Hindawi
Mobile Information Systems
Volume 2022, Article ID 4064512, 11 pages
https://doi.org/10.1155/2022/4064512

Research Article

Simplified Algorithm of Geometric Model Region Segmentation Using Neural Network

Yuan Wang, Wei Guo, Shuanfeng Zhao, Zhengxiong Lu, and Zhizhong Xing

College of Mechanical Engineering, Xi'an University of Science and Technology, Xi'an 710054, Shaanxi, China

Correspondence should be addressed to Yuan Wang; wangy1212@xust.edu.cn

Received 24 June 2022; Revised 27 July 2022; Accepted 1 August 2022; Published 28 August 2022

Academic Editor: Muhammad Zakarya

The simplification of three-dimensional (3D) models has always been a hot research topic for scholars. The researchers simplified different parts of the 3D point cloud data from both global and local information. Aiming at the need to retain detailed features in the simplification of 3D models, the neural network (NN) technology is firstly analyzed and studied, and a simplified algorithm for regional segmentation of geometric models based on Graph Convolutional Neural Network (GCNN) is proposed. Secondly, based on the idea of dense connection of DenseNet network structure, a symmetric segmentation model is established. The left part continuously performs Down-Sampling and local feature aggregation on the original geometric model through the Weighted Critical Points (WCPL) algorithm and edge convolution operation and performs compression encoding. At the same time, the right part uses the interpolation method for Up-Sampling the encoded data to increase the number of data points and feature dimensions, so as to restore the point cloud data to the dimensions before processing. Finally, it is restored to the dimension size of the original data to realize the end-to-end output of the segmentation model. Comparing the results with other segmentation models, it shows that (1) as the number of iterations increases, the regional accuracy of the training set increases; (2) after 1000 training rounds, from the perspective of the segmentation effect of a single category of objects, the model has good segmentation effect and has application prospects; and (3) compared with other models, the segmentation interaction ratio of the model is at a relatively mature level. The findings can provide a reference for the application of the segmentation technology of related geometric models and neural networks in the fields of similar models and image segmentation.

1. Introduction

In recent years, the three-dimensional (3D) image of the 3D model clearly describes the actual object due to the rapid progress of computer equipment and virtual reality technology. Therefore, 3D models are gradually widely used in scientific research and people's lives, such as in reverse engineering, 3D animation, professional medical field, design of the 3D game, and other fields [1]. The number of present 3D models is growing exponentially, and massive 3D models are created and published every day, which leads to a large amount of memory space for the number of these models [2]. Hence, separating the model's "subcomponents" from the overall model to improve its reusability has become an important topic for researchers [3]. The data attributes of the 3D geometric model are completely different from those of audio, image, and video. Sound can be regarded as a one-dimensional linear function of time. The image can be regarded as a two-dimensional (2D) plane function in the left, right, up, and down directions. Video can be taken as a function of 3D space about time and space, in which time is one-dimensional and space is 2D. They are all sampled orderly and regularly in Euclidean space, and then converted into binary data [4].

The commonly used model in the field of graph segmentation is the supervised neural network model based on the information diffusion mechanism. Its principle is that different graph nodes exchange information through interconnected edges, and update their respective states until they reach overall stability. The state of each node is synthesized as the final output of the network model, and the corresponding learning algorithm is introduced to estimate the model parameters [5]. At present, Graph Neural Networks (GNN) in neural networks are closely related to the segmentation of geometric models. However, due to the irregularity of 3D geometric model data, traditional convolutional neural networks (CNN) cannot directly process the data of geometric models [6]. Graph Convolutional Neural Network (GCNN) has received more and more attention recently, especially in the modeling of non-Euclidean domain data; some GCNN-based irregular data modeling methods have also achieved very mature results. Therefore, the geometric model data modeling based on GCNN is an effective and applicable method [7].

Gori et al. (2020) proposed a GNN model based on spectral methods. The model is guided by a solid mathematical theoretical foundation. It uses the overall structural information of graph data and processing methods such as Fourier transformation of graph signals. The state of the internal nodes of the graph data is calculated and updated recursively, and the expression output of the overall graph structure data is obtained. However, the spectral method-based GNN model has many limitations. First, when it processes graph data, the entire graph structure needs to be loaded into the memory for calculation at the same time, and cannot be calculated in batches. When the structure is very large, this method will be difficult to carry out; second, it can only calculate and process undirected graphs and graphs whose edges do not contain attribute features but cannot process directed graphs and other vector diagram structures. As a result, later researchers also proposed other structural forms of GNN models [8]. Scarselli et al. (2021) put forward a GNN model based on the spatiotemporal domain. The so-called GNN network structure takes a single graph node as the basic computing unit instead of the entire graph structure. The node contains the feature information of the graph data, and the edge denotes the dependencies between each node. The feature representation of each node is aggregated from the features of the node itself and the features of its adjacent nodes [9].

Looking at the relevant literature, it can be known that the previous research literature has proposed the relevant GCNN model and used the model to segment the image. However, there are few studies on 3D graphics segmentation, and most of the existing 3D segmentation models study single-image segmentation. Hence, on the basis of GCNN, it proposes a 3D geometric model segmentation algorithm based on GCN, uses the characteristics of GCNN to implement a segmentation model, and compares the segmentation effect and accuracy of the model to provide a reference for related geometric model segmentation research.

Firstly, the neural network technology is analyzed, and the specific network technology to be used in the research, namely, GNN technology is determined. Secondly, the GNN technology and GNN structure are further discussed, and the WCPL Down-Sampling algorithm is studied. Finally, a GCNN ensemble segmentation model is comprehensively proposed, and the designed GCNN segmentation model is tested using the ShapeNet parts dataset.

2. Methods

2.1. Analysis of Neural Network Technology. The Artificial Neural Network (ANN) is an intelligent model that classifies and processes information according to the structural characteristics of animal neural networks [10]. It has a complex structure, so it can process data in different ways according to the changes of internal nodes [11]. The core part of the biological neural network (BNN) is the human brain neural network, which is the basis of ANN. The research content of the human brain neural network is mainly the structure, application, and function of the human brain network [12].

