3D Shape Segmentation with Potential Consistency Mining and Enhancement

Zhenyu Shu, Shiyang Li*, Shiqing Xin, and Ligang Liu

Abstract—**3D shape segmentation is a crucial task in the field of multimedia analysis and processing, and recent years have seen a surge in research on this topic. However, many existing methods only consider geometric features of 3D shapes and fail to explore the potential connections between faces, limiting their segmentation performance. In this paper, we propose a novel segmentation approach that mines and enhances the potential consistency of 3D shapes to overcome this limitation. The key idea is to mine the consistency between different partitions of 3D shapes and to use the unique consistency enhancement strategy to continuously optimize the consistency features for the network. Our method also includes a comprehensive set of network structures to mine and enhance consistent features, enabling more effective feature extraction and better utilization of contextual information around each face when processing complex shapes. We evaluate our approach on public benchmarks through extensive experiments and demonstrate its effectiveness in achieving higher accuracy than existing methods.**

Index Terms—**3D shape segmentation, Consistency, Deep learning, Shape analysis**

I. Introduction

The task of 3D shape segmentation is a fundamental and challenging problem in multimedia analysis and processing with wide-ranging practical implications. It entails the assignment of correct classification labels to individual components of a 3D shape's faces. This process has diverse applications, including but not limited to 3D shape modeling [1], shape retrieval [2], and skeleton extraction [3]. Furthermore, the concept of 3D shape segmentation can be a source of inspiration for certain object detection innovations [4], [5].

Before the advent of deep learning techniques, 3D shape segmentation methods relied on hand-crafted feature descriptors to represent each face of a 3D shape as a feature vector. By clustering these feature vectors in feature space, corresponding labels could be assigned to faces. The combination of multiple descriptors has been widely adopted to enhance performance during the segmentation of 3D shapes and overcome the limitations of a single feature descriptor. However, the efficacy of these methods is still limited by their ability to extract features and handle the complexity of 3D shapes.

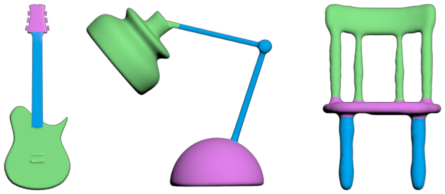


Fig. 1. Example segmentation results of our method.

The application of machine learning in 3D shape segmentation has been found to significantly enhance the ability of algorithms to extract features, leading to improved segmentation performance, such as [6] and [7]. Recent advancements in deep learning have further improved the capability of feature extraction. For instance, Guo et al. [8] employed convolutional neural networks to process matrices generated from geometric features, achieving satisfactory segmentation results. Similarly, MeshWalker [9] utilized recurrent neural networks for random walks on 3D meshes to segment 3D shapes, achieving state-of-the-art performance. Despite the significant improvement in segmentation accuracy, these methods primarily focus on geometry information, such as dihedral and internal angles, without fully exploiting the potential consistency of 3D shapes.

In comparison to the previously studied methods, our proposed approach presents a pioneering innovation in feature extraction for 3D shapes. Unlike the conventional practice of relying solely on geometric features on the surface, our approach emphasizes potential consistency within the 3D shapes. This consistency, when mined, enables our network to differentiate between the various partitions in 3D shapes more effectively. For labeling faces, we employ a basic deep-learning network while incorporating various strategies for processing diverse faces. These strategies facilitate the prioritization of data consistency while simultaneously assimilating contextual information from neighboring local faces. Sample results of our approach can be seen in Figure 1.

The main contributions of this paper are as follows:

- A novel strategy is proposed to mine and enhance the potential consistency of the 3D shape itself, which can significantly improve the network's ability to learn 3D shape features. Meanwhile, we design a learning network that can effectively use this consistency enhancement strategy to dynamically mine potential features in data

Zhenyu Shu is with School of Computer and Data Engineering, NingboTech University, Ningbo, PR China.
E-mail: shuzhenyu@nit.zju.edu.cn (Zhenyu Shu)
Shiyang Li is with College of Computer Science and Technology, Hangzhou, PR China. Corresponding author.
E-mail: shiyangli_paper@163.com (Shiyang Li)
Shiqing Xin is with School of Computer Science and Technology, ShanDong University, Jinan, PR China.
Ligang Liu is with Graphics & Geometric Computing Laboratory, School of Mathematical Sciences, University of Science and Technology of China, Anhui, PR China.

Manuscript received month day, year; revised month day, year.

to obtain more precise segmentation results.
- Extensive experimental results on publicly available benchmarks show that our method performs significantly better than the state-of-the-art approaches.

The rest of the paper is organized as follows. In Section II, we review recent works on 3D shape segmentation. In Section III, we elaborate on our consistency enhancement strategy and the structure of our network. We present the results of experiments used to demonstrate the validity of our method in Section IV. Section V summarizes the limitations of the current work and the prospects for the future. We conclude this paper in Section VI.

II. Related work

3D shape segmentation is a fundamental component of multimedia analysis and processing, involving the semantic labeling of faces within individual parts of a 3D shape. In this section, we present a comprehensive review of 3D shape segmentation methods, which are categorized into three groups: traditional, unsupervised, and supervised segmentation methods. By conducting a detailed analysis of each method's strengths and limitations, we provide a more in-depth overview of the 3D shape segmentation techniques.

Traditional 3D shape segmentation methods and unsupervised 3D shape segmentation methods. The segmentation of 3D shapes has traditionally relied on mathematically defined geometric features. Early approaches to 3D shape segmentation, as categorized in [10], include methods such as hierarchical clustering [11], iterative clustering [12], region growth [13], boundary segmentation [14], watershed segmentation [15], and medial axis transform [16]. Among these, the clustering-based method [17], [18] is characteristic, dividing the 3D shape based on the clustering result of the corresponding feature vector of each 3D shape in feature space. Region growing is another segmentation technique, whereby some faces or vertices are placed as seeds on each part of the 3D shape and allowed to spread around until the entire 3D shape is segmented. Based on topological theory, Watershed segmentation is particularly effective for detecting weak edges that are less obvious in 3D shapes. These methods form the basis of many state-of-the-art approaches to 3D shape segmentation.

Semi-supervised 3D shape segmentation methods. Recently, there has been a growing interest in semi-supervised segmentation methods for 3D shapes, alongside traditional techniques. Researchers have proposed innovative approaches to overcome the challenges of segmenting complex 3D shapes. For example, Sidi et al. [19] utilize spectral clustering and diffusion mapping to establish the relationship between faces during the 3D shape segmentation process, while Zhuang et al. [20], [21] employ mesh embedding and correlation clustering methods to achieve semi-automatic alignment of mesh boundaries with ridge and valley lines. In another study, Wu et al. [22] leverage a patch-based segmentation method followed by spectral clustering in descriptor space, resulting in an improved unsupervised segmentation outcome. Furthermore, Zhang et al. [23] propose a novel unsupervised segmentation method based on face-level descriptors and soft clustering. The findings from these studies significantly advance 3D shape segmentation and provide valuable insights for future research. Yu et al. [24] propose a novel deep learning model for hierarchical segmentation of 3D shapes, based on top-down recursive decomposition and recursive neural networks, which segments a 3D shape into an arbitrary number of parts, achieves state-of-the-art performance on public and new benchmarks for fine-grained and semantic segmentation, and can be applied for fine-grained part refinements in image-to-shape reconstruction.

Supervised 3D shape segmentation method. The supervised 3D shape segmentation techniques have gained considerable notoriety in recent years due to their capacity to establish a mapping relationship between features and labels through prior knowledge. These methods have several beneficial factors. The increasing completeness of the 3D shape repository has provided a strong data foundation for supervised 3D shape segmentation, and the continuous evolution of machine learning and deep learning has further propelled the field of 3D shape segmentation. All these factors have contributed to the superior performance of supervised 3D shape segmentation over traditional and unsupervised methods.

Kalogerakis et al. [25] propose a learn-based 3D shape segmentation method for the first time in the field of 3D shape segmentation. The proposed approach involves classifying and labeling the mesh and optimizing the loss function parameters based on the CRF model. Similarly, Kaick et al. [26] propose a method which considers the geometric similarity between 3D shapes and introduces an objective function composed of multiple loss terms to learn the data features of the mesh. This method effectively utilizes geometric features and ensures smooth transitions between different areas.

