3D Shape Analysis via Multi-modal Contrastive Learning

Zhenyu Shu[a], Xufei Sun[b,*], Chaoyi Pang[a]

[a]*School of Computer and Data Engineering, NingboTech University, China*
[b]*College of Computer Science and Technology, Zhejiang University, China*

Abstract

In recent years, 3D shape analysis has emerged as a crucial field with applications in various domains, such as multimedia processing, computer graphics, computer vision, and robotics. The ability to understand and interpret 3D shapes is fundamental for tasks like 3D shape segmentation, points of interest detection, shape retrieval, recognition, and generation. However, the complexity of 3D mesh models is a significant barrier that stops the topic from enhancing. Thus, we propose a novel 3D shape analysis framework in this paper by multi-modal contrastive learning techniques. Our framework makes use of the original mesh data and the projected images from various points of view of the mesh model. Those two modals contribute to providing more precise features with the help of our within-modal and cross-modal losses, which respectively calculate the distances of feature vectors within the mesh model and between feature vectors of mesh and image. Our framework is tested on downstream tasks, including 3D shape segmentation and points of interest detection, and outperforms most state-of-the-art methods on public datasets.

Keywords: 3D shape segmentation, Point of interest, Deep neural network, Contrastive learning

1. Introduction

3D shape analysis has become a crucial research area with far-reaching applications in diverse fields in recent years, including multimedia processing, computer graphics, computer vision, virtual reality, and robotics. Understanding and extracting meaningful information from 3D shapes are fundamental tasks that underpin various applications, such as shape retrieval, recognition, generation, or even industrial applications such as Zhuang et al. (2025); Liu et al. (2022). However, the inherent complexity and high-dimensional nature of 3D shapes present significant challenges for traditional analysis methods, demanding innovative and effective approaches to tackle these complexities.

Most existing 3D shape analysis methods heavily depend on the geometric similarity between faces. Therefore, extracting robust and effective geometric features for each face is the key to further improving the performance of 3D shape analysis frameworks. Earlier methods directly employ existing 3D shape feature descriptors, such as Shape Diameter Functions Shapira et al. (2008) (SDF), Average Geodesic Distance Shapira et al. (2010) (AGD), and Gaussian Curvature Gal and Cohen-Or (2006) (GC) to describe the geometric feature of each face. However, one single feature descriptor can only describe the features of faces in one aspect, which greatly prevents the performance of 3D shape analysis algorithms from further improvement. Hence, later methods Kalogerakis et al. (2010); Guo et al. (2015); Shu et al. (2016) tend to combine multiple feature descriptors together and hope to achieve better performance than using one only.

With the fast development of machine learning techniques, more and more 3D shape analysis methods utilize machine learning approaches, especially deep learning ways, to obtain more reliable geometric features. Those methods can be classified into two major categories. The first category of methods mainly exploits a deep neural network to map the existing low-level geometric features to high-level ones, such as Guo et al.

*Corresponding author
Email addresses: shuzhenyu@nit.zju.edu.cn (Zhenyu Shu), xufeisun_paper@163.com (Xufei Sun)

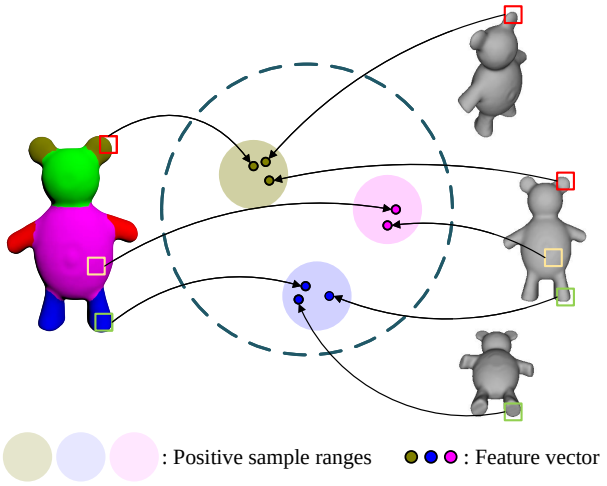

Figure 1: Taking 3D shape segmentation as an example, by adding multi-modal contrastive learning, our method trains an encoder which can make use of feature vectors extracted from the original 3D shapes (shown on the left) and corresponding projected 2D images from various angles (shown on the right). The labels are represented by different colors in this figure. By making feature vectors of the same label closer, and those of different labels farther away, the encoder is trained to be capable of learning more effective features used in downstream analysis tasks.

(2015); Shu et al. (2024a,b, 2025a). Usually, this kind of method depends on a large amount of high-quality labeled training 3D shapes to ensure satisfactory results. However, manually labeling each face of 3D shapes is widely deemed an extremely tedious and expensive task. The other category of methods Wang et al. (2013); Xie et al. (2015); Kalogerakis et al. (2017) try to project 3D shapes into several 2D views and convert the task of 3D shape analysis into 2D image analysis. Benefiting from transferring prior knowledge learning from existing 2D image datasets, this category of methods shows their superior performance over other approaches. However, it suffers from the occlusions occurring during projections, which stopped its performance from further improvement.

In this paper, we propose a novel 3D shape analysis method, which builds the contrastive learning framework between the high-level geometric features and the features of projected 2D views, to fully take advantage of the above two major categories of methods. Taking 3D shape segmentation as an example, we present the motivation of our framework in Figure 1. Benefiting from contrastive learning, our framework directly learns more robust and effective features usable in downstream tasks, including 3D shape segmentation and points of interest detection, which shows its great advantage over existing methods on public datasets.

Our contributions are two-fold:

- We propose a novel 3D shape analysis framework by using the contrastive learning technique. With our novel learning framework, we can directly learn robust and effective features which can be applied to downstream tasks and obtain satisfactory results.

- Different from existing methods, our framework utilizes the existing hand-crafted 3D geometric features and prior knowledge transferred from the 2D image domain together, so that the shortage of the two categories of existing methods can remedy each other and better performance is achieved than using any one of them only.

The remaining parts of this paper are organized as follows. First, we introduce related work in Section 2. Second, we describe the details of our framework in Section 3. Third, Section 4 shows the performance of our framework on downstream tasks and compares it to state-of-the-art methods on public benchmarks. Fourth, the limitations and future work of our framework are explained in Section 5. Finally, we conclude our paper in Section 6.

2. Related Work

In the field of multimedia processing, one of the most important research topics is shape analysis. In the following, we mainly review the methods of 3D shape analysis based on surface meshes.

2.1. Contrastive learning

As a large number of labeled meshes are needed for supervised methods to train algorithms, people tend to use unsupervised methods which do not need the help of tons of data. Contrastive learning, which is a self-supervised approach, needs to learn feature representation from data and use it in downstream tasks. In contrastive learning methods, the representation learning model is built by constructing similar and dissimilar instances and the process of pushing similar instances closer, while pulling dissimilar instances farther away in the projection space. Contrastive learning has been widely used in the field of computer vision. InstDisc Wu et al. (2018) uses a memory bank to form a dictionary, saving all features of images originated from data augmentation, which serves as the positive samples. All other images and their augmented version serve as negative samples. CMC Tian et al. (2019) is the first to introduce contrastive learning in multi-view models, which uses four types of images, including original image, depth information, surface normal, and segmentation image, to describe the same scene and form positive samples for each other, and all others are negative. MoCo He et al. (2019) introduces a new concept, momentum encoder, into contrastive learning and uses a queue to take the place of a memory bank in order to solve the problem of storage when the feature descriptor of images is to form a huge dictionary. SimCLR Chen et al. (2020b) forms a widely-used pipeline which uses two augmented data as positive samples and an encoder whose weights are shared to extract feature vectors. Projection heads using MLP layers are also used to project feature vectors into another feature space. BYOL Grill et al. (2020) starts a new stage of contrastive learning, which only uses positive samples to train the model, adding another projection head to prevent the model from collapsing.