ANN is the simplified technical model of BNN. Its main task is to rely on the theoretical basis of the human brain neural network, select the appropriate ANN model according to the actual needs, design the corresponding neural network algorithm, and simulate some intelligent activities of the human brain to achieve the target expectation and solve the problem [13]. Therefore, BNN mainly studies the mechanism of intelligence. ANN is the realization of intelligent mechanisms, and the two complement each other. Figure 1 shows the structure of neural network technology.

Figure 1 indicates that the structure of a neural network mainly includes three layers, and each layer is interconnected. Green is the input layer, blue is the output layer, and orange is the hidden layer. When constructing the structure of a neural network, the number of nodes in the input and output layers needs to be fixed, while the number of neural units in the hidden layer is not fixed, which can be any value. The meaning of the connected arrows in Figure 1 is the flow direction of data in the neural network operation. The data flow direction during operation is different from that during the test. Each arrow represents a diverse weight value. When calculating, the weight needs to be calculated according to the training situation. In addition to the structure expressed from left to right, another common expression is to represent a neural network from bottom to top. At this time, the input layer is at the bottom of the graph and the output layer is at the top [14].

The research of neural networks can be divided into two aspects: theoretical and applied research [15]. Theoretical research includes the following two categories. (1) It uses neurophysiology and human cognition to analyze human thinking and intelligence mechanisms. (2) It is to use the theoretical knowledge and existing theoretical literature on neural networks, and rely on mathematical methods to explore the neural network model with more complete function and better performance. The performance algorithms, such as network stability, convergence, fault tolerance, and robustness, as well as mathematical network theories, such as neural network dynamics and nonlinear neural fields, are further studied. Applied research can be divided into the following two categories [16]. (1) It is the research of software simulation and hardware implementation of neural networks. (2) It is the research on neural network application in various fields, such as pattern recognition, signal processing, knowledge engineering, expert

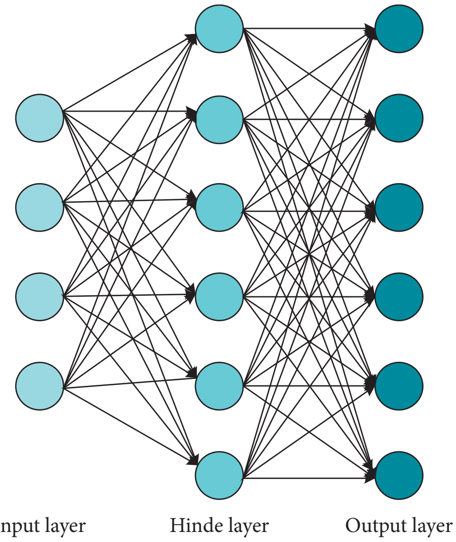

Figure 1: Structure of neural network.

system, optimal combination, and robot control. The application of neural networks will be more in-depth with the continuous progress of related theory itself and technologies. Figure 2 displays the detailed classification of the neural network model.

Figure 2 is to classify the neural network model from different perspectives of the internal structure of the neural network, which can be divided into the feedforward and feedback networks. Among them, the feedforward network is split into a single-layer feedforward network, a multi-layer feedforward network, and a linear neural network. Multi-layer neural networks include radial basis function neural networks, Back Propagation neural networks (BPNN), fully connected neural networks, and CNN. The application of single-layer feedforward neural network is mainly a single-layer perceptron, and the application of a linear neural network is mainly Madaline neural network. Feedback networks are divided into Recursive Neural Networks, Hopfield neural networks, and the brain-state-in-a-box model [17].

2.2. Analysis of GNN. When analyzing the data of the 3D geometric model, the traditional machine learning algorithm generally divides the model into simple parts and processes it through the segmentation of the geometric model. However, this processing method will destroy the overall structure relationship of 3D geometric model data, resulting in the lack of model structure data. These missing data will also lead to the loss of structural features of the geometric model. Therefore, the research on algorithms that can retain model data to the greatest extent is the focus in the field of data modeling and segmentation [18]. In 2009, scholars integrated the advantages of several previous neural network models and proposed a more practical GNN model [19]. The GNN model is based on the supervised neural network model of an information dispersion mechanism, and carries out data transmission and exchange through the connection edge between internal nodes to continuously update the internal state and maintain the stability of the structure. The output value of the model is obtained by calculating the state of each node, and the GNN model's parameter value is calculated according to the corresponding learning algorithm [20, 21]. GNN model regards the nodes connected inside the structure as a learning goal, and the connected edges between nodes represent the dependencies between nodes. Consequently, two nodes are connected, including the data characteristics expressed by each node and the common data characteristics between nodes. It means that the state vector x_v of each node is composed of its own state information and the state information of its adjacent nodes, that is,

$$\begin{aligned} x_v &= f_w\left(l_v, l_{co[v]}, l_{nc[v]}, x_{nc[v]}\right), \\ o_v &= g_w\left(x_v, l_v\right), \end{aligned} \tag{1}$$

x_v means the node state vector; o_v shows the output value of the node; and f_w refers to the local transformation function. g_w stands for a local output function, which describes the generation process of output value. l_v denotes label information.

According to GNN model theory, the 3D geometric model includes two parts of data. One is node information and the other is edge information. For nodes, each node can be regarded as a learning goal. This state is composed of the data characteristics represented by adjacent nodes and connecting edges and the common data characteristics between nodes. Moreover, it is learned through a corresponding learning algorithm. Hence, the most important content of GNN model research is to calculate the state of internal nodes. How to select and train an appropriate local transformation function f_w and local output function g_w is the key to the model establishment [22]. However, the nodes in the 3D geometric model are connected with each other to form a circular and discrete structure. Consequently, GNN uses iterative theory to calculate the state of nodes in the graph. Through continuous iteration, the attribute characteristics of nodes in the graph are recursively propagated until convergence. Finally, it can reach the equilibrium state of the whole graph. Therefore, the status and output of the node can be represented as follows:

$$\begin{aligned} x_v(t+1) &= f_w\left(l_v, l_{co[v]}, l_{nc[v]}, x_{nc[v]}(t)\right), \\ o_v(t+1) &= g_w\left(x_v(t), l_v\right), \end{aligned} \tag{2}$$

$x_v(t)$ represents the state vector of the t-th iteration of the graph node. Iterative state vector $x_v(t+1)$ of round $t+1$ is determined by the state vector and attribute label of the t-th round of the node, as well as the information of adjacent nodes and connecting edges. f_w and g_w represent the transformation function.