Several supervised learning methods for 3D shape segmentation that rely on geometric description features have been proposed [27], [28], [29], [30]. These methods exhibit improved performance and rely on hand-crafted feature descriptors, such as mean geodesic distance(AGD) [31] to represent global position information, the shape diameter function (SDF) [3] to distinguish the fat and thin parts of 3D shapes by measuring the diameter of local facial shapes, and the Gaussian curvature (GC) to represent the curvature of each vertex in 3D shapes. Xie et al. [7] try to use extreme learning machines in the field of 3D shape segmentation. The application of 3D shape segmentation in deep convolutional networks originates from the method of Guo et al. [8], which uses a 2D matrix composed of feature descriptors as training data. Zhu et al. [32] present an efficient point cloud segmentation method based on prototypes for bias rectification by reducing the distribution distance between the support set and query set features. Apart from the algorithms that rely on feature descriptors, there are excellent methods that do not use feature descriptors. The method proposed by Wang et al. [33] is different from the previous methods using feature descriptors. It uses projection to introduce the semantic segmentation of 2D images into the segmentation of 3D shapes. Besides, Kalogerakis et al. [25] also use an image-based 2D segmentation network to label the projection and achieve a good effect in segmenting 3D

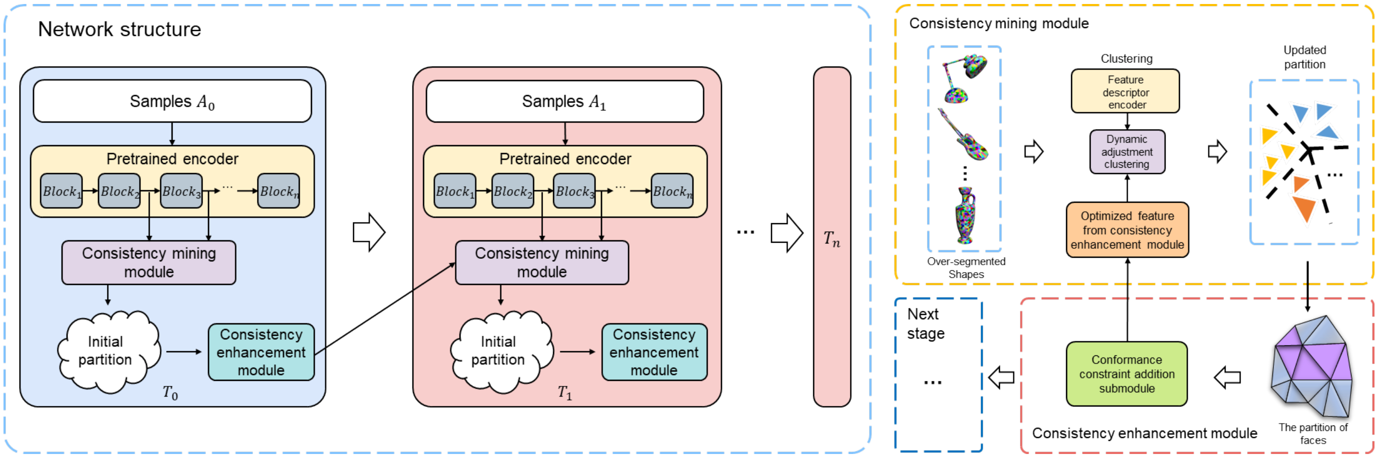

Network structure
Samples A_0
Pretrained encoder
$Block_1$
$Block_2$
$Block_3$
$Block_n$
Consistency mining module
Initial partition
Consistency enhancement module
T_0
Samples A_1
T_1
T_n
Consistency mining module
Over-segmented Shapes
Clustering
Feature descriptor encoder
Dynamic adjustment clustering
Optimized feature from consistency enhancement module
Updated partition
Next stage
Conformance constraint addition submodule
Consistency enhancement module
The partition of faces

Fig. 2. The overview of our algorithm, which mainly contains four steps. 1) Sample faces on 3D shapes. 2) Cluster the feature vectors, and obtain the partition of 3D shape. 3) Add constraints for consistency to the network based on partitions of the 3D shape. 4) Update the feature vector for the next iteration, and send the optimized feature vector to the consistency mining module. The corresponding label of the face is the output of our network.

shapes. Le et al. [34] use RNN in segmentation tasks and make innovations in data format, inputting multiple sequences composed of 2D projections transformed from 3D shapes into RNN networks. Similarly, MeshWalker [9] also uses the sequence input RNN network for segmentation. The difference is that the sequence used by MeshWalker is generated by random walking on the surface of the mesh. Guo et al. [35] introduce Point Cloud Transformer, a novel framework based on Transformer for point cloud learning, which addresses the challenges of designing deep neural networks for point cloud processing due to the irregular domain and lack of order. Han et al. [36] propose a novel point cloud segmentation method by capturing much richer contextual dependencies semantically from the perspective of position and channel. Weng et al. [37] present a plane-assisted module by enhancing semantic segmentation of touching objects and large surface objects in point clouds. Wu et al. [38] propose a comprehensive intra- and cross-modal contrastive learning method for segmenting 3D point clouds by combining rich learning signals from point clouds and rendered images.

Unlike the above methods, our method focuses on mining the potential features of the 3D shape itself and leveraging them to enhance the network's performance by strengthening the capacity to understand and extract the consistency of the features. Notably, our approach integrates an innovative strategy for mining and enhancing data consistency that considers the contextual relationships and local scale information among faces, thereby playing an original role in this field.

III. Our Method

Our algorithm's general overview is outlined in Figure 2. In our method, feature descriptors that capture semantic information from various perspectives are employed to encode the characteristics of the 3D shape. Initially, a preliminary partition of the 3D shape is generated to ensure that most faces within each partition share the same label. With the preliminary partition, our network, which comprises two functional modules, including the consistency mining module and the consistency enhancement module, subsequently introduces a training constraint to enforce consistent and accurate labeling of the partitioned results. It is worth noting that the partition would undergo optimization throughout the training iterations. After the network assigns the corresponding label to each face, the graph-cut algorithm is used to refine the segmentation results further. A detailed description of our algorithm is introduced in the following.

A. Consistency mining module

The objective of the consistency mining module is to generate partitions of 3D shapes that exhibit a strong label consistency among the samples in each partition. Depending on this module's effect, the other module we proposed in our algorithm, the consistency enhancement module, employs robust constraints during the training process, thereby enhancing the overall labeling accuracy and minimizing any potential inconsistencies.

As described in Figure 2, in the beginning, our algorithm employs the k-means clustering algorithm to initially over-segment the input 3D shape. To ensure the reliability of the initial partition, five feature descriptors, including shape-diameter function (SDF), Gaussian curvature (GC), average geodesic distance (AGD), scale-invariant heat kernel signature (SIHKS, [39]), and wavelet kernel signature (WKS, [40]), are selected and used to extract feature vectors. The AGD, SDF, and GC descriptors yield one-dimensional geometric features, while WKS and SIHKS produce 19-dimensional and 100-dimensional vectors, respectively. These feature vectors are concatenated into a 122-dimensional feature vector for each face.

The effectiveness of the consistency constraints implemented by the consistency enhancement module relies on the accuracy of partitions generated by the consistency mining module. Therefore, it is essential to ensure the accuracy of partitioning. We propose a learnable segmentation strategy that ensures partition accuracy during network training. Specifically, after each training iteration, feature vectors generated

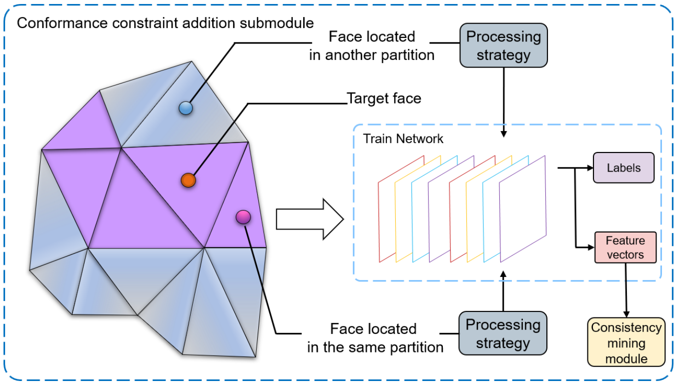

Conformance constraint addition submodule
Face located in another partition
Processing strategy
Target face
Train Network
Labels
Feature vectors
Face located in the same partition
Processing strategy
Consistency mining module

Fig. 3. After receiving the partitions divided by the consistency mining module, different strategies are adopted for the faces belonging to different partitions, and the constraints for consistency are added to the network according to different process strategies. After the labels are generated, the optimized feature vectors would be passed to the consistency mining module for subsequent iteration.

by the network's last layer are selected as the feature representation for each face. These vectors are then used to update the feature vectors utilized in the clustering process of the consistency mining module. Our self-optimizing segmentation strategy establishes a two-way connection between the consistency mining and enhancement modules, thus preventing sample overfitting caused by one-way information propagation.