2.2. 3D shape segmentation

Early 3D shape segmentation approaches focus on utilizing hand-crafted feature descriptors to segment 3D shapes. Naturally, the faces with the same label in a 3D shape should have similar geometric features. Thus, many researchers have designed various feature descriptors to map all faces into feature space and then applied clustering algorithms to divide them into several classes for segmentation. AGD, calculated by the average geodesic distance between each vertex and all other vertices, represents the global position information of 3D shapes. SDF measures the diameter of the local shape of the face to identify the thin part and the fat part of the 3D shape. GC describes the bending degree of each vertex in the 3D shape. Extensive results show that utilizing these feature descriptors and others obtain satisfactory results in 3D shape segmentation. To further improve the performance of segmentation, Huang et al. (2011); Sidi et al. (2011); Hu et al. (2012); Kim et al. (2013); Kaick et al. (2014) combine multiple feature descriptors to extract the feature vectors of 3D shapes from multiple aspects.

Benefiting from the rapid development of 3D shape repositories and machine learning techniques, especially deep learning, a growing number of researchers focus on supervised learning-based segmentation methods. Compared with the traditional and unsupervised methods, the supervised learning-based methods can learn the mapping relationship from feature vectors to labels through prior knowledge, so that they often achieve superior performance.

The first supervised learning-based method introduced in 3D shape segmentation is the work of Kalogerakis et al. (2010). They design an objective function with learnable parameters based on the CRF model. The objective function is optimized by utilizing the manually labeled shapes. Similar to Kalogerakis et al. (2010), Kaick et al. (2014) propose a novel shape segmentation approach, which utilizes the knowledge by analyzing geometric similarity between the matched shapes. Some researchers use supervised learning methods on multiple geometric feature descriptors to segment shapes. Xie et al. (2014) propose a fast segmentation method on the mesh by using Extreme Learning Machine. Guo et al. (2015) pioneer the deep convolution neural networks in 3D shape segmentation by transforming multiple geometric feature descriptors into a two-dimensional matrix. Liu et al. (2021) use the Point Context Encoding method, which enables the method to capture semantic contexts of a point cloud and adaptively highlight intermediate feature

maps. Chen et al. (2021) use a gated graph attention network to solve the problem of treating different neighbor points equally in previous methods. Besides feature-based methods, view-based methods were also applied for segmentation by building up the connection between 3D shapes and their 2D projection collections. Wang et al. (2013) label each projection through the knowledge learned from the labeled projections and then projects back to the labels on the mesh. Kalogerakis et al. (2017) apply the image-based Fully Convolutional Network to label the projections and obtain excellent segmentation results. Le et al. (2017) suggest a method that treats multiple 2D projections of the 3D shape as the format of sequence and employs the RNN for segmentation. MeshWalker Lahav and Tal (2020) also uses RNN to segment 3D shapes, but their sequences were obtained by random walking on the mesh surface.

2.3. Points of interest detection

Points of interest (POIs), defined as distinctive points on the surface of 3D shapes, play a crucial role in 3D shape analysis tasks. Detecting these POIs serves various purposes, such as facilitating shape-based searches across distinct regions Shilane and Funkhouser (2007) or selecting the most informative views of a given 3D model Leifman et al. (2012).

Initially, researchers identified 3D POIs by analyzing multiple 2D projected views, such as Guy and Medioni (1997); Yee et al. (2005); Mantiuk et al. (2003). However, in the past decade, the focus has shifted toward directly detecting POIs on the input polygonal surface. This involves assessing the saliency based on geometric properties in the local neighborhood. Depending on the size of this neighborhood, existing methods can be classified into two categories.

The first category consists of algorithms that measure saliency on a local scale. For instance, Koch and Ullman (1987) proposed that salient regions should exhibit distinctiveness from their immediate surroundings. Lee et al. (2005) defined scale-dependent saliency using a center-surround operator on Gaussian-weighted mean curvatures. Gal and Cohen-Or (2006) introduced salient geometric features that represent the geometry of local surface regions by combining low-level features into a high-level representation. Spectral analysis techniques Hou and Zhang (2007) have also been explored, involving the transformation of spectral residuals from the spectral domain back to the spatial domain for this purpose. The second category of methods Cheng et al. (2015); Duan et al. (2011) approaches saliency assessment differently. These methods often require evaluating global contrast differences and spatial coherence. The central idea is to establish a measurement that highlights the visually striking regions on a global scale.

From the above, it is evident that previous approaches to 3D shape analysis faced certain challenges. Some approaches involved utilizing feature descriptors to analyze 3D shapes, but this heavily relied on manually annotated data for each downstream task. Consequently, obtaining satisfactory results often became challenging when high-quality labeled data was unavailable. Another set of methods Shu et al. (2025b) attempted to address 3D shape analysis by converting 3D shapes into 2D images through projections and then applying image analysis techniques. However, relying solely on projections made it difficult to overcome occlusion issues, significantly impacting the efficiency and effectiveness of the analysis. To address these limitations, we propose a novel approach that leverages multi-modal contrastive learning. By effectively combining 3D shapes and 2D projected images, we exploit the correspondence and similarity between the two modalities to extract more accurate and effective feature vectors. This approach aims to enhance various downstream tasks related to 3D shape analysis, offering improved performance and better outcomes.

3. Our framework

In this section, we will introduce the details of our method. As shown in Figure 2, our framework consists of two branches. The first is the 3D mesh branch, which uses the original model to extract feature vectors, and the other is the image branch, which projects the 3D mesh to a series of 2D images in different angles and generates feature vectors via pre-trained feature extractors. Finally, we conduct contrastive learning between feature vectors of images and feature vectors of 3D meshes in order to train an encoder for 3D meshes and use the encoder to analyze shapes in the testing phase.