2.3. Structural Analysis of GCNN. CNN is the basis of image data processing by the deep learning (DL) method. However, the traditional convolution method can only be applied to the grid image data conforming to the "Euclidean domain," and cannot be directly applied to the geometric model. With

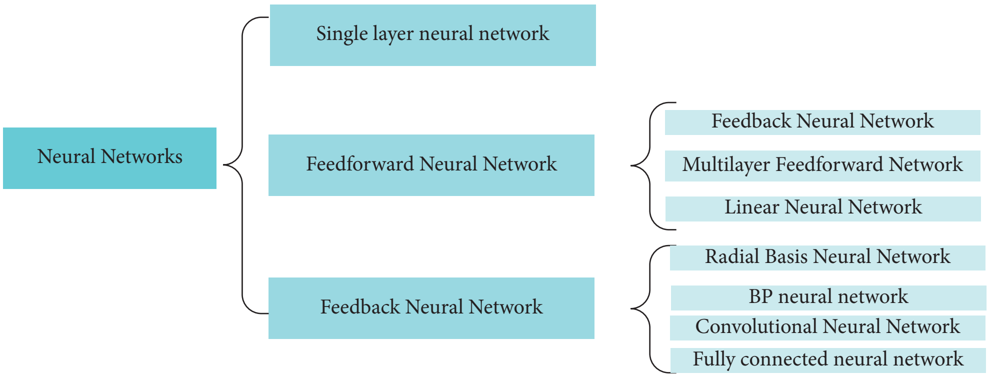
Neural Networks
Single layer neural network
Feedforward Neural Network
Feedback Neural Network
Feedback Neural Network
Multilayer Feedforward Network
Linear Neural Network
Radial Basis Neural Network
BP neural network
Convolutional Neural Network
Fully connected neural network

Figure 2: Classification of neural network models.

the increasing frequency of using the "non-Euclidean domain" in life, GCNN that can be applied to geometric model segmentation has gradually attracted the attention of researchers [23]. The GCNN model mainly applies the concept of convolution operation in the traditional CNN model to the geometric model data based on the GNN model. Unlike the mosaic learning method, the GCNN model does not need to convert the geometric model data into a low-dimensional continuous space vector in advance. It can directly input the whole geometric model into the structure. The graph convolution operation can retain the local feature data of the geometric model, ensuring its structure's overall invariance and integrity to extract the overall features of geometric model data more efficiently and output the feature expression accordingly [24]. The flow of the GCNN model is indicated in Figure 3:

Figure 3 refers that the operation of GCNN is divided into the forward propagation and back propagation. In the forward propagation process, the input vector and target output value are given after the initial value is input, the hidden layer is calculated according to the internal operation rules, and the output value of the output layer is calculated. The output value is compared with the target value to calculate the deviation between the target value and the actual output value. If the deviation value is within the allowable range, the training will end and the weight and threshold are fixed. If it is not within the allowable range, the error of neurons in the network layer will be calculated. Moreover, the error gradient is calculated to update the weight according to it. The output values of the hidden and output layers are retrained to continue comparing the error until the error value is within the allowable range [25].

Spectral Graph Convolution transforms node features into signals in the Fourier domain. Fourier transform is to convert the signal in the time domain into the signal in the frequency domain. The processing method is to integrate the signal in the time domain with the characteristic function of the Laplace operator. Based on Fourier transform, Spectral Graph Convolution integrates the signal in the time domain with the graph Laplace operator, aggregates the node features by Laplace transform, and finally calculates the output [26]. For the input image G, the Laplace operator matrix is

$$L = D - A. \tag{3}$$

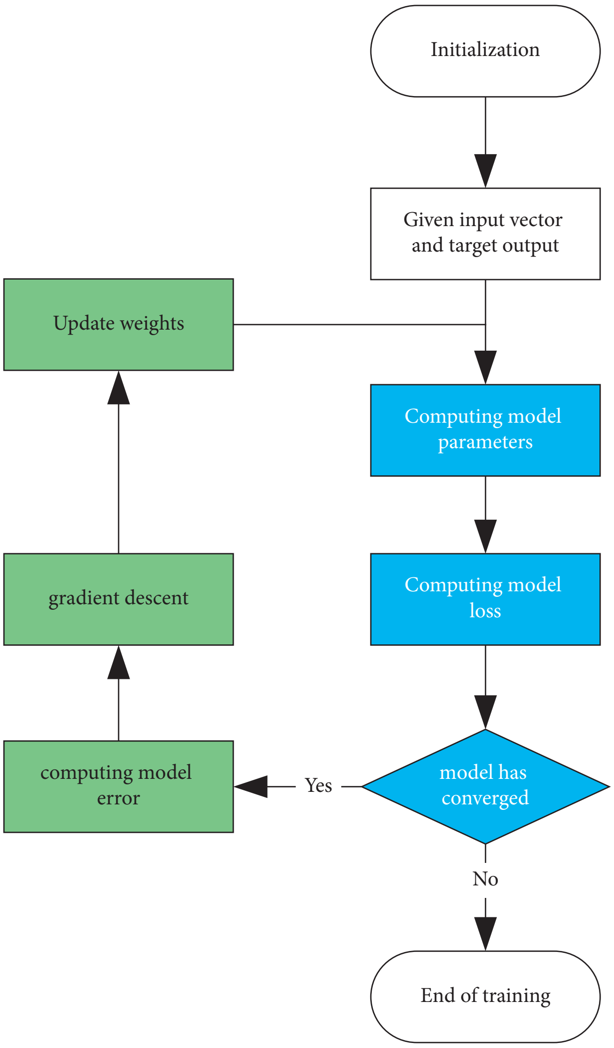

Figure 3: Flow of GCNN.