While our approach may appear similar to deep clustering, it differs in that deep clustering combines clustering and deep learning, but the connection between clustering and deep learning is generally one-way and static. In deep clustering, the clustering result is only used as additional information to assist training and is not adjusted further during the network's training process. To ensure the success of deep clustering's training strategy, the clustering accuracy must be very high. In contrast, our method enables dynamic adjustments to the clustering process through feedback from the network optimization iterations in the consistency enhancement module. Our approach effectively integrates clustering into the supervised learning process, thereby improving accuracy in guiding model learning and adjustment.

B. Consistency enhancement module

The purpose of this module is to regulate the training process of the network. Our method leverages the partitioning performed by the consistency mining module to add a constraint to the network, generating labels with solid consistency. To achieve this, we introduce the concept of conditional entropy as a constraint on the consistency between faces. In our approach, we define conditional entropy as the expectation of the entropy of the face *F* corresponding to the label *Y*, given that the face is located in the partition P_x. This definition represents the uncertainty of the face category in a partition. By adding this constraint to the loss function, we can enhance the consistency of the faces in 3D shape by reducing the uncertainty of the class to which the faces belong.

Figure 2 illustrates the detailed architecture of the consistency enhancement module. Following the partitioning phase by the consistency mining module, the module traverses the adjacent faces of each face *F* in the triangular mesh to identify their location in the partition. After identifying, the treatment of the adjacent faces is divided into two cases: For the faces located within the same region as per the partitioning, the module does not perform any further processing, just computing their conditional entropy, which is then integrated into the loss function as a regularization term. The face situated at the boundary between partitions is the second situation. This kind of face plays a crucial role in improving the accuracy of 3D shape segmentation, and we consider it a critical factor in enhancing the model's performance. The first step in handling this type of face is to traverse the adjacent faces of *F*. During traversal, the surrounding faces can be divided into two sets. Set S_1 consists of faces located in the same partition with *F* according to the consistency mining module, while set S_2 comprises faces with different partitions from *F*. Upon obtaining these two sets, we employ different processing strategies for faces based on their distribution within each set. Merely imposing constraints on network training based on the proportion of faces in each set is not applicable to avoid overfitting. Therefore, only when the difference between the number of faces in S_1 and the number of faces in each partition within S_2 exceeds a certain threshold, we add the conditional entropy of the current partition's face *F* as a regularization term to the loss function. As a result, the network's loss function can be expressed as follows:

$$Loss = L_{origin} + \lambda H_{in} + \lambda H_{edge}, \tag{1}$$

$$L_{origin} = -\sum_{i} y_i \log p_i, \tag{2}$$

where L_{origin} represents the cross-entropy loss of the network itself, measuring the difference between the prediction of a measurement model within a class of three-dimensional shapes and the true labels of all classes, y_i is the i-th element of the real label, and p_i is the i-th class probability predicted by the model. H_{in} consists of the expectation of the conditional entropy of the face within the partition, H_{edge} represents the expectation of conditional entropy of the face on the marginal part between partitions, and λ denotes their regularization coefficient. The mathematical expressions of H_{in} and H_{edge} are:

$$H_{in} = E_{(F_{in},Y,A)}[P(Y|F_{in} \in A)log(P(Y|F_{in} \in A))], \tag{3}$$

$$H_{edge} = E_{F_{edge} \in A}[H'], \tag{4}$$

$$H' = P(Y|Z, F_{edge} \in A)log(P(Y|Z, X_{edge} \in A)), \tag{5}$$

$$Z : \sum_{i \in S_1} F_i - \sum_{j \in S_2} F_j > \mu. \tag{6}$$

Among the above equations, condition Y represents the probability that the category of face F is the same as most faces located in partition A. We use the expectation of H' to represent H_{edge}, where H' indicates conditional entropy of the face on the marginal part between partitions, i.e., the entropy of the situation where condition Y is satisfied under condition Z. μ represents the threshold of the difference in the number of faces between S_1 and S_2. Only the conditional

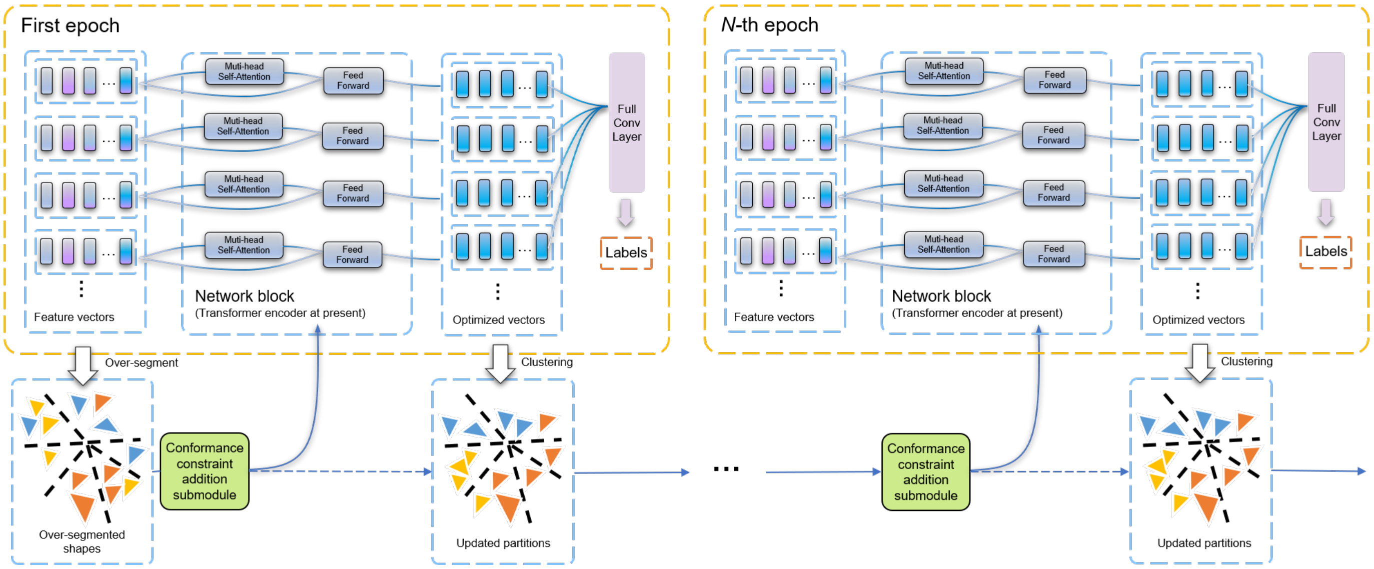


Fig. 4. The detailed structure of the deep learning network. Initially, we generate a preliminary partition of the 3D shape. Subsequently, we will subject consistency constraints to the training network. After each training epoch, the optimized vectors will be generated and passed to the next iteration. Throughout the network's training iterations, the updated partitions outcome by clustering gradually undergoes optimization. We employ the Transformer model in the network block. The Transformer encoder takes feature vectors and encodes them into optimized vectors. The self-attention mechanism allows the model to know the importance of each feature descriptor. The encoded embeddings are fed to the fully connected layer connected to softmax for classification. In addition, the network framework in the network block can be replaced with other applicable networks, such as MLP or LSTM.

entropy of face F satisfying condition Z can be included in the loss function.

The conformance constraint addition submodule is a submodule inside the consistency enhancement module. Figure 3 illustrates the detailed network structure of it. This submodule is designed to add a constraint to the network. The optimized feature vectors within the network are then transmitted to the subsequent iteration of the consistency mining module to partition the faces. Moreover, the deep network structure used in the study is depicted in Figure 4. We employ the Transformer model in the network block to account for complex relationships between input feature vectors. The Transformer's multi-head self-attention mechanism can simultaneously focus on different parts of the input, facilitating the learning of more complex features.

After training, the consistency processing strategy, which is contained in the consistency enhancement module, is obtained and can ensure consistent and accurate labels. Simultaneously, this constraint-based approach enables the perception of contextual information from the surrounding faces of each face. In traversing the faces around the target to establish a face set, the local information around the target face is also extracted. This context-aware module structure mitigates the risk of overfitting during the inclusion of training constraints.

C. Algorithm

The training and testing process of our segmentation method is executed on each category of 3D shapes. The inputs and outputs of each stage in our method and the operations of each module within the network, can be summarized as Algorithm 1.

