.-M. Vong, Y.-S. Liu, M. Zwicker, J. Han, and C. L. P. Chen, "SeqViews2SeqLabels: Learning 3D global features via aggregating sequential views by RNN with attention," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 658–672, Feb. 2019.
- [12] C. Ma, Y. Guo, J. Yang, and W. An, "Learning multi-view representation with LSTM for 3-D shape recognition and retrieval," *IEEE Trans. Multimedia*, vol. 21, no. 5, pp. 1169–1182, May 2019.

- [13] Y. Xie, Y. Zhang, M. Gong, Z. Tang, and C. Han, "MGAT: Multi-view graph attention networks," *Neural Netw.*, vol. 132, no. 12, pp. 180–189, Dec. 2020.
- [14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.
- [15] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3D ShapeNets: A deep representation for volumetric shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1912–1920.
- [16] D. Maturana and S. Scherer, "VoxNet: A 3D convolutional neural network for real-time object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Hamburg, Germany, Sep. 2015, pp. 922–928.
- [17] C. B. Choy, D. Xu, J. Gwak, K. Chen, and S. Savarese, "3D-R2N2: A unified approach for single and multi-view 3D object reconstruction," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 628–644.
- [18] N. Sedaghat, M. Zolfaghari, E. Amiri, and T. Brox, "Orientation-boosted voxel nets for 3D object recognition," Oct. 2016, *arXiv:1604.03351*. [Online]. Available: <http://arxiv.org/abs/1604.03351>
- [19] Y. Li, S. Pirk, H. Su, C. R. Qi, and L. J. Guibas, "FPNN: Field probing neural networks for 3D data," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 307–315.
- [20] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 77–85.
- [21] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5099–5108.
- [22] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Trans. Graph.*, vol. 38, no. 5, pp. 1–12, Nov. 2019.
- [23] H. Su, V. Jampani, D. Sun, S. Maji, E. Kalogerakis, M.-H. Yang, and J. Kautz, "SPLATNet: Sparse lattice networks for point cloud processing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 2530–2539.
- [24] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," Nov. 2014, *arXiv:1405.3531*. [Online]. Available: <http://arxiv.org/abs/1405.3531>
- [25] Y. Feng, Z. Zhang, X. Zhao, R. Ji, and Y. Gao, "GVCNN: Group-view convolutional neural networks for 3D shape recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 264–272.
- [26] C. R. Qi, H. Su, M. NieBner, A. Dai, M. Yan, and L. J. Guibas, "Volumetric and multi-view CNNs for object classification on 3D data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 5648–5656.
- [27] A. Kanezaki, Y. Matsushita, and Y. Nishida, "RotationNet: Joint object categorization and pose estimation using multiviews from unsupervised viewpoints," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 5010–5019.
- [28] X. He, Y. Zhou, Z. Zhou, S. Bai, and X. Bai, "Triplet-center loss for multi-view 3D object retrieval," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 1945–1954.
- [29] J. Jiang, D. Bao, Z. Chen, X. Zhao, and Y. Gao, "MLVCNN: Multi-loop-view convolutional neural network for 3D shape retrieval," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 8513–8520.
- [30] X. Wei, R. Yu, and J. Sun, "View-GCN: View-based graph convolutional network for 3D shape analysis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 1847–1856.
- [31] Z. Xue, P. Du, J. Li, and H. Su, "Simultaneous sparse graph embedding for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 11, pp. 6114–6133, Nov. 2015.
- [32] G. Shi, H. Huang, and L. Wang, "Unsupervised dimensionality reduction for hyperspectral imagery via local geometric structure feature learning," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 8, pp. 1425–1429, Aug. 2020.
- [33] F. Luo, B. Du, L. Zhang, L. Zhang, and D. Tao, "Feature learning using spatial-spectral hypergraph discriminant analysis for hyperspectral image," *IEEE Trans. Cybern.*, vol. 49, no. 7, pp. 2406–2419, Jul. 2019.
- [34] P. Frasconi, M. Gori, and A. Sperduti, "A general framework for adaptive processing of data structures," *IEEE Trans. Neural Netw.*, vol. 9, no. 5, pp. 768–786, Sep. 1998.
- [35] A. Sperduti and A. Starita, "Supervised neural networks for the classification of structures," *IEEE Trans. Neural Netw.*, vol. 8, no. 3, pp. 714–735, May 1997.
- [36] M. Gori, G. Monfardini, and F. Scarselli, "A new model for learning in graph domains," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, Montreal, QC, Canada, Jul./Aug. 2005, pp. 729–734.
- [37] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61–80, Jan. 2009.
- [38] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," May 2013, *arXiv:1312.6203*. [Online]. Available: <http://arxiv.org/abs/1312.6203>
- [39] M. Henaff, J. Bruna, and Y. LeCun, "Deep convolutional networks on graph-structured data," Jun. 2015, *arXiv:1506.05163*. [Online]. Available: <http://arxiv.org/abs/1506.05163>
- [40] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3844–3852.
- [41] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," Feb. 2016, *arXiv:1609.02907*. [Online]. Available: <http://arxiv.org/abs/1609.02907>
- [42] J. Atwood and D. Towsley, "Diffusion-convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 1993–2001.
- [43] M. Niepert, M. Ahmed, and K. Kutzkov, "Learning convolutional neural networks for graphs," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 2014–2023.
- [44] F. Monti, D. Boscaini, J. Masci, E. Rodola, J. Svoboda, and M. M. Bronstein, "Geometric deep learning on graphs and manifolds using mixture model CNNs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 5425–5434.
- [45] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1024–1034.
- [46] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–12.
- [47] T. He, Z. Zhang, H. Zhang, Z. Zhang, J. Xie, and M. Li, "Bag of tricks for image classification with convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 558–567.
- [48] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An information-rich 3D model repository," 2015, *arXiv:1512.03012*. [Online]. Available: <http://arxiv.org/abs/1512.03012>
- [49] A. Brock, T. Lim, J. M. Ritchie, and N. Weston, "Generative and discriminative voxel modeling with convolutional neural networks," Aug. 2016, *arXiv:1608.04236*. [Online]. Available: <http://arxiv.org/abs/1608.04236>
- [50] C. Wang, M. Cheng, F. Sohel, M. Bannamoun, and J. Li, "NormalNet: A voxel-based CNN for 3D object classification and retrieval," *Neurocomputing*, vol. 323, pp. 139–147, Jan. 2019.
- [51] R. Klokov and V. Lempitsky, "Escape from cells: Deep Kd-networks for the recognition of 3D point cloud models," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 863–872.
- [52] M. Yavartanoo, E. Y. Kim, and K. M. Lee, "SPNet: Deep 3D object classification and retrieval using stereographic projection," in *Proc. Asian Conf. Comput. Vis.*, 2018, pp. 691–706.
- [53] T. Yu, J. Meng, and J. Yuan, "Multi-view harmonized bilinear network for 3D object recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 186–194.
- [54] B. Li and H. Johan, "3D model retrieval using hybrid features and class information," *Multimedia Tools Appl.*, vol. 62, no. 3, pp. 821–846, Feb. 2013.
- [55] M. Savva et al., "SHREC'17 track: Large-scale 3D shape retrieval from ShapeNet Core55," in *Proc. Eurograph. Workshop 3D Object Retr.*, 2017, pp. 1–12.
- [56] T. Furuya and R. Ohbuchi, "Deep aggregation of local 3D geometric features for 3D model retrieval," in *Proc. Eurograph. Workshop 3D Object Retr.*, 2016, pp. 121.1–121.12.
- [57] S. Bai, X. Bai, Z. Zhou, Z. Zhang, Q. Tian, and L. J. Latecki, "GIFT: Towards scalable 3D shape retrieval," *IEEE Trans. Multimedia*, vol. 19, no. 6, pp. 1257–1271, Jun. 2017.
- [58] S. Bai, X. Bai, Z. Zhou, Z. Zhang, and L. J. Latecki, "GIFT: A real-time and scalable 3D shape search engine," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 5023–5032.