D is the diagonal matrix of nodes, and each element on the diagonal of the matrix is the degree of each node in turn. A is the adjacency matrix of graph nodes. Generally, the symmetric positive definite normalization of the Laplace matrix is indicated as follows:

$$L = I_N - D^{1/2} * A * D * D^{-1/2}, \tag{4}$$

I_N represents the identity matrix of graph nodes. The feature decomposition of L can be expressed as follows:

$$L = U \cap U^T. \tag{5}$$

$U = \{u_0, u_1, \ldots, u_n\}$ is the eigenvector matrix sorted by eigenvalue.

In the process of graph signal processing, the graph signal represents the eigenvector X of graph nodes, $X = \{x_i\}_{i \in N}$. The Fourier transform of X is as follows:

$$F(x) = U^T * x. \tag{6}$$

The inverse Fourier transform can be expressed as follows:

$$F^{-1}(\hat{x}) = U * \hat{x}, \tag{7}$$

$\hat{x}$ represents the converted signal obtained by the Fourier transform of graph signal x. According to Laplace transform principle, in image G, for the graph signal x, the convolution operation through the convolution kernel g can be written as follows:

$$x_G g = F^{-1}(F(x) \odot F(g)). \tag{8}$$

The mathematical expression of spectral graph convolution operation suggests that the convolution operation of spectral GNN is similar to that in traditional CNN. Through the inner product operation of the convolution kernel, the information expression of the previous layer is aggregated to the next layer as the network input, and the final spectral GCNN model is formed by stacking multiple convolution layers [27].

2.4. WCPL Down-Sampling Algorithm. In the process of processing point cloud data with a graph neural network, it is often necessary to Down-Sample the point cloud to calculate the high-level features of the point cloud. Common Down-Sampling algorithms include farthest point sampling (PointNet++ [28], ShellNet [29]), random sampling (RandLA-Net [30]), grid sampling (KPConv [31]), and so on. Although these algorithms can Down-Sample 3D point cloud data simply and clearly, one problem with these sampling algorithms is that they do not consider the importance of each point.

Based on the criterion that the more 3D point cloud data contributes to the feature after max-pooling, the higher the importance of the point, the WCPL Down-Sampling algorithm is proposed [32]. As shown in Figure 4, suppose there are n points, and the feature dimension of each point is m. The purpose of WCPL is to select the d most important points (Critical Points) from the n points. The algorithm operates as follows:

(1) Calculate a feature 1×d with max pooling. As can be seen from the above figure, the result is actually: the maximum value (*max*) of each column in the feature matrix, and the obtained points are called Critical Points. Here, the index of the maximum value (arg_max) in each column is additionally recorded, and an index array *idx* is obtained.

(2) In the obtained *idx*, a point may appear multiple times. Use a Set to remove duplicates, and the resulting index set *uidx* is called: Critical Set. At the same time, the items of the same index are accumulated to obtain a new feature *fs*, each of which corresponds to each item of *uidx* one-to-one, and *fr* represents the number of times each point contributes.

(3) For *uidx*, sort in descending order according to *fs*, and the index with the highest contribution comes first. Got a sorted list of indices: *suidx*.

(4) Then according to this number of times, a repeat operation is added later to get *midx*.

(5) To make the midx of different point cloud feature maps the same length, use the resize operation to adjust the midx.

(6) Use *rmidx* to generate a point set from the input gather to complete the Down-Sampling operation.

2.5. Construction of GCNN Set Segmentation Model. To accurately segment each point of the 3D point cloud data, the network model constructed in this paper should be able to identify deeper features of the point cloud data. The DenseNet [33] network proposes a dense connection mechanism that connects all layers with each other. Specifically, each layer of the neural network accepts all the previous layers as its additional input. In view of the great success of the DenseNet network in the field of image semantic segmentation, this paper proposes a 3D point cloud data segmentation network with a connection density similar to the DenseNet. The structure is shown in Figure 5.

The network proposed in this paper adopts the modes of Encoder and Decoder. The Encoder module can be divided into four parts. The first part consists of a k-NN Graph layer with $k = 40$ and an Edge Convolution Multilayer Perceptron (Edge Conv MLP) layer with 3 parameters in input and 64 parameters in output. The remaining three parts are composed of a WCLP layer, a k-NN Graph layer, and 2 Edge Conv MLP layers. The WCPL sampling method can dynamically adjust the number of data points calculated by each module of the model and reduce the computational complexity of the model. The WCPL sampling points of part 2, 3, and 4 are 2048, 1024, and 512, respectively. In the Encoder module, the point cloud data points are locally divided by using k-NN Graph layers of different scales, and local features are extracted and aggregated by using Edge Conv layers of different scales. The features output after the convolution of the last side and the features output by the previous layer of MLP together constitute the input of the modified MLP. After a series of nonlinear transformations of the MLP layer, the features are finally output to complete the extraction of 3D point cloud data information.

In the Decoder module of the model, corresponding to each module of the Encoder module, k-NN interpolation is used to interpolate the set of sampled data points and upsampling to expand the number of data points and the number of features in the dataset. Set the number of k-NN

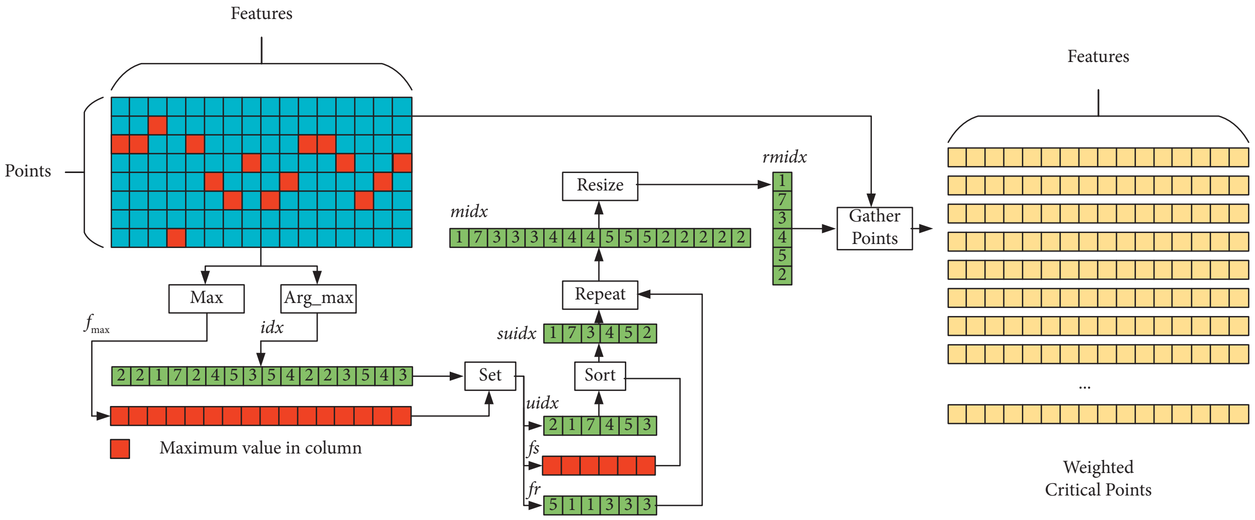

Figure 4: WCPL Down-Sampling flowchart.