Algorithm 1 Consistency mining and enhancement

```
Inputs: Training 3D shapes and human-assigned labels for all faces
Outputs: Predicted label for each face on test 3D shapes
Training process:
1: Compute feature vectors for each face in training 3D shapes using feature descriptors, including AGD, SDF, GC, SIHKS, and WKS. Concatenate those feature vectors into high-dimensional vectors;
2: repeat
3:     Cluster the feature vectors, and obtain the updated partition of 3D shape;
4:     Add constraints for consistency to the network based on partitions of the 3D shape through conformance constraint addition submodule;
5:     Train our network, shown in Figure 4, using the training data prepared in Steps 1 and 3;
6:     Update the feature vectors;
7: until Iteration over.
Testing process:
1: Compute feature vectors for each face in testing shapes;
2: Predict the probability distributions with the trained network and obtain segmentation labels;
3: Employ the graph-cuts algorithm to the predicted results for smoothing boundaries.
```

D. Network training

Some examples of segmentation results after each training stage are shown in Figure 5. In summary, our network primarily performs classification tasks. During network training, we input feature vectors corresponding to each face. The output

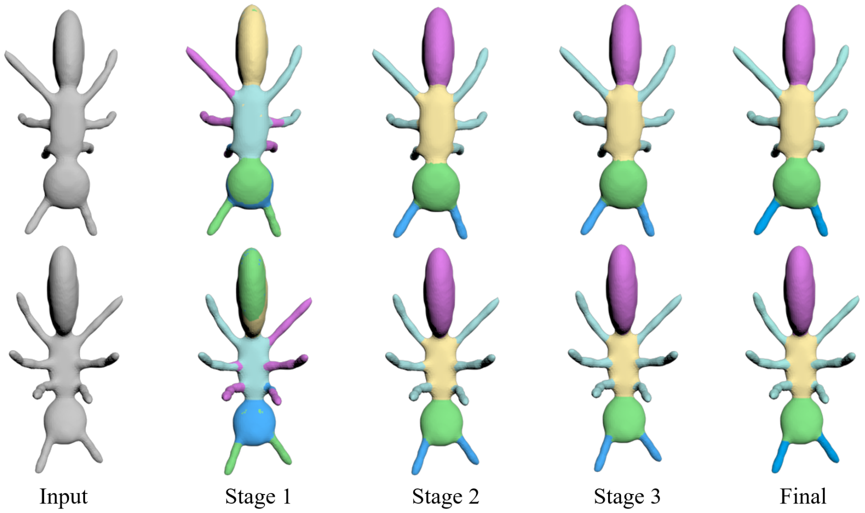


Fig. 5. The corresponding segmentation results after each training stage by using the intermediate-state network to predict and segment a testing shape. The results in the first row represent the top view of the testing shape, and the other row is the bottom view. The final result is the refined one by using graph-cuts.

of the network is the classification results for each face in the 3D shape, representing the corresponding label for each face. To address the total data required for training, we employ a strategy that avoids feeding all faces into the network. This strategy prevents the network from processing redundant and similar faces. Specifically, we utilize uniform sampling to select a representative subset of faces. The sampling ratio is determined based on our processing strategy for partitioned faces. Since each face needs to consider the surrounding faces to establish constraints for consistency, and the local information of the surrounding faces has already been incorporated through these constraints, including the surrounding faces as training samples would lead to duplicated training data. Thus, we set the sampling range to 1/2 to reduce the training burden of the network. Furthermore, our customized loss function for the network is described in Section III-B. After the network, we employ the graph-cut algorithm [41], commonly used in shape segmentation [8], [42], [43], to optimize our segmentation results further. This additional step enhances the accuracy and refinement of the segmentation output.

IV. Experiment

This section showcases the efficacy of our proposed method by subjecting it to a diverse set of datasets. Results obtained from these experiments are then evaluated alongside those generated by current state-of-the-art methods to ascertain the superiority of our approach. Furthermore, the validity and rationality of each component of our method are established through a series of ablation experiments.

A. Experimental Setting

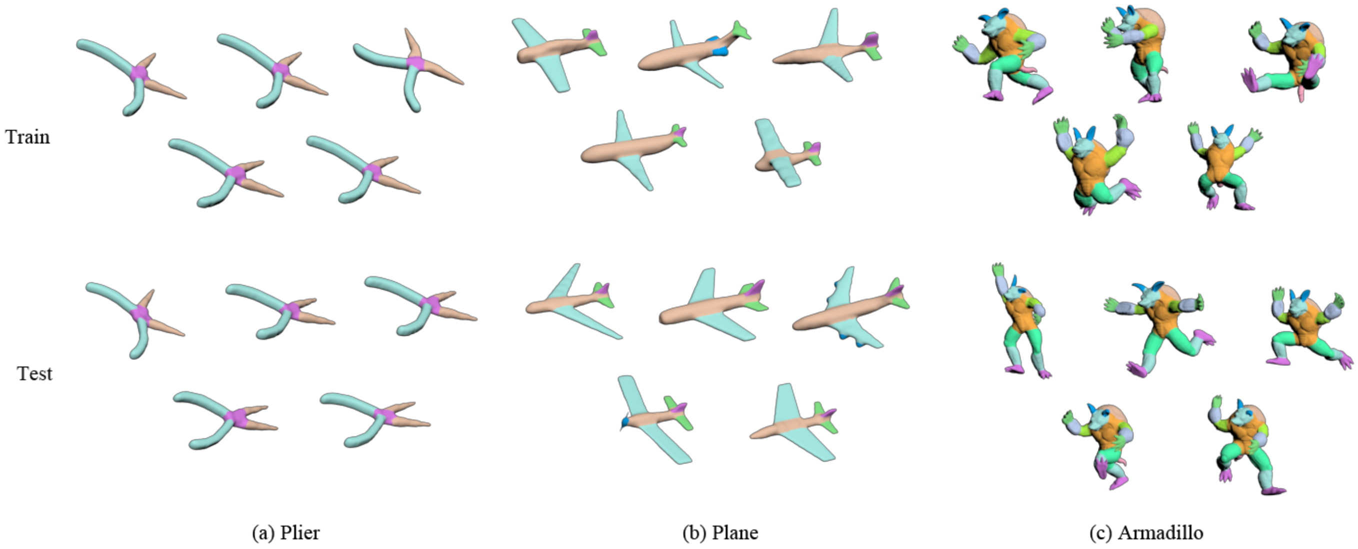
Dataset. In this paper, we conduct experiments using several datasets, including the Princeton Segmentation Benchmark (PSB) dataset [54], COSEG dataset [55], ShapeNetCore dataset [56], and HumanBody dataset [53], to demonstrate the performance of our approach. The PSB and COSEG datasets are two widely used benchmark datasets for evaluating 3D shape segmentation algorithms. The PSB dataset comprises 19 categories, each of which contains 20 models. We removed three categories from the PSB dataset, namely, Bust, Bearing, and Mech, as the models in these categories lack consistent semantic labels. The COSEG dataset is divided into two parts, including a small dataset with eight categories and 190 shapes, and a large dataset with 400 chair shapes, 300 vase shapes, and 200 special-shaped shapes. The ShapeNetCore dataset contains 16 classes, with a total of 4916 models. The HumanBody dataset involves the segmentation task of mannequins and contains a total of 11 categories, where the training set contains 381 models, and the test set contains 18 models. Only the HumanBody dataset maintains the original training test set assignment. For other datasets like PSB and COSEG, we randomly selected 60% of the models as the training set and the rest as the testing set. Figure 6 shows some examples of training sets and testing sets.

Experiment details. Our algorithm is implemented using Python, C++, and Matlab. The network weights are initialized to follow Gaussian distributed random variables with zero mean and 0.001 variance. The Adam optimizer is chosen, and the learning rate is set to 0.001. The experiments are executed on a single NVIDIA GeForce GTX 3090Ti GPU. For each 3D model consisting of 20K-30K faces, the training process takes approximately 30 minutes, and the evaluation time is approximately 50 seconds.

B. Results

The performance evaluation metrics adopted in our method are consistent with those presented in Guo et al. [8]. The following is the mathematical expression of the evaluation index:

$$Accuracy = \sum_{i \in T} C_i \mathbf{g}_t\left(l_i\right) / \sum_{i \in T} C_i, \tag{7}$$



(a) Plier (b) Plane (c) Armadillo

Fig. 6. Some examples of the training dataset and the testing dataset used in our experiments.