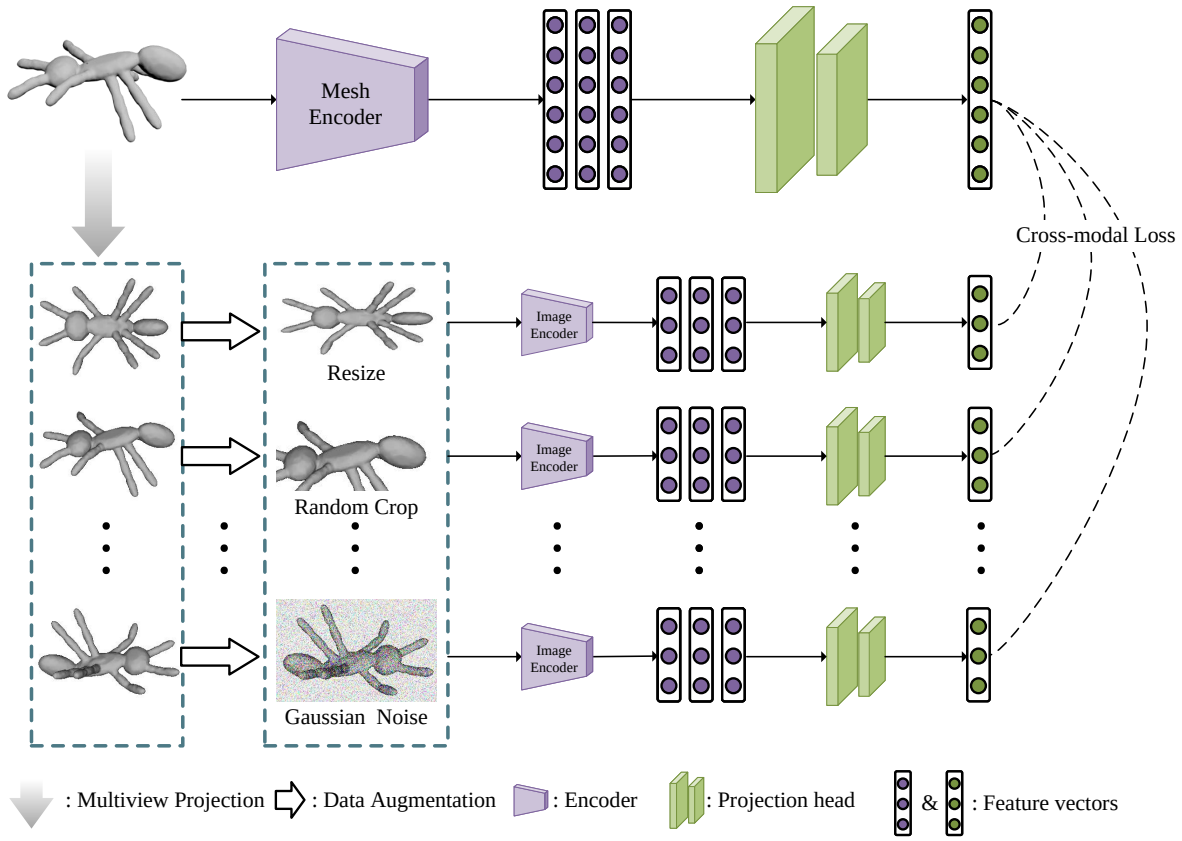

Figure 2: Given a 3D mesh model, we are working on two branches. The upper part, the mesh branch, extracts the feature vectors from the original mesh and projects them into the feature space via the projection head. The lower part, the image branch, gets the projected images and applies data augmentation to them. By extracting feature vectors from the pre-trained encoder, our method projects them into feature space through projection heads. Feature vectors are compared using within and cross-modal loss in the feature space, and therefore, our approach trains the encoder of the mesh branch aiming at making similar vectors closer and different ones farther away.

3.1. Preliminaries

Notations. In each epoch, we have several 3D models $M = \{M_1, M_2, \ldots\ldots, M_n\}$, where $M_i = \{V, E, F\}$ representing the set of vertices, edges, and faces of the mesh model, each of which is projected into 2D images $I_i = \{I_i^1, I_i^2, \ldots\ldots, I_i^m\}$ from several different view-points. We are supposed to train a feature extractor for 3D models $f_m(\cdot)$, with the help of image feature extractor $f_i(\cdot)$. $g_m(\cdot)$ and $g_i(\cdot)$ are projection heads for mesh models and images respectively.

Preliminaries related to contrastive learning. The main goal of contrastive learning is to train an encoder f which can extract the representation, that is, the feature vectors of input samples, and can be used to adapt to other downstream tasks. In order to train the encoder, positive and negative pairs are needed to calculate the contrastive loss. Specifically, positive pairs have similar features, for example, selected from the same category, or generated from the same image, whereas negative pairs are typically different in shapes, colors, and other ways, for example, different types of images. We train the network to differentiate feature vectors of negative pairs and gather those of positive pairs using contrastive loss.

However, the feature vectors generated from the encoder cannot be directly used in contrastive losses, for the reason that contrastive learning is not always suitable for downstream tasks. Therefore, in the training phase, we add another network called projection head g consisting of MLP with one hidden layer, which

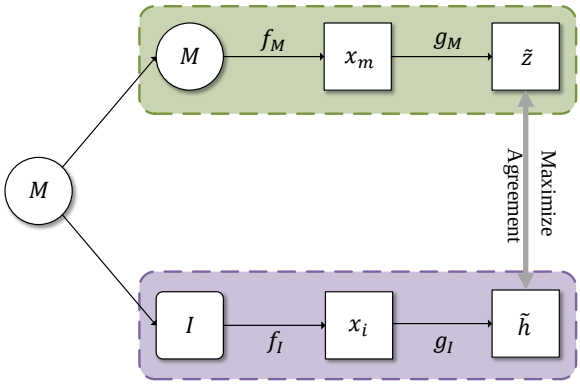

Figure 3: A simplified pipeline of the contrastive learning method which is used in the segmentation method. M denotes the original mesh model and I denotes the image projected from M. f_M and f_I are encoders which extract feature vectors from mesh and image respectively. g_M and g_I are projection heads constructed with MLPs used to project feature vectors into feature spaces where we compare and calculate losses.

maps the vectors to another feature space to calculate the contrastive loss. Projection heads are usually simple networks consisting of MLP layers.

After training the encoder, the projection head, as well as positive and negative pairs, are neglected, and a softmax layer is added to predict the downstream task labels. Also, for most methods, negative pairs are not used for the great difference in the number of positive and negative pairs. Figure 3 shows the simplified pipeline of contrastive learning used in our method.

In the following, we introduce the contrastive learning settings used to conduct 3D mesh analyzing tasks. Section 3.2 and Section 3.3 discuss the mesh and image branches, respectively. Section 3.4 concludes the overall algorithm in total.

3.2. Mesh branch

Inspired by contrastive learning algorithms used on images He et al. (2020); Chen et al. (2020a), we train the encoder based on the idea that the feature vectors of faces with the same label should be closer in the feature space, and at the same time, those of faces with different labels should be separated. Therefore, we treat faces with the same labels as positive pairs and minimize the contrastive loss between them.

Given a mesh model $M_i = \{V, E, F\}$, we use the feature extractor $f_m(\cdot)$ to map the faces in M_i into feature vectors z. Then we project the feature vectors into another feature space where the contrastive loss is applied by using the projection head $g_m(\cdot)$. We denote the result $\tilde{z}$ where $\tilde{z} = g_m(f_m(M_i))$. The contrastive loss function between faces in the 3D shape is:

$$L_{mesh} = -\frac{1}{N}\sum_{i=1}^{N}\frac{1}{N_{y_i}}\sum_{j=1}^{N}[y_i = y_j]\log\left(\frac{e_{ij}}{\sum_{k=1}^{N} e_{ik}}\right), \tag{1}$$

where N is the number of total faces in the 3D mesh model, N_{y_i} is the number of faces whose label equals the label of face i. $e_{ij} = \exp(f_i \cdot f_j/\tau)$ where f_i and f_j are feature vectors of face i and j respectively and τ is the temperature parameter.

Furthermore, to extract rich geometric feature information on 3D shapes, we use multiple feature descriptors to calculate the feature vector on each face as input. Instead of stacking more feature descriptors, we only select five feature descriptors that perform well in previous studies. These feature descriptors include SDF, GC, AGD, Wavelet Kernel Signature Aubry et al. (2011) (WKS), and Scale Invariant Heat Kernel Signatures Bronstein and Kokkinos (2010) (SIHKS). SDF, GC, and AGD are used to measure the inherent geometric features of 3D shapes, whereas WKS and SIHKS are feature descriptors based on spectral shape analysis. SDF, GC, and AGD are one-dimensional, while WKS and SIHKS are 100-and- 19-dimensional respectively. These vectors are concatenated into a 122-dimensional feature vector.