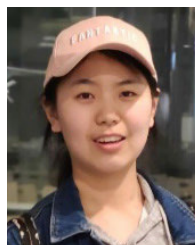
- [59] K. Sfikas, T. Theoharis, and I. Pratikakis, "Exploiting the PANORAMA representation for convolutional neural network classification and retrieval," in *Proc. Eurograph. Workshop 3D Object Retr.*, 2017, pp. 1–7.
- [60] K. Sfikas, I. Pratikakis, and T. Theoharis, "Ensemble of PANORAMA-based convolutional neural networks for 3D model classification and retrieval," *Comput. Graph.*, vol. 71, pp. 208–218, Apr. 2018.



HUI ZENG received the B.S. and M.S. degrees from Shandong University, in 2001 and 2004, respectively, and the Ph.D. degree from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, in 2007. She is currently a Professor with the School of Automation and Electrical Engineering, University of Science and Technology Beijing, China. Her main research interests include computer vision, pattern recognition, and machine learning.



TIANMENG ZHAO received the B.S. degree from the University of Science and Technology Beijing, in 2020, where he is currently pursuing the master's degree. His main research interests include computer vision, deep learning, and point cloud processing.



RUTING CHENG received the B.S. degree from the University of Science and Technology Beijing, in 2019, where she is currently pursuing the master's degree. Her main research interests include computer vision, deep learning, and point cloud processing.



FUZHOU WANG received the B.S. degree from the University of Science and Technology Beijing in 2018, where he is currently pursuing the master's degree. His main research interests include computer vision, deep learning, and point cloud processing.



JIWEI LIU received the B.Sc. degree from the University of Science and Technology of China and the M.Sc. degree from the University of Science and Technology Beijing. He is currently an Associate Professor with the University of Science and Technology Beijing. His main research interests include image processing and pattern recognition.

...