TABLE I
THE ACCURACY COMPARISON OF OUR METHOD WITH SEVERAL STATE-OF-THE-ART METHODS, INCLUDING SHAPEBOOST [6], WANG ET AL. [33], GUO ET AL. [8], SHAPEPFCN [25], MESHCNN [44], MESHWALKER [9], AND XU ET AL. [45] ON THE PSB DATASET. THE RESULTS OF MESHCNN AND MESHWALKER ARE OBTAINED ON A LOW-RESOLUTION PSB DATASET, AS SUGGESTED IN THEIR PAPERS, SINCE THE METHODS ARE ONLY SUITABLE FOR LOW-RESOLUTION SHAPES FOR THE HEAVY COMPUTATION BURDENS. OTHER METHOD RESULTS ARE COMPUTED ON THE ORIGINAL PSB DATASET. THE "OURS" REPRESENTS THE ACCURACY OF OUR APPROACH.

Category	ShapeBoost	Wang et al.	Guo et al.	ShapePFCN	MeshCNN	MeshWalker	Xu et al.	Ours
Human	93.20%	55.60%	91.22%	93.80%	74.76%	87.02%	94.08%	**94.44%**
Cup	99.60%	99.60%	99.73%	93.70%	95.86%	99.54%	99.79%	**99.82%**
Glasses	97.20%	-	97.60%	96.30%	93.94%	96.11%	98.69%	**99.04%**
Airplane	96.10%	-	96.67%	92.50%	84.36%	96.20%	97.66%	**98.10%**
Ant	98.80%	-	98.80%	98.90%	91.83%	97.36%	**98.98%**	98.84%
Chair	98.40%	**99.60%**	98.67%	98.10%	84.75%	97.61%	99.35%	99.42%
Octopus	98.40%	-	98.79%	98.10%	98.21%	97.86%	99.34%	**99.71%**
Table	99.30%	**99.60%**	99.55%	99.30%	96.78%	99.33%	99.59%	99.53%
Teddy	98.10%	-	98.24%	96.50%	84.29%	95.57%	**99.08%**	98.99%
Hand	88.70%	-	88.71%	88.70%	68.83%	83.31%	88.61%	**90.67%**
Plier	96.20%	-	96.22%	95.70%	83.69%	92.24%	97.14%	**97.35%**
Fish	95.60%	-	95.64%	95.90%	89.05%	94.58%	97.05%	**97.49%**
Bird	87.90%	-	88.35%	86.30%	68.09%	92.76%	90.39%	**94.28%**
Armadillo	90.10%	-	92.27%	93.30%	50.24%	89.12%	**93.82%**	93.57%
Vase	85.80%	**90.50%**	89.11%	85.70%	68.94%	84.56%	89.31%	90.31%
FourLeg	86.20%	54.30%	87.02%	**89.50%**	68.73%	80.93%	87.42%	89.29%
Average	94.35%	-	94.79%	93.89%	81.40%	92.76%	95.64%	**96.30%**

TABLE II
THE ACCURACY COMPARISON OF OUR METHOD WITH THREE OTHER METHODS, INCLUDING SHAPEBOOST [6], MESHCNN [44], AND SHAPEPFCN [25], ON THE SMALL COSEG DATASET.

Methods	ShapeBoost	MeshCNN	ShapePFCN	Ours
Candelabra	85.50%	83.52%	**95.40%**	94.93%
Chairs	94.80%	92.87%	96.10%	**96.88%**
Fourleg	92.30%	86.19%	90.40%	**92.44%**
Goblets	97.00%	92.62%	97.20%	**97.99%**
Guitars	97.70%	91.34%	98.00%	**98.73%**
Irons	87.20%	81.26%	88.00%	**91.22%**
Lamps	76.30%	83.64%	**93.00%**	87.18%
Vases	86.40%	77.43%	84.80%	**91.25%**
Average	89.65%	86.11%	92.86%	**93.83%**

where T is the triangle set of the testing shapes, and C_i is the area of the triangle i. $\mathbf{g}_t\left(l_i\right)$ equals 1 if the label prediction of l_i is correct and 0 otherwise.

The accuracy of our proposed method on four different datasets, namely PSB, small COSEG, ShapeNetCore, and HumanBody, is presented in Tables I, II, III, and IV, respectively. We achieve high accuracy on these datesets, where 96.30% on the PSB dataset, 93.83% on the small COSEG dataset, 88.9% on the ShapeNetCore, and 93.50% on the HumanBody dataset. We also demonstrate the effectiveness of our proposed method using several examples in Figure 7. Specifically, we compare the segmentation results before and after applying our method to the baseline for a few selected images. The comparison shows that our method significantly improves the precision of the segmentation and produces results that are very close to

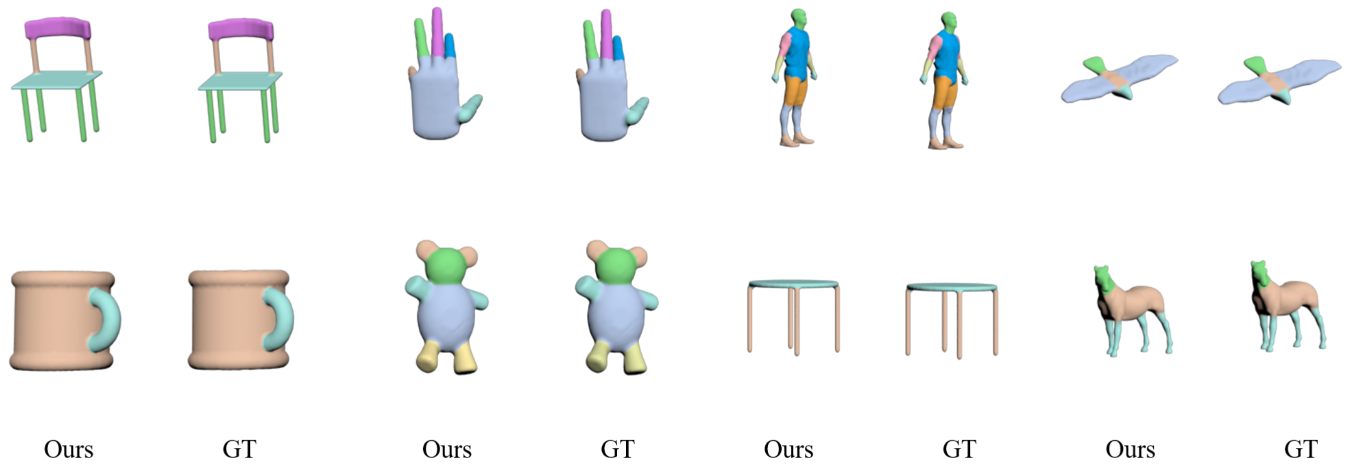


Fig. 7. Our segmentation results on 3D shapes compared with the ground-truth. “GT” represents the ground-truth. “Ours” represents the results of our method.

TABLE III
THE ACCURACY COMPARISON OF OUR METHOD WITH SEVERAL STATE-OF-THE-ART METHODS, INCLUDING SHAPEBOOST [6], KIM'S METHOD [46], AND POINT-BERT [57], ON THE SHAPENETCORE DATASET. THE ”OURS” REPRESENTS THE ACCURACY OF OUR APPROACH.

Category	ShapeBoost	Kim's method	Point-BERT	Ours
Airplane	85.8%	87.4%	84.3%	**91.2%**
Bag	93.1%	91.0%	84.8%	**95.7%**
Cap	85.9%	85.7%	88.0%	**94.4%**
Car	79.5%	80.1%	79.8%	**87.1%**
Chair	70.1%	66.8%	**91.0%**	81.2%
Earphone	81.4%	79.8%	81.7%	**85.5%**
Guitar	89.0%	89.9%	91.6%	**92.6%**
Knife	81.2%	77.1%	**87.9%**	83.3%
Lamp	71.7%	71.6%	**85.2%**	80.1%
Laptop	86.1%	82.7%	95.6%	**95.6%**
Motorbike	77.2%	80.1%	75.6%	**87.3%**
Mug	94.9%	95.1%	94.7%	**95.7%**
Pistol	88.2%	84.1%	84.3%	**91.4%**
Rocket	79.2%	76.9%	63.4%	**84.2%**
Skateboard	91.0%	89.6%	76.3%	**92.6%**
Table	74.5%	77.8%	81.5%	**85.2%**
Average	83.0%	82.9%	84.1%	**88.9%**

TABLE IV
THE ACCURACY COMPARISON OF OUR METHOD WITH SEVERAL STATE-OF-THE-ART METHODS, INCLUDING FC [47], DIFFUSIONNET [48], HODGENET [49], MDGCNN [50], PFCNN [52], AND SUBDIVNET [51] ON THE HUMANBODY DATASET[53]. THE ”OURS” REPRESENTS THE ACCURACY OF OUR APPROACH.