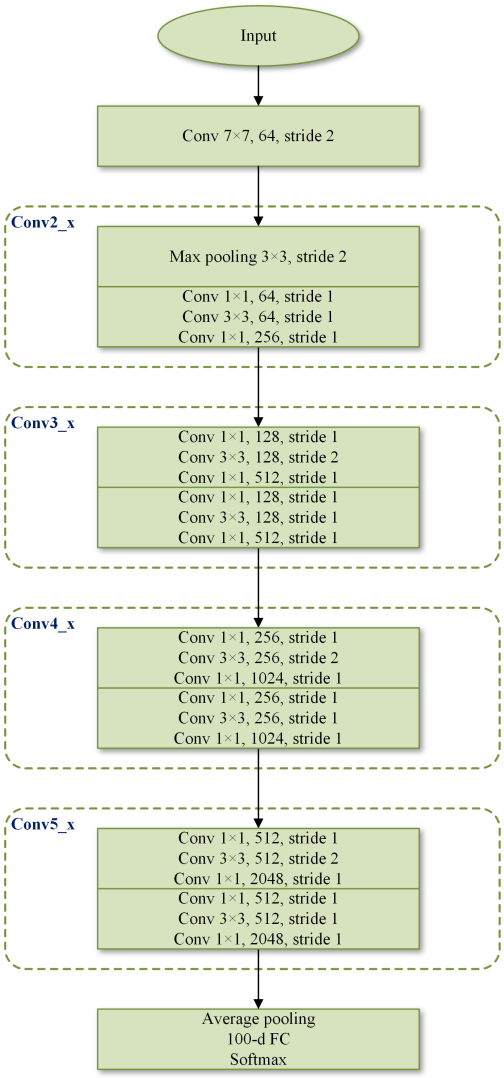

Figure 4: Architectures of ResNet. It is naturally divided into 4 parts, namely conv2_x, conv3_x, conv4_x, and conv5_x. In our method, we take the middle two parts and use them as the encoder of images in order to extract the feature vectors.

3.3. Image branch

In order to generate more positive pairs for each face and improve the performance, we make use of the images projected from 3D shapes and build an image branch to extract features and exploit them in our algorithm.

We place a set of cameras at different angles and positions relative to the 3D model. The details about the setting of cameras can be found in section 4.2. Each camera can form a 2D image containing rendering information from the mesh and also depth information. Although there might be occlusion, we can capture information for all faces to the most extent. Labels are also allocated to the images in the projection process. For every single pixel in the image, if only one face is related, we directly take the label of the face as its label. Otherwise, if more than one face is included in the pixel, we make votes to decide the labels referring to the area of each label.

Also, as each image contains different parts of the original mesh, we use each projected image to be contrastive samples against 3D mesh models instead of calculating contrastive loss among images. Data augmentation is also applied in order to learn more features from images, such as random crop, resize,

Algorithm 1 : 3D Shape Analysis via Contrastive Learning

Inputs: Training 3D shapes and human-assigned labels for all faces
Outputs: Predicted label for each face on test 3D shapes
Training process:
Mesh branch:
Step 1: Compute features for each face in training 3D shapes using feature descriptors, including AGD, SDF, GC, SIHKS, and WKS.Concatenate those features into high-dimensional vectors;
Step 2: Project the feature vectors to the feature space via f_m and g_m and get $\tilde{z}$;
Image branch:
Step 1: Obtain 2D images using the projection method through different angles;
Step 2: Extract feature vectors from the augmented images, project them into feature space via g_i, and get h_i for each image;
Step 3: Train our network and minimize the contrastive loss between h and $\tilde{z}$;
Testing process:
Step 1: Compute feature vectors for each face in testing shapes;
Step 2: Use the trained encoder f_m to extract feature vectors from the input feature vectors;
Step 3: Add different layers to meet the requirements of each downstream task.

rotate, Gaussian noise, and color distortion. Those augmentation methods are randomly selected and applied to each image. To extract feature vectors, we opt for the well-known ResNet He et al. (2015) which is commonly used and also effective as the encoder for images (f_I). As shown in Figure 4, ResNet is divided into 4 parts. The first part of which extracts typically lower-level features such as colors, while the last part is mainly designed to match the needs of downstream tasks. The middle two parts are chosen as the encoder for images in our method. Like the mesh branch, we also use projection head g_i to project the feature vectors to the feature space mentioned above and calculate contrastive loss between each image and the original mesh.

For each image, we use the same weight due to their equal importance, that is, the total contrastive loss between mesh and images is:

$$L_{image} = -\frac{1}{N}\sum_{i=1}^{N}\frac{1}{I}\sum_{m=1}^{I}\frac{1}{M_{y_i}^m}\sum_{j=1}^{M}[y_i = y_{m_i}]\log\left(\frac{e_{im_j}}{\sum_{k=1}^{M} e_{im_k}}\right), \tag{2}$$

where I denotes the number of images projected from the mesh, M denotes the number of pixels in each image, and $M_{y_i}^m$ denotes the number of pixels in image m whose label is equal to the label of face i.

Finally, we use the combination of the two contrastive losses mentioned above, representing mesh and image respectively, as the resultant loss function, which is: $L = L_{mesh} + L_{image}$.

3.4. Algorithm

Our algorithm is trained and tested on each category of 3D shapes, which can be summarized as Algorithm 1.

Throughout the entire contrastive learning training process, both the encoder and projection head in the two branches participate in training. It should be noted that we fine-tune the ResNet encoder after initializing it with pretrained weights instead of freezing it. This design ensures that all parts of the network, even those pretrained on other datasets, are fully optimized to capture the unique characteristics of our 3D shape analysis task.

3.5. Downstream tasks

In this section, we primarily focus on the specific applications of our framework in downstream tasks of 3D shape segmentation and 3D feature point detection. Specifically, by fully leveraging cross-modal contrastive

learning techniques, we design a two-stage processing to apply the network architecture to downstream tasks.

3D shape segmentation. In the context of 3D shape segmentation, we simplify it as a process of classifying each face of the 3D shape. Therefore, in our 3D shape segmentation task, we first utilize the existing 122-dimensional feature vector as input to our proposed framework. In order to obtain more effective features for each face of the 3D shape, we train the network using two complementary loss functions: an intra-modal loss and a cross-modal loss. Subsequently, these refined features are treated as input to a classification network containing two hidden layers, each having 32 neurons and one softmax layer for the final prediction of segmentation labels. Finally, we apply graph-cut Boykov et al. (2001) to smooth the boundary and enhance the effectiveness of 3D shape segmentation results. Graph-cut has three core steps in 3D mesh segmentation: First, each face is treated as a graph node, and edges are constructed between nodes based on adjacency relationships. The unary cost for each node is derived from the softmax outputs of our classification network, representing the cost or uncertainty of assigning a particular label to that face. Second, pairwise cost is generated by computing geometric differences (e.g., discrepancies in normal vectors or spatial proximity) between adjacent faces, leveraging geometric consistency to optimize label propagation. Lastly, the overall energy function, combining the unary and pairwise costs, is then minimized using a minimum cut/maximum flow algorithm. This process yields a segmentation that balances data fidelity with boundary smoothness, resulting in more coherent and visually pleasing segmentation outcomes. Our experiments show that integrating this graph cut step improves segmentation accuracy by approximately 3-4 percentage points on average.

3D points of interest detection. Similar to 3D shape segmentation, we treat 3D point of interest extraction as a binary classification problem for each vertex, determining whether it is a point of interest or not. Therefore, we use similar methods to obtain feature vectors for each vertex. By utilizing these feature vectors, we calculate the probability distribution of each vertex being classified as a point of interest. Finally, employing a method similar to Shu et al. (2023), we obtain the predictions for the points of interest based on this probability distribution and use the Density Peak Clustering method to sort out POIs.

After finishing contrastive learning training, the algorithm can be adapted to various downstream tasks by incorporating additional classification layers. Notably, the network employs an extra supervised training phase to facilitate this adaptation. In our experiments, we partitioned the 3D model dataset into training and testing sets with a 6:4 ratio, maintaining consistency with prior methodologies.

4. Experiments

In this section, we present the experimental results of our framework and compare them with current state-of-the-art approaches. Then, we set up several ablation experiments to verify the rationality of our approach.

4.1. Dataset

To validate the effectiveness of our framework in handling different downstream tasks, we conducted extensive experiments on a substantial amount of publicly available datasets. For 3D shape segmentation tasks, we employed datasets such as PSB, COSEG, and Human Body. For 3D point of interest detection tasks, we used datasets including SHREC 2007 and 2011.