interpolation in each stage to be the same as the current point cloud data, then double the number of point clouds through an Up-sampling layer, and then use.

K-NN Graph layers of different scales to complete the local division of point cloud data. After that, through the edge convolution layer and through a series of MLP layer nonlinear transformation finally output point cloud feature data. Through the Up-Sampling of the Decoder module, the Down-Sampling feature dimension in the Encoder module is restored to the original size.

In addition to the main Encoder and Decoder modules, we also add cross-layer horizontal connections at the deepest left and right ends of the model. The fusion of different scale features in the process of enhanced data segmentation improves the model's ability to obtain more local multi-scale features.

3. Experiments and Results

3.1. The Experiment of GCNN Set Segmentation Model. The computer configuration used in this experiment is Intel@CoreTMi7-7800X CPU, NVIDIA GeForce 3090 Ti(24G) GPU, and the operating system is Windows Professional Edition. The network model of this experiment is built based on Pytorch1.2. The segmentation experiments of the geometric model use the ShapeNet part dataset. The original data set is trained and cleaned by CNN. The specific data cleaning steps are exhibited in Figure 6:

In Figure 6, first, the data of the initial data set is reviewed to determine whether it is complete and whether there are errors in the data, screen the wrong data, and discard the unavailable data. Second, the preliminarily processed data is sorted and cleaned. Data cleaning mainly refers to "cleaning" the data by filling in missing values, smoothing noisy data, identifying or removing outliers, and resolving inconsistencies. It is mainly to achieve the following goals: format standardization, abnormal data removal, error correction, and removal of duplicate data. The step of data integration is further carried out. Data integration refers to the combination of data from multiple data sources and unified storage to establish a data warehouse. Meanwhile, the data are transformed into a form suitable for analysis. We uniformly sample 4096 data points for each 3D point cloud model to generate a point cloud dataset of a single object, which is used as the original input of the point cloud separation model, and each data point has a unique pre-label information label, which is used as a supervised Point 3D Cloud segmentation experiment test basis.

The segmentation model parameters include edge convolution layers, fully connected layers, activation functions, and other feature parameters. At the same time, each layer of the network uses standardized operations to standardize the input data, and adds a Dropout operation to the MLP layer at the end of the network to prevent the network from overfitting. The dropout rate of neurons is 0.6; the optimization process of the parameter uses the Adam optimization function, and the adjustment momentum is set to 0.85. The initial learning rate is set to 0.001, and it is set to dynamically decay, after every 20 training epochs, the learning rate decays to the original 0.8. For each Up-Sampling process, the number of interpolation data points of the KNN interpolation algorithm is $k = 1$, the dimension is the average of the feature dimensions of adjacent nodes, the batch size is16, and the total epochs is 1000.

3.2. Accuracy Analysis of the Segmented Region. The segmented geometric model area is more accurate and abstract. The training sample set is trained for 20 iterations, that is, the model after the previous training is used as the initial value of the model for the next training. Figure 7 exhibits the iterative training results of the segmentation model. Figure 7 reveals that as the number of training iterations increases, the region recognition accuracy of the model on the training set is higher. It means that the model's performance is better and better with the increase of training iterations, and The IoU performance of the model in the final training stage can be stabilized at around 0.91.