Methods	FC	DiffusionNet	HodgeNet	MDGCNN	PFCNN	SubdivNet	Ours
Accuracy	92.90%	91.50%	85.03%	89.47%	91.79%	93.00%	**93.50%**

the ground truth.

C. Comparison

We present a comprehensive comparison of various segmentation methods used in shape segmentation. The methods include Wang et al. [33], ShapeBoost [6], Guo et al. [8], Kim et al. [46], MeshCNN [44], SubdivNet [51], Point-BERT [57], ShapePFCN [25], and MeshWalker [9]. Guo et al. use a 2D convolutional neural network to process a matrix of geometric feature vectors. ShapeBoost, on the other hand, employs a conditional random fields model to process multiple geometric shape descriptors. Wang et al. and ShapePFCN transform 3D shapes into an ensemble of 2D projections using distinct image-based segmentation techniques. MeshCNN defines the convolution and pooling operations on edges to facilitate shape segmentation. In contrast to MeshCNN, MeshWalker utilizes RNNs to execute random traversals across the mesh surface, thus enabling 3D shape segmentation. SubdivNet constructs a subdivision structure, facilitating the acquisition of a multi-resolution representation for a general mesh. Point-BERT designs a new Transformers pre-training method to help standard Transformers simultaneously learn low-level structural and high-level semantic information.

Across the PSB and COSEG datasets, our algorithm consistently attains the highest average accuracy. Table I presents

This article has been accepted for publication in IEEE Transactions on Multimedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMM.2024.3521674

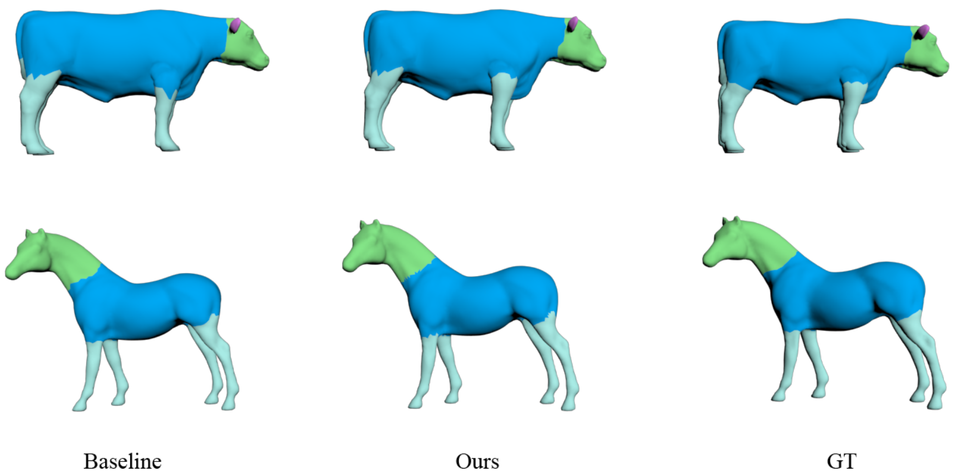


Fig. 8. Qualitative comparison of baseline, our method, and the ground truth. Baseline refers to the simple use of a Transformer network for training and prediction, and our method uses two modules of consistency mining and enhancement on the basis of baseline.

TABLE V
RESULTS OF ABLATION EXPERIMENTS ON TWO COMPONENT STRUCTURES OF OUR ALGORITHM. THE CHECK MARK ✓IN THE TABLE INDICATES WHETHER THE MODULE WAS USED IN THE EXPERIMENT.

Baseline	Consistency mining	Consistency enhancement	Accuracy on the HumanBody dataset	Accuracy on the PSB dataset
Transfomer	-	-	89.55%	93.88%
	✓	-	90.69%	94.80%
	✓	✓	**93.50%**	**96.30%**
MLP	-	-	84.87%	86.49%
	✓	-	85.51%	86.95%
	✓	✓	87.00%	88.39%

the 3D segmentation accuracy of different methods on the PSB dataset. The accuracy of our algorithm exceeds that of other algorithms in nine categories. As showcased in Table II, our algorithm outperformed other methods in six categories on the COSEG dataset.

Table III and Table IV compare the accuracy of our method against ShapeBoost, Kim's method, ShapePFCN, and other methods on the ShapeNetCore and Humanbody datasets. It is manifest that our method excels in terms of segmentation accuracy across two datasets.

D. Ablation Studies

In order to validate the effectiveness of our proposed method, we conduct ablation experiments on two key components of the algorithm on the HumanBody dataset and the PSB dataset. Specifically, we evaluate the performance of the consistency mining module by comparing the segmentation accuracy before and after its use. To better validate the independent impact of the consistency mining module, we exclude the consistency enhancement module from this evaluation. Additionally, we test the performance of the consistency enhancement module and demonstrate the performance improvement achieved using this module based on the consistency mining module. The results of these experiments are presented in Table V. Upon adding the consistency mining module to the baseline, we observe an insignificant improvement in the segmentation performance of the network, as depicted in Table V. This can be attributed to the fact that while consistency mining brings additional information of the 3D shapes to the network, it only provides one-time feature enhancement to the network and cannot be significantly enhanced during iterative training. However, after applying the consistency enhancement strategy, the consistency mining strategy can be continuously updated. The consistency discovered by the network is in a dynamic self-correcting state during the iterative process, leading to more accurate segmentation results.

Figure 8 shows some example results comparing the baseline (only a Transformer Network), our method, and the ground truth. Our proposed segmentation method performs better in precisely identifying the boundaries between partitions of 3D shapes, in contrast to the baseline method. Specifically, the neural network faces a significant challenge in accurately delineating boundaries within the edge regions with high feature similarity. This lack of consistency in information incorporation impedes the precise classification of such regions.

We also make an ablation study about selecting the network we use. We utilize two network frameworks, namely MLP and Transformer, as the backbone of our method, and evaluate the accuracy when using these two networks on the HumanBody dataset and the PSB dataset. As shown in Table V, since the self-attention mechanism of the Transformer facilitates the network to understand the geometric features, its accuracy is higher than that of ordinary MLP.

This article has been accepted for publication in IEEE Transactions on Multimedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMM.2024.3521674

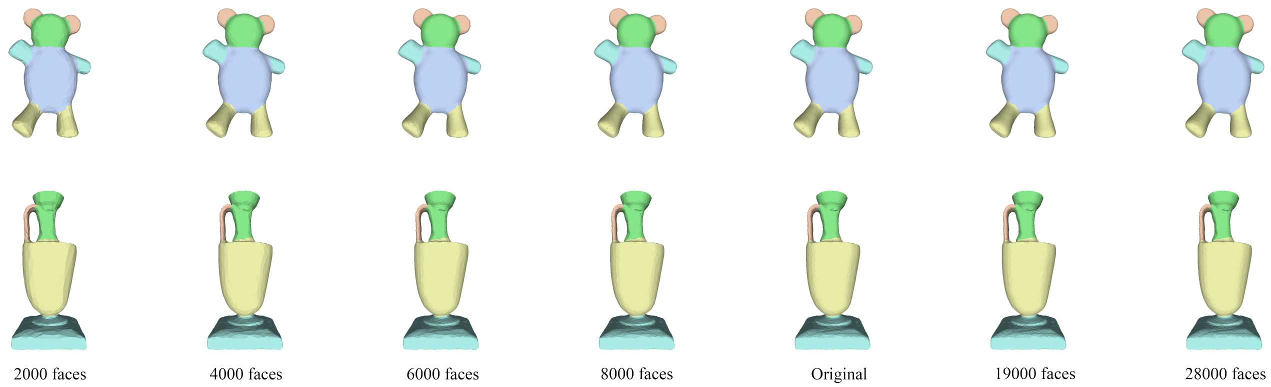


Fig. 9. Some examples of 3D shapes with different resolutions.

TABLE VI
THE RESULTS OF THE ABLATION STUDY ABOUT MESHES' RESOLUTIONS.