3D shape segmentation. In 3D shape segmentation tasks, we employ the Princeton Segmentation Benchmark Chen et al. (2009) (PSB), the COSEG benchmark Wang et al. (2012), and the Human Body Dataset proposed by Maron et al. (2017) in the experiments to evaluate our algorithm. PSB and COSEG are the two most popular datasets for benchmarking 3D manifold shape segmentation algorithms. The PSB dataset contains 19 categories, with 20 models for each category. We remove the three categories of bust, bearing, and mech because the models in these categories lack consistent semantic labels. The small dataset of the COSEG contains shapes for eight classes, and the large dataset consists of three classes. The Human Body Dataset is a newly constructed and recently popular dataset formed by 381 training models and 18 testing models. The division of the training and testing sets for PSB is referenced from Guo et al. (2015). We take 12 models as the training sets for each category and the rest as the validation sets.

Table 1: The accuracy comparison of our method against three supervised methods, including ShapePFCN Kalogerakis et al. (2017), MeshCNN Hanocka et al. (2019), and MeshWalker Lahav and Tal (2020) on the PSB dataset.

Category	ShapePFCN	MeshCNN	MeshWalker	Ours
Cup	93.70%	95.86%	**99.54%**	98.52%
Table	99.30%	96.78%	**99.33%**	99.10%
Teddy	96.50%	84.29%	95.57%	**97.25%**
Bird	86.30%	68.09%	**92.76%**	90.61%
Hand	**88.70%**	68.83%	83.31%	86.67%
Fish	95.90%	89.05%	94.58%	**96.09%**
Human	93.80%	74.76%	87.02%	**93.88%**
Glasses	96.30%	93.94%	96.11%	**96.67%**
Airplane	92.50%	84.36%	96.20%	**96.81%**
Ant	**98.90%**	91.83%	97.36%	98.73%
Chair	**98.10%**	84.75%	97.61%	97.95%
Octopus	98.10%	98.21%	97.86%	**98.56%**
Plier	95.70%	83.69%	92.24%	**95.79%**
Armadillo	93.30%	50.24%	89.12%	**94.01%**
Vase	85.70%	68.94%	84.56%	**88.12%**
FourLeg	89.50%	68.73%	80.93%	**90.03%**
Average	93.89%	81.40%	92.76%	**94.94%**

Table 2: The accuracy of segmentation for each category of 3D shapes in the small COSEG dataset compared with three other methods, including ShapeBoost Kalogerakis et al. (2010), MeshCNN Hanocka et al. (2019), and ShapePFCN Kalogerakis et al. (2017).

Category	ShapeBoost	MeshCNN	ShapePFCN	Ours
Candelabra	85.50%	83.52%	95.40%	**96.20%**
Chairs	94.80%	92.87%	**96.10%**	95.84%
Fourleg	92.30%	86.19%	90.40%	**92.58%**
Goblets	97.00%	92.62%	97.20%	**97.72%**
Guitars	97.70%	91.34%	98.00%	**98.93%**
Irons	87.20%	81.26%	88.00%	**88.31%**
Lamps	76.30%	83.64%	**93.00%**	90.49%
Vases	86.40%	77.43%	84.80%	**88.03%**
Average	89.65%	86.11%	92.86%	**93.51%**

POI detection. In POI detection tasks, we test the ability of our method on the SHREC 2007 and 2011 datasets, which is an open dataset originally used for 3D shape classification and retrieval. The SHREC 2007 dataset contains 20 categories of 3D shapes, each with 20 3D shapes, while the SHREC 2011 dataset contains 30 categories of shapes. We developed a small visual tool and manually marked POIs for each 3D shape in the dataset. 10 shapes are randomly selected from each category as the training set, and the rest are as the testing set.

4.2. Experiment details.

We implement our algorithm in Python and Matlab. In our network, the initial weights are set to variables subject to a Gaussian distribution with a variance of 0.001 and a mean of zero. The optimizer is Adam, with a learning rate of 0.001. Our algorithm runs on a single NVIDIA GeForce RTX 3090 GPU. With the consumption of shape preprocessing, for each model with 20K-30K faces, our algorithm needs 10 minutes for training (including contrastive learning and subsequent task-specific training) and 30 seconds for evaluation.

For the projection phase, we employ a total of 26 virtual cameras for each shape. These cameras are positioned at various locations, with their distances from the shape's bounding sphere radius. Initially, we randomly determine an azimuth direction and assign the first camera to that position. Along the equator, we place an additional 7 virtual cameras at 45-degree intervals. When the elevation angle reaches 45 and -45 degrees, we add 16 more virtual cameras at the same azimuth positions as the first 8 cameras. Finally, we position the last 2 cameras on the poles. Furthermore, to expand the training dataset, each camera is rotated 4 times at 90-degree intervals.

Table 3: The accuracy of segmentation on the Human Body Dataset compared with six other methods, including Maron et al. Maron et al. (2017), DiffusionNet Sharp et al. (2020), Field Convolutions Mitchel et al. (2021), HodgeNet Smirnov and Solomon (2021), MDGCNN Poulenard and Ovsjanikov (2018) and PFCNN Yang et al. (2018).

Method	Accuracy	Method	Accuracy
Maron et al.	88%	DiffusionNet	90.80%
Field Convolutions	92.90%	HodgeNet	85.03%
MDGCNN	92.90%	PFCNN	85.03%
Ours	**93.02%**		

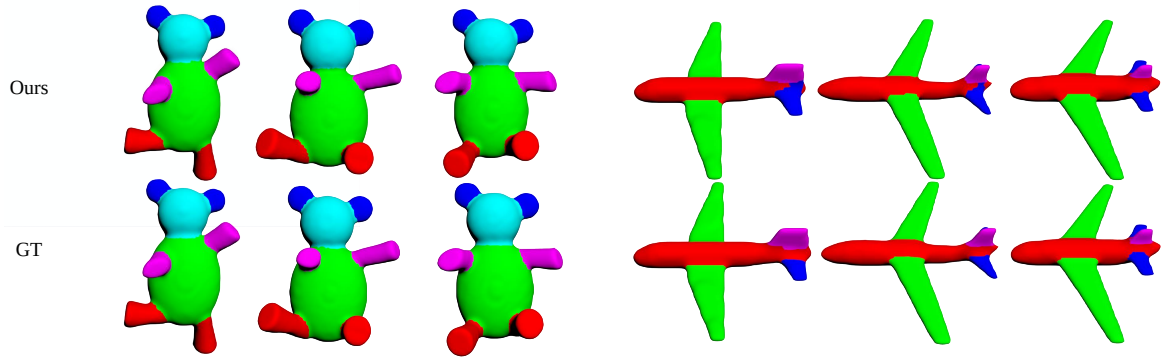

Figure 5: The comparison on the PSB dataset between our segmentation result ("Ours" in the image) and the ground truth ("GT" in the image).

4.3. 3D shape segmentation

Similar to Guo et al. (2015), we use the following segmentation accuracy metric to evaluate the performance of our approach:

$$Accuracy = \sum_{i \in T} t_i \mathbf{u}\left(l_i\right) / \sum_{i \in T} t_i, \tag{3}$$

where T is the face set of the testing 3D shapes, t_i is the area of the face i, and l_i is the predicted label of face i. $\mathbf{u}\left(l_i\right)$ is equal to 1 if the prediction is correct, otherwise, it is 0.