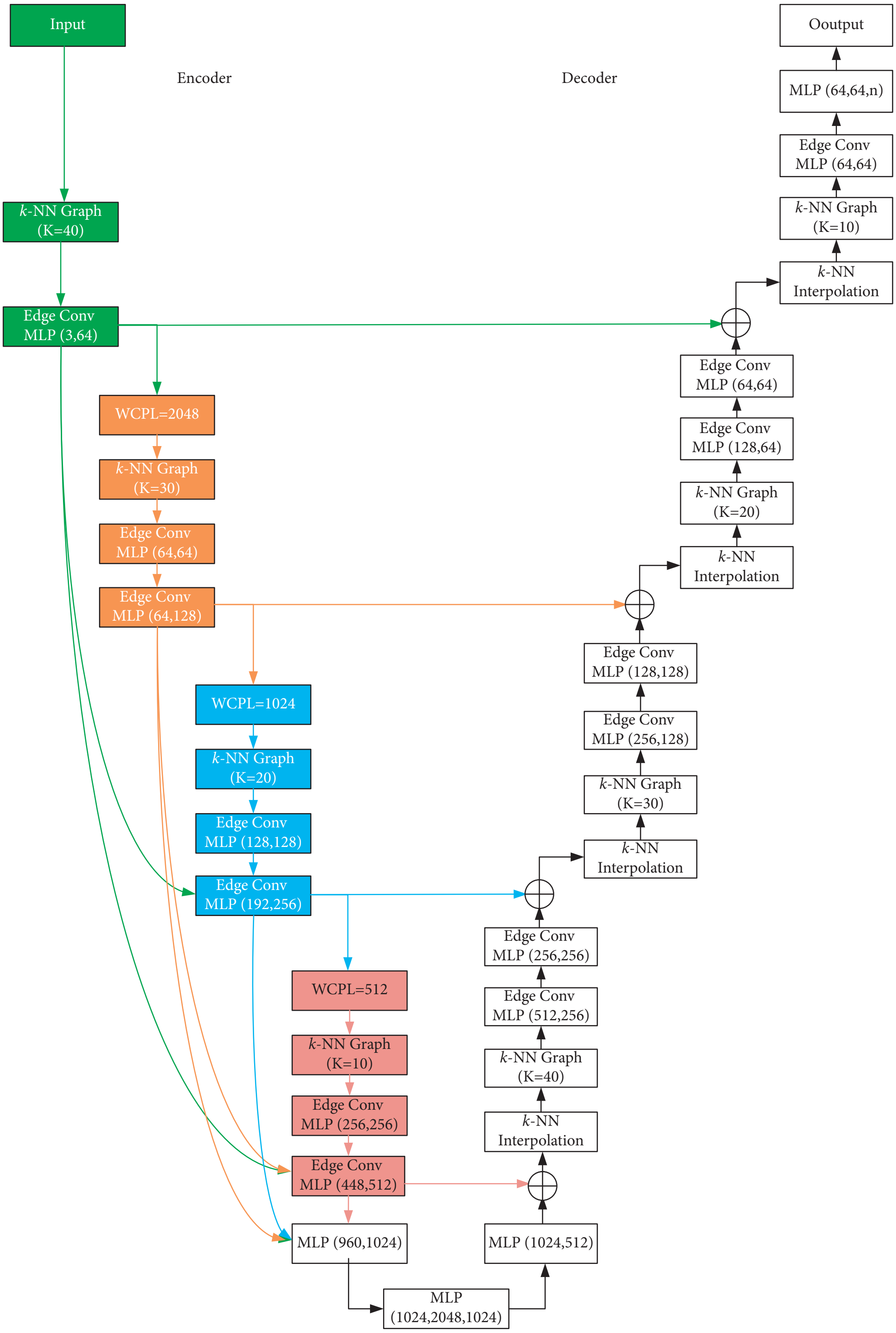

Figure 5: Structural diagram of the geometric segmentation model.

3.3. Intersection over Union (IoU) Analysis of Model Region Segmentation. The model area segmentation IoU is analyzed. IoU refers to the ratio of the intersection and union of "prediction frame" and "real frame," which is a performance's quantitative index at the pixel level. The larger the IoU is, the higher the overlap between the predicted value and the real value is, and the more accurate the predicted result is. This formula 9 is used to describe the degree of coincidence between two categories. The formula numerator is the intersection of the two classes, and the formula denominator is the union of the two classes, so their ratio represents the intersection ratio.

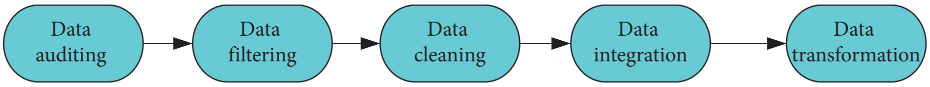

Figure 6: Steps for data cleaning.

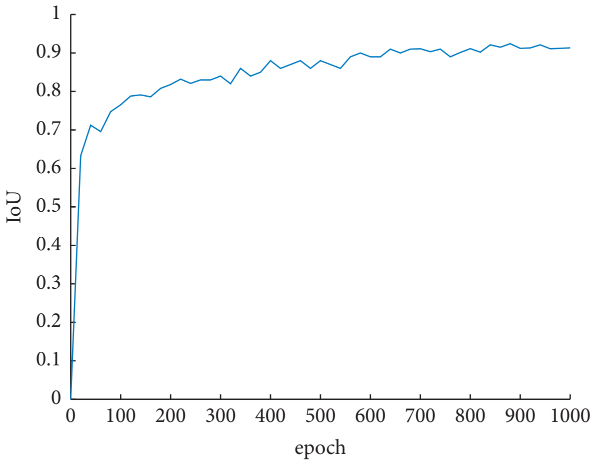

Figure 7: Test results of iterative training.

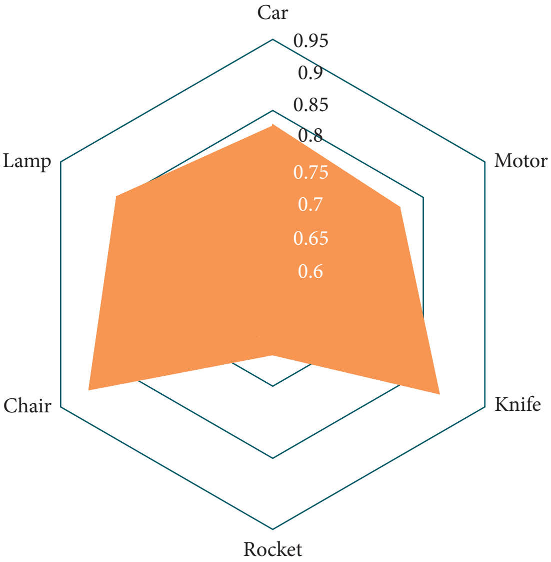

Figure 8: Schematic diagram of IoU and its analysis results.

$$\mathrm{IoU} = \frac{A \cap B}{A \cup B}. \tag{9}$$

Figure 8 indicates that from the segmentation effect of single category objects, the average IoU value of the segmentation model has reached more than 0.85, and the segmentation effect has reached a very mature level. The segmentation results of this model are compared with other existing geometric segmentation models for different objects. Figure 9 denotes the results.

In Figure 9, the proposed algorithm and mainstream point cloud data segmentation algorithms including PointNet [34], KD-Net [35], PCNN [36], and Point CNN [37] are used to segment Car, Lamp, Rocket, Knife, Chair, and Motor. PCNNs can connect and create highly flexible physiological filters. GCNN is a model based on CNN and a simple attention mechanism, which integrates the relatively new dilated convolution and gated convolution, and adds some artificial features. From the overall effect, the segmentation effect of the model proposed in this