Resolution	28000 faces	19000 faces	Origin	8000 faces	6000 faces	4000 faces	2000 faces
Average	95.75%	96.01%	**96.30%**	95.84%	94.44%	93.20%	92.59%

At the same time, to verify the robustness of our algorithm with regard to the resolutions of 3D meshes, we also conducted ablation experiments by controlling the number of faces of 3D shapes. We compare the results of our algorithm for the 3D shape of 28000 faces, 19000 faces, the original 3D shape, the 3D shape of 8000 faces, 4000 faces, and 2000 faces. The subdivided meshes are obtained by applying Loop's subdivision method to the original mesh models. In contrast, the simplified 3D meshes are obtained by simplifying the original ones using the well-known QEM algorithm. The results obtained are shown in Table VI. Some examples of 3D shapes with different resolutions used in this ablation experiment are shown in Figure 9.

The results presented in Table VI indicate that our proposed method is robust to variations in the mesh resolution and can achieve satisfactory results. However, we observe a slight decrease in performance when the resolution of the mesh is too high or too low. Specifically, lower resolutions may result in losing geometric details, affecting the algorithm's performance. Conversely, higher resolutions can enhance the algorithm's ability to capture more geometry details, leading to more effective feature learning. Nevertheless, increased mesh resolution does not constantly improve accuracy, as we observe a downward trend. We attribute this trend to the excessive redundant information that the network accepts. Moreover, our method primarily focuses on optimizing the boundary classification between distinct partitions of the 3D shape. When the resolution of the 3D shape is increased, the proportion of the area and number of edge parts in the entire shape is compressed, thereby partially limiting the benefits of our algorithm and resulting in a slight decline in accuracy.

TABLE VII
THE RESULTS OF THE ABLATION STUDY ABOUT GRAPH-CUTS (GC) OF OUR METHOD ON THE SMALL COSEG DATASET.

Methods	Bi-LSTM without GC	Bi-LSTM with GC
Candelabra	80.43%	**94.93%**
Chairs	75.07%	**96.88%**
Fourleg	86.73%	**92.44%**
Goblets	81.65%	**97.99%**
Guitars	93.04%	**98.73%**
Irons	84.94%	**91.22%**
Lamps	79.75%	**87.18%**
Vases	77.99%	**91.25%**
Average	82.45%	**93.83%**

Finally, we also conducted an ablation experiment about graph-cuts of our method on the small COSEG dataset, the results of which are shown in Table VII. One can see that graph-cuts can ensure our results are consistent with the *minima rule*, thus improving the segmentation performance.

V. LIMITATION AND FUTURE WORK

Our method relies on feature descriptors on 3D shapes, therefore the input 3D shapes must be manifold. Making our method not limited to handling manifold shapes is one of the directions of our future work. In addition, the network architecture used by our method may not be optimal, and we will try different network frameworks in future work to fully exploit our method's advantages. Our method can also inspire related work on point cloud or 2D image segmentation in future work.

VI. CONCLUSION

This paper proposes a 3D shape segmentation method based on consistency mining and enhancement. Unlike approaches that only focus on network architecture, our approach starts by guiding the network to learn the potential consistency of the data itself. Our method first generates a preliminary partition of the 3D shape, and adds a constraint that enables the network to get consistent and accurate labels in each partition. The results of the partitions are gradually optimized during the training iteration of the network. In general, our method has three advantages: 1) Our method mines and enhances the original consistency of data and improves the performance of segmentation by strengthening the network's ability to extract data features. 2) The method dynamically adjusts the region division in mining consistency, improving the network's performance while avoiding overfitting. 3) Since the consistency enhancement strategy of the method will also extract the features of the faces around the target face, the contextual information of the face will also be fully taken into account. Our method is validated on publicly available datasets such as the Princeton Shape Benchmark and COSEG dataset, and experimental results show that our method performs better than existing 3D shape segmentation methods.

ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China (62172356, 61872321), Zhejiang Provincial Natural Science Foundation of China (LY22F020026), the Ningbo Major Special Projects of the "Science and Technology Innovation 2025" (2020Z005, 2020Z007, 2021Z012).

REFERENCES

[1] M. Z. Zia, M. Stark, B. Schiele, and K. Schindler, "Detailed 3D representations for object recognition and modeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2608–2623, 2013.

[2] G. McNeill and S. Vijayakumar, "Hierarchical procrustes matching for shape retrieval," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2006, pp. 885–894.

[3] L. Shapira, A. Shamir, and D. Cohen-Or, "Consistent mesh partitioning and skeletonisation using the shape diameter function," *The Visual Computer*, vol. 24, no. 4, pp. 249–259, 2008.

[4] J. J. Lim, A. Khosla, and A. Torralba, "FPM: Fine pose parts-based model with 3D CAD models," in *European Conference on Computer Vision*. Springer, 2014, pp. 478–493.

[5] B. Pepik, M. Stark, P. Gehler, and B. Schiele, "Multi-view and 3D deformable part models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 11, pp. 2232–2245, 2015.

[6] E. Kalogerakis, A. Hertzmann, and K. Singh, "Learning 3D mesh segmentation and labeling," *ACM Transactions on Graphics*, vol. 29, no. 4, pp. 1–12, 2010.

[7] Z. Xie, K. Xu, L. Liu, and Y. Xiong, "3D shape segmentation and labeling via extreme learning machine," *Computer Graphics Forum*, vol. 33, no. 5, pp. 85–95, 2014.

[8] K. Guo, D. Zou, and X. Chen, "3D mesh labeling via deep convolutional neural networks," *ACM Transactions on Graphics*, vol. 35, no. 1, pp. 1–12, 2015.

[9] A. Lahav and A. Tal, "MeshWalker: Deep mesh understanding by random walks," *ACM Transactions on Graphics*, vol. 39, no. 6, pp. 1–13, 2020.

[10] R. S. Rodrigues, J. F. Morgado, and A. J. Gomes, "Part-based mesh segmentation: a survey," *Computer Graphics Forum*, vol. 37, no. 6, pp. 235–274, 2018.

[11] S. Katz and A. Tal, "Hierarchical mesh decomposition using fuzzy clustering and cuts," *ACM Transactions on Graphics*, vol. 22, no. 3, pp. 954–961, 2003.

[12] S. Shlafman, A. Tal, and S. Katz, "Metamorphosis of polyhedral surfaces using decomposition," *Computer Graphics Forum*, vol. 21, no. 3, pp. 219–228, 2002.

[13] B. Chazelle, D. P. Dobkin, N. Shouraboura, and A. Tal, "Strategies for polyhedral surface decomposition: An experimental study," *Computational Geometry*, vol. 7, no. 5-6, pp. 327–342, 1997.

[14] A. Golovinskiy and T. Funkhouser, "Randomized cuts for 3D mesh analysis," *ACM Transactions on Graphics*, vol. 27, no. 5, pp. 1–12, 2008.

[15] A. P. Mangan and R. T. Whitaker, "Surface segmentation using morphological watersheds," in *Proc. IEEE Visualization*, 1998.

[16] C. Lin, L. Liu, C. Li, L. Kobbelt, B. Wang, S. Xin, and W. Wang, "SEG-MAT: 3D shape segmentation using medial axis transform," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 6, pp. 2430–2444, 2020.

[17] R. Liu and H. Zhang, "Segmentation of 3D meshes through spectral clustering," in *Proceedings of 12th Pacific Conference on Computer Graphics and Applications*. IEEE, 2004, pp. 298–305.

[18] S. Shlafman, A. Tal, and S. Katz, "Metamorphosis of polyhedral surfaces using decomposition," *Computer Graphics Forum*, vol. 21, no. 3, pp. 219–228, 2002.

[19] O. Sidi, O. van Kaick, Y. Kleiman, H. Zhang, and D. Cohen-Or, "Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering," *ACM Transactions on Graphics*, vol. 30, no. 6, p. 1–10, 2011.

[20] Y. Zhuang, M. Zou, N. Carr, and T. Ju, "Anisotropic geodesics for live-wire mesh segmentation," *Computer Graphics Forum*, vol. 33, no. 7, pp. 111–120, 2014.

[21] Y. Zhuang, H. Dou, N. Carr, and T. Ju, "Feature-aligned segmentation using correlation clustering," *Computational Visual Media*, vol. 3, no. 2, pp. 147–160, 2017.

[22] Z. Wu, Y. Wang, R. Shou, B. Chen, and X. Liu, "Unsupervised co-segmentation of 3D shapes via affinity aggregation spectral clustering," *Computers & Graphics*, vol. 37, no. 6, pp. 628–637, 2013.

[23] F. Zhang, Z. Sun, M. Song, X. Lang, and H. Yan, "3D shapes co-segmentation by combining fuzzy c-means with random walks," in *2013 International Conference on Computer-Aided Design and Computer Graphics*, 2013, pp. 16–23.