Tables 1, 2, and 3 compare the accuracy of our method and other approaches on the PSB dataset, the small COSEG dataset, and the Human Body dataset, respectively. From the tables, we can see that our method obtains an average accuracy of 94.94% on the PSB dataset, 93.51% on the small COSEG dataset, and

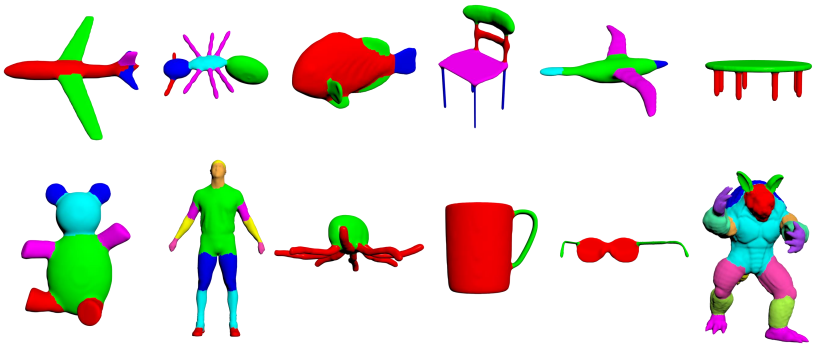

Figure 6: The examples of the 3D shape segmentation results obtained from our method on the PSB dataset.

Figure 7: The 3D shape segmentation results of our method on the COSEG dataset.

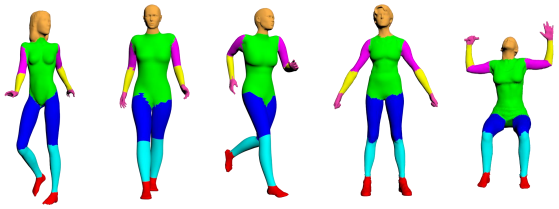

Figure 8: The 3D shape segmentation results of our method on the Human Body dataset.

Table 4: The AUC Score of our algorithm compared with SOTA algorithms, including Saliency of Large Point Sets (LS, Shtrom et al. (2013)), Schelling Point (SP, Chen et al. (2012)), Cluster-Based Point Set Saliency (CS, Tasse et al. (2015)), PCA-Based Saliency (PS, Tasse et al. (2016)), and Mesh Saliency via Spectral Processing (MS, Song et al. (2014)) on SHREC 2007 Dataset.

Algorithms	SH2011	LS	SP	CS	PS	MS	Ours
Airplane	0.6597	**0.6705**	0.6609	0.6308	0.6409	0.6160	0.6625
Human	**0.6383**	0.5893	0.6311	0.5745	0.5921	0.5695	0.6325
Cup	–	0.6192	0.5864	0.6122	0.6135	0.5934	**0.6379**
Glass	0.6018	0.5727	0.6097	0.5225	0.5530	0.5297	**0.6351**
Ant	**0.6468**	0.6349	0.6029	0.6056	0.5791	0.5696	0.6391
Octopus	**0.6592**	0.6237	0.5679	0.5480	0.5726	0.5342	0.6448
Table	–	0.6651	0.6162	0.6313	0.6168	0.5802	**0.6738**
Buste	–	0.6260	0.5757	0.6236	**0.6352**	0.5620	0.6261
Teddy	–	0.5641	0.6596	0.5671	0.5682	0.5530	**0.7033**
Hand	–	0.6339	0.5668	0.6060	0.6022	0.5763	**0.6341**
Plier	0.6182	**0.6236**	0.6128	0.5997	0.5754	0.5615	0.6165
Fish	0.5902	0.6717	0.6562	0.6651	**0.6736**	0.6309	0.6039
Four-legged	0.6140	**0.6168**	0.6056	0.6024	0.6112	0.6005	0.6006
Bird	**0.6390**	0.6217	0.6169	0.6010	0.5984	0.5627	0.6297
Spring	–	0.5545	0.5751	0.5512	0.5339	0.5523	**0.6013**
Armadillo	**0.6859**	0.6656	0.6792	0.6560	0.6570	0.5996	0.6825
Chair	–	0.6566	0.6492	0.5871	0.5799	0.5505	**0.6726**
Mechanic	–	0.6932	0.6548	0.6964	**0.7065**	0.5325	0.6924
Bearing	–	0.6387	0.6063	0.6472	0.6322	0.4986	**0.6517**
Vase	–	0.6217	0.5540	0.6158	**0.6251**	0.6058	0.6190
Average	–	0.6282	0.6144	0.6072	0.6083	0.5689	**0.6419**

93.02% on the Human Body dataset, which achieves significantly better performance than other methods. Figure 5 shows the comparison between the segmentation results of our method and the ground truth. Figure 6, Figure 7, and Figure 8 show some samples of the segmentation results of our method on the PSB, COSEG, and Human Body datasets, respectively.

4.4. Points of interest detection

In the POI-detection task, we apply the Area Under the ROC Curve (AUC Score) to quantify the performances of each method. The Receiver Operating Characteristic (ROC) curve is a plot illustrating the performance of a binary classifier for different threshold values. The area under the ROC curve is previously widely used to compare saliency models in the 2D case.

Table 4 shows the AUC Score comparison between our method and some of the state-of-the-art methods, including Saliency of Large Point Sets (Shtrom et al. (2013)), Schelling Point (Chen et al. (2012)), Cluster-Based Point Set Saliency (Tasse et al. (2015)), PCA-Based Saliency (Tasse et al. (2016)), and Mesh Saliency via Spectral Processing (Song et al. (2014)) on SHREC 2007 dataset. From the table, we can conclude that our method obtains better performance than other methods on average and on a large proportion of categories. Moreover, our visualized result of POI detection on the SHREC 2011 dataset is presented in Figure 9.

4.5. Ablation studies

The ablation study performed in this study aims to evaluate the impact of different factors on the performance of our proposed method. In order to conduct ablation studies and verify the effectiveness of our framework, we focus on the downstream task of 3D shape segmentation. More specifically, we evaluate the impact of the loss function and the use of pre-trained ResNet models on the segmentation accuracy. Regarding the loss function, we compare the segmentation results of our method using different loss functions, including no loss function, within-modal loss, cross-modal loss, and a combination of within-modal and cross-modal loss. Our results, as shown in Figure 10, evidence that using a combination of the two loss functions resulted in higher accuracy compared to using any single loss function. The within-modal loss enables the model to find more discriminative features within the same modality, which helps

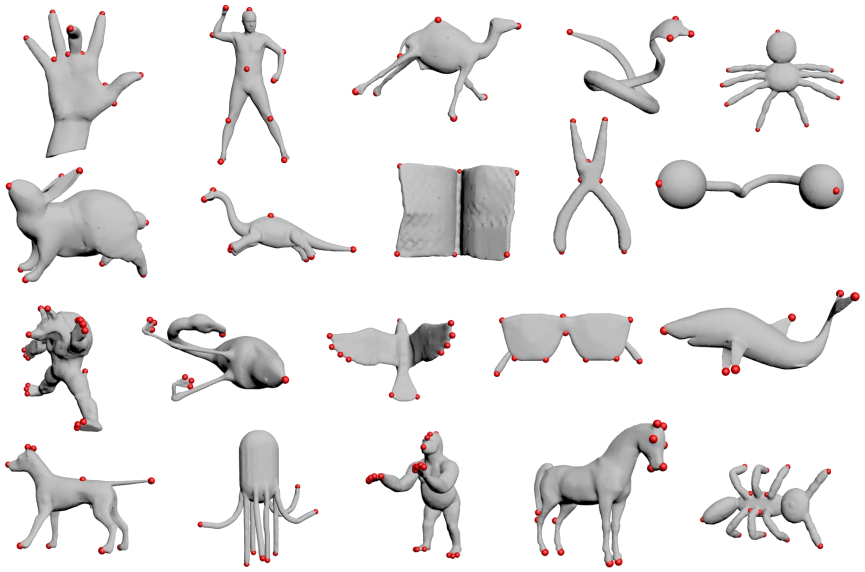

Figure 9: Some of the visualized results of POI detection of our algorithm on the SHREC 2011 dataset. The red balls represent POIs detected by our method.