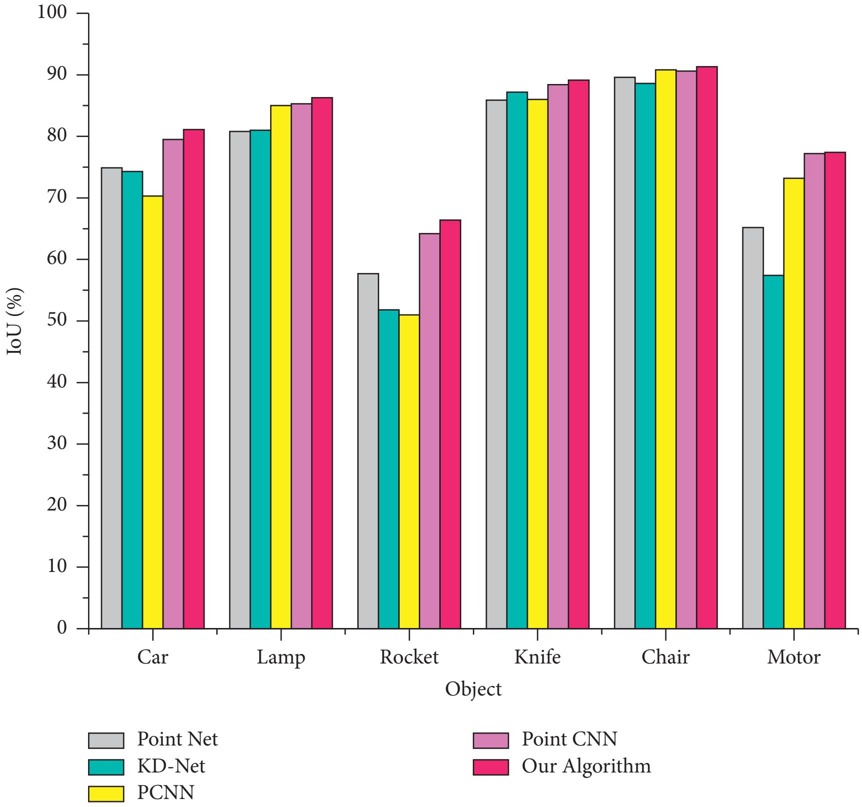

Figure 9: Comparison of segmentation effect.

paper is better than other 3D segmentation models, which proves the superiority of the algorithm proposed in this paper.

4. Conclusion

The 3D segmentation method based on the GNN model is a recent research hotspot and difficulty in the field of data modeling. Due to the irregularity of geometric model data, traditional DL algorithms cannot be directly and effectively applied to the modeling processing of geometric model data. GNN model is an emerging hotspot in the field of DL. In view of the need to retain detailed features in the simplification of 3D geometric model segmentation, a region segmentation algorithm based on GCNN 3D geometric model is proposed. The algorithm avoids the disadvantage that the traditional CNN cannot directly act on the data processing of the geometric model, and has strong anti-noise ability and robustness. Through the comparison experiments with other segmentation algorithms and the IoU of segmentation results, the findings manifest that this method is a convenient, practical, and effective region segmentation algorithm, and the segmentation model has a good segmentation effect, which can provide an auxiliary role for the segmentation of geometric models. The research results can provide a reference for the application of related geometric model segmentation technology and neural network in the field of geometric model and image segmentation. The disadvantage is that due to the short research time and the limited number of samples, the scope and depth of the investigation have certain deficiencies, and the sample size is small. In the future, the scope of the investigation will be expanded for further research. Meanwhile, neural network technology keeps pace with the times, and new technologies will be updated and used in the future. The theory and practice will be deeply combined, and follow-up in-depth research will be carried out.

Data Availability

The datasets used and analyzed during the current study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was supported by the National Key R&D Program of China (grant no. 2017YFC0804310) and the General Special Scientific Research Plan of Shaanxi Provincial Department of Education (grant no. 20JK0754).

References

[1] F. Grotto, S. Rivallant, and C. Bouvet, “Development of a 3D finite element model at mesoscale for the crushing of unidirectional composites: application to plates crushing,” *Composite Structures*, vol. 287, Article ID 115346, 2022.

[2] N. O. Moraga and D. R. Rivera, "Advantages in predicting conjugate freezing of meat in a domestic freezer by CFD with turbulence k-ε 3D model and a local exergy destruction analysis," *International Journal of Refrigeration*, vol. 126, pp. 76–87, 2021.

[3] Q. Lin, H. Huang, L. Chen, J. Wang, K. Huang, and Y. Liu, "Using the 3D model RAPID to invert the shoot dieback ratio of vertically heterogeneous Yunnan pine forests to detect beetle damage," *Remote Sensing of Environment*, vol. 260, Article ID 112475, 2021.

[4] P. Bangde, T. Pant, G. Gaikwad, R. Jain, and P. Dandekar, "Trimethyl chitosan coated palladium nanoparticles as a photothermal agent and its in vitro evaluation in 2D and 3D model of breast cancer cells," *Colloids and Surfaces B: Biointerfaces*, vol. 211, Article ID 112287, 2022.

[5] Y. G. Zhang, J. Tang, R. P. Liao et al., "Application of an enhanced BP neural network model with water cycle algorithm on landslide prediction," *Stochastic Environmental Research and Risk Assessment*, vol. 35, no. 6, pp. 1273–1291, 2021.

[6] G. Lee, S. J. Lee, and C. Lee, "A convolutional neural network model for abnormality diagnosis in a nuclear power plant," *Applied Soft Computing*, vol. 99, Article ID 106874, 2021.

[7] C. B. S. Maior, J. M. M. Santana, I. D. Lins, and M. J. C. Moura, "Convolutional neural network model based on radiological images to support COVID-19 diagnosis: evaluating database biases," *PLoS One*, vol. 16, no. 3, Article ID e0247839, 2021.

[8] M. Gori, G. Monfardini, and F. Scarselli, "A new model for learning in graph domains," in *Proceedings of the IEEE International Joint Conference on Neural Networks Proceedings*, pp. 729–734, Montreal, QC, Canada, December 2005.

[9] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, 2009.

[10] D. Kavitha and S. Ravikumar, "IOT and context-aware learning-based optimal neural network model for real-time health monitoring," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 1, Article ID e4132, 2021.

[11] Q. Lu, U. Hasson, and K. A. Norman, "A neural network model of when to retrieve and encode episodic memories," *Elife*, vol. 11, Article ID e74445, 2022.

[12] C. J. Huang, Y. Shen, Y. H. Chen, and H. Chen, "A novel hybrid deep neural network model for short-term electricity price forecasting," *International Journal of Energy Research*, vol. 45, no. 2, pp. 2511–2532, 2021.