[24] F. Yu, K. Liu, Y. Zhang, C. Zhu, and K. Xu, "PartNet: A recursive part decomposition network for fine-grained and hierarchical shape segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9491–9500.

[25] E. Kalogerakis, M. Averkiou, S. Maji, and S. Chaudhuri, "3D shape segmentation with projective convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3779–3788.

[26] O. Van Kaick, A. Tagliasacchi, O. Sidi, H. Zhang, D. Cohen-Or, L. Wolf, and G. Hamarneh, "Prior knowledge for part correspondence," *Computer Graphics Forum*, vol. 30, no. 2, pp. 553–562, 2011.

[27] Z. Wu, Y. Wang, R. Shou, B. Chen, and X. Liu, "Unsupervised co-segmentation of 3D shapes via affinity aggregation spectral clustering," *Computers & Graphics*, vol. 37, no. 6, pp. 628–637, 2013.

[28] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 509–522, 2002.

[29] R. Liu, H. Zhang, A. Shamir, and D. Cohen-Or, "A part-aware surface metric for shape analysis," *Computer Graphics Forum*, vol. 28, no. 2, pp. 397–406, 2009.

[30] Z. Shu, C. Qi, S. Xin, C. Hu, L. Wang, Y. Zhang, and L. Liu, "Unsupervised 3D shape segmentation and co-segmentation via deep learning," *Computer Aided Geometric Design*, vol. 43, pp. 39–52, 2016.

[31] L. Shapira, S. Shalom, A. Shamir, D. Cohen-Or, and H. Zhang, "Contextual part analogies in 3D objects," *International Journal of Computer Vision*, vol. 89, no. 2-3, pp. 309–326, 2010.

[32] G. Zhu, Y. Zhou, R. Yao, and H. Zhu, "Cross-class bias rectification for point cloud few-shot segmentation," *IEEE Transactions on Multimedia*, vol. 25, pp. 9175–9188, 2023.

[33] Y. Wang, M. Gong, T. Wang, D. Cohen-Or, H. Zhang, and B. Chen, "Projective analysis for 3D shape segmentation," *ACM Transactions on Graphics*, vol. 32, no. 6, pp. 1–12, 2013.

[34] T. Le, G. Bui, and Y. Duan, "A multi-view recurrent neural network for 3D mesh segmentation," *Computers & Graphics*, vol. 66, pp. 103–112, 2017.

[35] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R. R. Martin, and S.-M. Hu, "PCT: Point cloud transformer," *Computational Visual Media*, vol. 7, pp. 187–199, 2021.

[36] X.-F. Han, Y.-F. Jin, H.-X. Cheng, and G.-Q. Xiao, "Dual transformer for point cloud analysis," *IEEE Transactions on Multimedia*, vol. 25, pp. 5638–5648, 2023.

[37] T. Weng, J. Xiao, F. Yan, and H. Jiang, "Context-aware 3D point cloud semantic segmentation with plane guidance," *IEEE Transactions on Multimedia*, vol. 25, pp. 6653–6664, 2023.

[38] Y. Wu, J. Liu, M. Gong, P. Gong, X. Fan, A. K. Qin, Q. Miao, and W. Ma, “Self-supervised intra-modal and cross-modal contrastive learning for point cloud understanding,” *IEEE Transactions on Multimedia*, vol. 26, pp. 1626–1638, 2024.

[39] M. M. Bronstein and I. Kokkinos, “Scale-invariant heat kernel signatures for non-rigid shape recognition,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 1704–1711.

[40] M. Aubry, U. Schlickewei, and D. Cremers, “The wave kernel signature: A quantum mechanical approach to shape analysis,” in *IEEE International Conference on Computer Vision Workshops*. IEEE, 2011, pp. 1626–1633.

[41] Y. Boykov, O. Veksler, and R. Zabih, “Fast approximate energy minimization via graph cuts,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.

[42] Z. Shu, X. Shen, S. Xin, Q. Chang, J. Feng, L. Kavan, and L. Liu, “Scribble-based 3D shape segmentation via weakly-supervised learning,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 8, pp. 2671–2682, 2019.

[43] D. George, X. Xie, and G. K. Tam, “3D mesh segmentation via multi-branch 1D convolutional neural networks,” *Graphical Models*, vol. 96, pp. 1–10, 2018.

[44] R. Hanocka, A. Hertz, N. Fish, R. Giryes, S. Fleishman, and D. Cohen-Or, “MeshCNN: A network with an edge,” *ACM Transactions on Graphics*, vol. 38, no. 4, pp. 1–12, 2019.

[45] H. Xu, M. Dong, and Z. Zhong, “Directionally convolutional networks for 3D shape segmentation,” in *IEEE International Conference on Computer Vision*, 2017, pp. 2717–2726.

[46] V. G. Kim, W. Li, N. J. Mitra, S. Chaudhuri, S. DiVerdi, and T. Funkhouser, “Learning part-based templates from large collections of 3D shapes,” *ACM Transactions on Graphics*, vol. 32, no. 4, pp. 1–12, 2013.

[47] T. W. Mitchel, V. G. Kim, and M. Kazhdan, “Field convolutions for surface CNNs,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 10 001–10 011.

[48] N. Sharp, S. Attaiki, K. Crane, and M. Ovsjanikov, “DiffusionNet: Discretization agnostic learning on surfaces,” *ACM Transactions on Graphics*, vol. 41, no. 3, pp. 1–16, 2022.

[49] D. Smirnov and J. Solomon, “HodgeNet: Learning spectral geometry on triangle meshes,” *ACM Transactions on Graphics*, vol. 40, no. 4, pp. 1–11, 2021.

[50] A. Poulenard and M. Ovsjanikov, “Multi-directional geodesic neural networks via equivariant convolution,” *ACM Transactions on Graphics*, vol. 37, no. 6, pp. 1–14, 2018.

[51] S. Hu, Z. Liu, M. Guo, J. Cai, J. Huang, T. Mu, and R. R. Martin, “Subdivision-based mesh convolution networks,” *CoRR*, vol. abs/2106.02285, 2021.

[52] Y. Yang, S. Liu, H. Pan, Y. Liu, and X. Tong, “PFCNN: Convolutional neural networks on 3D surfaces using parallel frames,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 13 578–13 587.

[53] H. Maron, M. Galun, N. Aigerman, M. Trope, N. Dym, E. Yumer, V. G. Kim, and Y. Lipman, “Convolutional neural networks on surfaces via seamless toric covers,” *ACM Transactions on Graphics*, vol. 36, no. 4, jul 2017.

[54] X. Chen, A. Golovinskiy, and T. Funkhouser, “A benchmark for 3D mesh segmentation,” *ACM Transactions on Graphics*, vol. 28, no. 3, pp. 1–12, 2009.

[55] Y. Wang, S. Asafi, O. Van Kaick, H. Zhang, D. Cohen-Or, and B. Chen, “Active co-analysis of a set of shapes,” *ACM Transactions on Graphics*, vol. 31, no. 6, pp. 1–10, 2012.

[56] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su *et al.*, “ShapeNet: An information-rich 3D model repository,” *arXiv preprint arXiv:1512.03012*, 2015.

[57] X. Yu, L. Tang, Y. Rao, T. Huang, J. Zhou, and J. Lu, “Point-BERT: Pre-training 3D point cloud transformers with masked point modeling,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 19 313–19 322.

Zhenyu Shu got his Ph.D. degree in 2010 at the Zhejiang University, China. He is now working as a full professor at NingboTech University. His research interests include multimedia processing, computer graphics, digital geometry processing, and machine learning. He has published over 30 papers in international conferences or journals.

Shiyang Li is a graduate student of the College of Computer Science and Technology at Zhejiang University. His research interests include multimedia processing, computer graphics, and machine learning.

Shiqing Xin is an associate professor at the Faculty of School of Computer Science and Technology in Shandong University. He received his Ph.D. degree in applied mathematics at Zhejiang University in 2009. His research interests include computer graphics, computational geometry, and 3D printing.

Ligang Liu received the BSc degree in 1996 and the Ph.D. degree in 2001 from Zhejiang University, China. He is a professor at the University of Science and Technology of China. Between 2001 and 2004, he was at Microsoft Research Asia. Then he was at Zhejiang University during 2004 and 2012. He paid an academic visit to Harvard University during 2009 and 2011. His research interests include geometric processing and image processing. He serves as the associated editors for journals of IEEE Transactions on Visualization and Computer Graphics, IEEE Computer Graphics and Applications, Computer Graphics Forum, Computer Aided Geometric Design, and The Visual Computer. His research works could be found at his research website: http://staff.ustc.edu.cn/lgliu