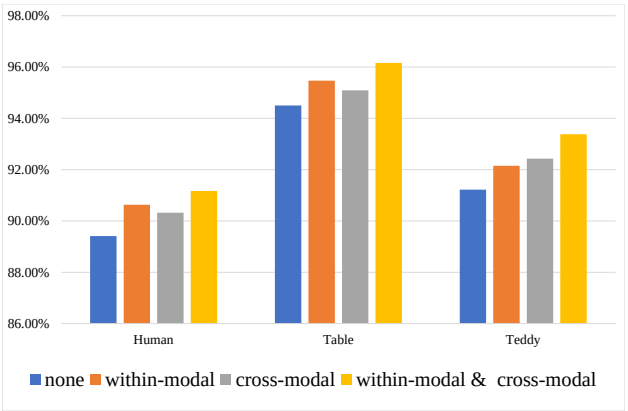

Figure 10: Comparison of segmentation results for different combinations of losses. The "none" represents using neither within-modal nor cross-modal losses, the "within-modal" represents using only within-modal loss while training, the "cross-modal" represents using only cross-modal loss likewise, and the "within-modal and cross-modal" represents using both losses in the training process.

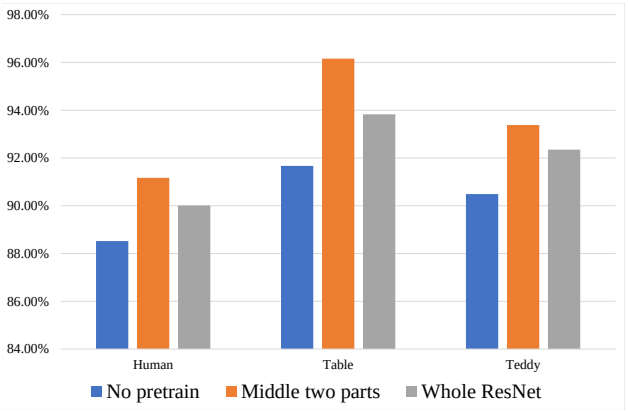

Figure 11: The comparison of segmentation results among different types of pre-trained encoders of image. The "no pre-train" stands for not using any pre-trained and using MLPs with randomized parameters in the image encoder, the "middle two parts" stands for using the middle two parts of ResNet with pre-trained parameters, and the "whole ResNet" stands for using the whole ResNet. The ResNet is pre-trained using the ImageNet Deng et al. (2009) dataset, and all parameters are fixed in the training process.

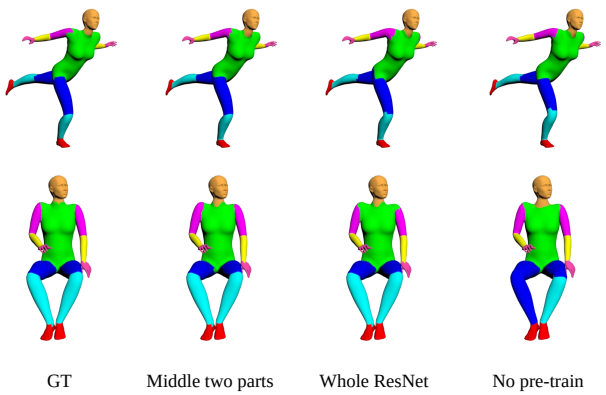

Figure 12: Results for ablation studies on ResNet pre-training. "GT" stands for ground truth segmentation. The "Middle Two parts", "Whole ResNet", and "No pre-train" is the same as in Figure 11.

improve the model's performance in 3D mesh segmentation tasks. On the other hand, the cross-modal loss strengthens the correlation between different modalities, enhancing the model's multi-angle understanding of 3D shapes. This indicates the importance of jointly considering both within-modal and cross-modal information in our proposed method. We also conduct an investigation into the effectiveness of pre-training ResNet models. In particular, we compare the segmentation results using a pre-trained ResNet model with intermediate layers against those obtained without pre-training or with full ResNet pre-training. Our results presented in Figure 11 show that utilizing a pre-trained ResNet model with intermediate layers led to superior segmentation performance compared to not using pre-training or training on the entire ResNet model. Using the intermediate layers of ResNet as the initialization for the image encoder performs better than other configurations. The feature representations in the intermediate layers already possess strong abstraction capabilities for image processing tasks, making them more suitable for supporting 3D shape analysis tasks. The samples of segmentation results in this experiment are shown in Figure 12. This highlights the importance of using pre-trained models, especially those with intermediate layers, to achieve better segmentation results.

5. Limitation and future works

There are a few limitations that our algorithm currently faces. Firstly, feature descriptors need to be computed for each face using our algorithm, which requires the 3D shape to be manifold. We plan on addressing this by extending our approach to non-manifold shapes in the future. Secondly, the computational cost of using our proposed face classification network is relatively high. To mitigate this issue, we will explore more efficient network architectures in our future work.

6. Conclusion

In this paper, we propose a novel 3D shape analysis framework based on multi-modal contrastive learning algorithm. Previous methods usually use a large amount of human-labeled 3D mesh data, which is costly. We design within-modal and cross-modal loss in our method and train the encoder, which can effectively extract the features on 3D mesh models. The projection from 3D to 2D enables the algorithm to exploit the features with the help of ResNet layers and conduct contrastive learning within 3D shapes as well as between 3D shapes and 2D projected images. This mechanism synergistically combines the strengths of multiple modalities, leading to the extraction of more informative and discriminative features. Experimental results on public benchmarks show that our method outperforms other approaches.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (62172356, 61872321), Zhejiang Provincial Natural Science Foundation of China (LZ25F020012), the Ningbo Major Special Projects of the "Science and Technology Innovation 2025" (2020Z005, 2020Z007, 2021Z012).

References

Aubry, M., Schlickewei, U., Cremers, D., 2011. The wave kernel signature: A quantum mechanical approach to shape analysis. 2011 IEEE International Conference on Computer Vision Workshops , 1626–1633.

Boykov, Y., Veksler, O., Zabih, R., 2001. Fast approximate energy minimization via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence 23, 1222–1239.

Bronstein, M.M., Kokkinos, I., 2010. Scale-invariant heat kernel signatures for non-rigid shape recognition. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition , 1704–1711.

Chen, C., Qian, S., Fang, Q., Xu, C., 2021. HAPGN: Hierarchical attentive pooling graph network for point cloud segmentation. IEEE Transactions on Multimedia 23, 2335–2346. doi:10.1109/TMM.2020.3009499.

Chen, T., Kornblith, S., Norouzi, M., Hinton, G., 2020a. A simple framework for contrastive learning of visual representations, in: International conference on machine learning, PmLR. pp. 1597–1607.