[13] S. M. J. Jalali, S. Ahmadian, A. Khosravi, M. Shafie-khah, S. Nahavandi, and J. P. S Catalao, "A novel evolutionary-based deep convolutional neural network model for intelligent load forecasting," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 8243–8253, 2021.

[14] E. B. Moustafa, A. H. Hammad, and A. H. Elsheikh, "A new optimized artificial neural network model to predict thermal efficiency and water yield of tubular solar still," *Case Studies in Thermal Engineering*, vol. 30, Article ID 101750, 2022.

[15] A. K. Verma, S. Nagpal, A. Desai, and R. Sudha, "An efficient neural-network model for real-time fault detection in industrial machine," *Neural Computing & Applications*, vol. 33, no. 4, pp. 1297–1310, 2021.

[16] H. Bi, W. L. Shang, Y. Chen, K. Wang, Q. Yu, and Y. Sui, "GIS aided sustainable urban road management with a unifying queueing and neural network model," *Applied Energy*, vol. 291, Article ID 116818, 2021.

[17] X. Han, Z. Wei, B. Zhang, Y. Li, T. Du, and H. Chen, "Crop evapotranspiration prediction by considering dynamic change of crop coefficient and the precipitation effect in back-propagation neural network model," *Journal of Hydrology*, vol. 596, Article ID 126104, 2021.

[18] A. Deng and B. Hooi, "Graph neural network-based anomaly detection in multivariate time series," in *Proceedings of the AAAI Conference on Artificial Intelligence*, no. 5, pp. 4027–4035, Singapore, 2021.

[19] Z. Guo, L. Tang, T. Guo, K. Yu, M. Alazab, and A. Shalaginov, "Deep graph neural network-based spammer detection under the perspective of heterogeneous cyberspace," *Future Generation Computer Systems*, vol. 117, pp. 205–218, 2021.

[20] P. Saha, D. Mukherjee, P. K. Singh, A. Ahmadian, M. Ferrara, and R. Sarkar, "Retracted ARTICLE: GraphCovidNet: a graph neural network based model for detecting COVID-19 from CT scans and X-rays of chest," *Scientific Reports*, vol. 11, no. 1, pp. 8304–8316, 2021.

[21] C. W. Park, M. Kornbluth, J. Vandermause, C. Wolverton, B. Kozinsky, and J. P. Mailoa, "Accurate and scalable graph neural network force field and molecular dynamics with direct force architecture," *Npj Computational Materials*, vol. 7, no. 1, pp. 73–79, 2021.

[22] S. Chen, J. Dong, P. Y. J. Ha, Y. Li, and S. Labi, "Graph neural network and reinforcement learning for multi-agent cooperative control of connected autonomous vehicles," *Computer-Aided Civil and Infrastructure Engineering*, vol. 36, no. 7, pp. 838–857, 2021.

[23] Y. D. Zhang, S. C. Satapathy, D. S. Guttery, J. M. Gorriz, and S. H. Wang, "Improved breast cancer classification through combining graph convolutional network and convolutional neural network," *Information Processing & Management*, vol. 58, no. 2, Article ID 102439, 2021.

[24] Y. Yin, X. Zheng, B. Hu, Y. Zhang, and X. Cui, "EEG emotion recognition using fusion model of graph convolutional neural networks and LSTM," *Applied Soft Computing*, vol. 100, Article ID 106954, 2021.

[25] S. H. Wang, V. V. Govindaraj, J. M. Górriz, X. Zhang, and Y. D. Zhang, "Covid-19 classification by FGCNet with deep feature fusion from graph convolutional network and convolutional neural network," *Information Fusion*, vol. 67, pp. 208–229, 2021.

[26] X. Yu, S. Lu, L. Guo, S. H. Wang, ResGNet-C, and Y. D Zhang, "ResGNet-C: a graph convolutional neural network for detection of COVID-19," *Neurocomputing*, vol. 452, pp. 592–605, 2021.

[27] W. Chen, M. Jiang, and W. G. Zhang, *A novel graph convolutional feature based convolutional neural network for stock trend prediction*, Information Sciences, vol. 556, , pp. 67–94, 2021.

[28] C. R. Qi, Li Yi, H. Su, and J Leonidas, "Pointnet++: deep hierarchical feature learning on point sets in a metric space," in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 5099–5108, California, USA, December 2017.

[29] Z. Zhang, B. S. Hua, and S. K. Yeung, "Efficient point cloud convolutional neural networks using concentric shells statistics," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1607–1616, Seoul, Korea, 2019.

[30] Q. Hu, B. Yang, L. Xie et al., "Randla-net: efficient semantic segmentation of large-scale point clouds," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11108–11117, 2020.

[31] H. Thomas, C. R. Qi, J. E. Deschaud, B Marcotegui, F Goulette, and L Guibas, "Kpconv: flexible and deformable convolution for point clouds," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6411–6420, Seoul, Korea (South), November 2019.

[32] E. Nezhadarya, E. Taghavi, R. Razani, B Liu, and J Luo, "Adaptive hierarchical down-sampling for point cloud classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12956–12964, Seattle, WA, USA, June 2020.

[33] F. Iandola, M. Moskewicz, S. Karayev, R Girshick, T Darrell, and K Keutzer, "Densenet: implementing efficient convnet descriptor pyramids," 2014, https://arxiv.org/abs/1404.1869.

[34] C. R. Qi, H. Su, K. Mo, and L J Guibas, "PointNet: deep learning on point sets for 3D classification and segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 652–660, Honolulu, HI, USA, July 2017.

[35] R. Klokov and V. Lempitsky, "Escape from cells: deep kd-networks for the recognition of 3d point cloud models," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 863–872, Venice, Italy, October 2017.

[36] K. Zhan, J. Shi, H. Wang, Y. Xie, and Q Li, "Computational mechanisms of pulse-coupled neural networks: a comprehensive review," *Archives of Computational Methods in Engineering*, vol. 24, no. 3, pp. 573–588, 2017.

[37] Y. Li, R. Bu, M. Sun, W Wu, X Di, and B Chen, "Pointcnn: convolution on x-transformed points," in *Proceedings of the Neural Information Processing Systems*, pp. 820–830, 2018.