Chen, T., Kornblith, S., Norouzi, M., Hinton, G.E., 2020b. A simple framework for contrastive learning of visual representations. ArXiv abs/2002.05709.
Chen, X., Golovinskiy, A., Funkhouser, T., 2009. A benchmark for 3D mesh segmentation. ACM Transactions on Graphics 28, 1–12.
Chen, X., Saparov, A., Pang, B., Funkhouser, T., 2012. Schelling points on 3D surface meshes. ACM Transactions on Graphics 31, 29.
Cheng, M.M., Mitra, N.J., Huang, X., Torr, P.H.S., Hu, S.M., 2015. Global contrast based salient region detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 37, 569–582. doi:10.1109/TPAMI.2014.2345401.
Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. ImageNet: A large-scale hierarchical image database. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition , 248–255.
Duan, L., Wu, C., Miao, J., Qing, L., Fu, Y., 2011. Visual saliency detection by spatially weighted dissimilarity, in: Proceedings of IEEE Computer Vision and Pattern Recognition, pp. 473–480.
Gal, R., Cohen-Or, D., 2006. Salient geometric features for partial shape matching and similarity. ACM Transactions on Graphics 25, 130–150.
Grill, J.B., Strub, F., Altch'e, F., Tallec, C., Richemond, P.H., Buchatskaya, E., Doersch, C., Pires, B.Á., Guo, Z.D., Azar, M.G., Piot, B., Kavukcuoglu, K., Munos, R., Valko, M., 2020. Bootstrap your own latent: A new approach to self-supervised learning. ArXiv abs/2006.07733.
Guo, K., Zou, D., Chen, X., 2015. 3D mesh labeling via deep convolutional neural networks. ACM Transactions on Graphics 35, 1–12.
Guy, G., Medioni, G.G., 1997. Inference of surfaces, 3D curves, and junctions from sparse, noisy, 3D data. IEEE Transactions on Pattern Analysis and Machine Intelligence 19, 1265–1277.
Hanocka, R., Hertz, A., Fish, N., Giryes, R., Fleishman, S., Cohen-Or, D., 2019. MeshCNN: A network with an edge. ACM Transactions on Graphics 38, 1–12.
He, K., Fan, H., Wu, Y., Xie, S., Girshick, R., 2020. Momentum contrast for unsupervised visual representation learning, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9729–9738.
He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.B., 2019. Momentum contrast for unsupervised visual representation learning. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition , 9726–9735.
He, K., Zhang, X., Ren, S., Sun, J., 2015. Deep residual learning for image recognition. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition , 770–778.
Hou, X., Zhang, L., 2007. Saliency detection: A spectral residual approach, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1–8.
Hu, R., Fan, L., Liu, L., 2012. Co-segmentation of 3D shapes via subspace clustering. Computer Graphics Forum 31, 1703–1713.
Huang, Q., Koltun, V., Guibas, L., 2011. Joint shape segmentation with linear programming. ACM Transactions on Graphics 30, 1–12.
Kaick, O.V., Fish, N., Kleiman, Y., Asafi, S., Cohen-OR, D., 2014. Shape segmentation by approximate convexity analysis. ACM Transactions on Graphics 34, 1–11.
Kalogerakis, E., Averkiou, M., Maji, S., Chaudhuri, S., 2017. 3D shape segmentation with projective convolutional networks, in: Proceedings of IEEE Computer Vision and Pattern Recognition, pp. 3779–3788.
Kalogerakis, E., Hertzmann, A., Singh, K., 2010. Learning 3D mesh segmentation and labeling. ACM Transactions on Graphics 29, 1–12.
Kim, V.G., Li, W., Mitra, N.J., Chaudhuri, S., DiVerdi, S., Funkhouser, T., 2013. Learning part-based templates from large collections of 3D shapes. ACM Transactions on Graphics 32, 1–12.
Koch, C., Ullman, S., 1987. Shifts in selective visual attention: towards the underlying neural circuitry. Human neurobiology 4, 219–227.
Lahav, A., Tal, A., 2020. MeshWalker: Deep mesh understanding by random walks. ACM Transactions on Graphics 39, 1–13.
Le, T., Bui, G., Duan, Y., 2017. A multi-view recurrent neural network for 3D mesh segmentation. Computer & Graphics 66, 103–112.
Lee, C.H., Varshney, A., Jacobs, D.W., 2005. Mesh saliency. ACM SIGGRAPH 2005 Papers 24, 659–666.
Leifman, G., Shtrom, E., Tal, A., 2012. Surface regions of interest for viewpoint selection. IEEE Transactions on Pattern Analysis and Machine Intelligence 38, 2544–2556.
Liu, H., Guo, Y., Ma, Y., Lei, Y., Wen, G., 2021. Semantic context encoding for accurate 3D point cloud segmentation. IEEE Transactions on Multimedia 23, 2045–2055. doi:10.1109/TMM.2020.3007331.
Liu, J., Zhao, Y., Chen, S., Zhang, Y., 2022. A 3D mesh-based lifting-and-projection network for human pose transfer. IEEE Transactions on Multimedia 24, 4314–4327. doi:10.1109/TMM.2021.3115628.
Mantiuk, R., Myszkowski, K., Pattanaik, S., 2003. Attention guided MPEG compression for computer animations, in: SCCG '03: Proceedings of the 19th Spring Conference on Computer Graphics, Association for Computing Machinery, New York, NY, USA. p. 239–244.
Maron, H., Galun, M., Aigerman, N., Trope, M., Dym, N., Yumer, E., Kim, V.G., Lipman, Y., 2017. Convolutional neural networks on surfaces via seamless toric covers. ACM Transactions on Graphics 36, 1–10.
Mitchel, T.W., Kim, V.G., Kazhdan, M.M., 2021. Field convolutions for surface CNNs. 2021 IEEE/CVF International Conference on Computer Vision , 9981–9991.
Poulenard, A., Ovsjanikov, M., 2018. Multi-directional geodesic neural networks via equivariant convolution. ACM Transactions on Graphics 37, 1–14.
Shapira, L., Shalom, S., Shamir, A., Cohen-Or, D., Zhang, H., 2010. Contextual part analogies in 3D objects. International Journal of Computer Vision 89, 309–326.

Shapira, L., Shamir, A., Cohen-Or, D., 2008. Consistent mesh partitioning and skeletonisation using the shape diameter function. The Visual Computer 24, 249–259.
Sharp, N., Attaiki, S., Crane, K., Ovsjanikov, M., 2020. DiffusionNet: Discretization agnostic learning on surfaces. ACM Transactions on Graphics 41, 1–16.
Shilane, P., Funkhouser, T.A., 2007. Distinctive regions of 3D surfaces. ACM Transactions on Graphics 26, 7.
Shtrom, E., Leifman, G., Tal, A., 2013. Saliency detection in large point sets, in: IEEE International Conference on Computer Vision, pp. 3591–3598.
Shu, Z., Gao, L., Yi, S., Wu, F., Ding, X., Wan, T., Xin, S., 2023. Context-aware 3D points of interest detection via spatial attention mechanism. ACM Transactions on Multimedia Computing, Communications and Applications 19, 1–19.
Shu, Z., Li, S., Xin, S., Liu, L., 2025a. 3D shape segmentation with potential consistency mining and enhancement. IEEE Transactions on Multimedia 27, 133–144.
Shu, Z., Qi, C., Xin, S., Hu, C., Wang, L., Zhang, Y., Liu, L., 2016. Unsupervised 3D shape segmentation and co-segmentation via deep learning. Computer-Aided Geometric Design 43, 39–52.
Shu, Z., Sun, X., Xin, S., Liu, L., 2024a. 3D shape segmentation via attentive nonuniform downsampling. IEEE Transactions on Circuits and Systems for Video Technology 34, 12184–12196.
Shu, Z., Wu, T., Shen, J., Xin, S., Liu, L., 2024b. Semi-supervised 3D shape segmentation via self refining. IEEE Transactions on Image Processing 33, 2044–2057.
Shu, Z., Yu, J., Chao, K., Xin, S., Liu, L., 2025b. A multi-modal attention-based approach for points of interest detection on 3D shapes. IEEE Transactions on Visualization and Computer Graphics 31, 1698–1712.
Sidi, O., van Kaick, O., Kleiman, Y., Zhang, H., Cohen-Or, D., 2011. Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering. ACM Transactions on Graphics 30, 1–10.
Smirnov, D., Solomon, J.M., 2021. HodgeNet: Learning spectral geometry on triangle meshes. ACM Transactions on Graphics 40, 166:1–166:11.
Song, R., Liu, Y., Martin, R.R., Rosin, P.L., 2014. Mesh saliency via spectral processing. ACM Transactions on Graphics 33, 1–17.
Tasse, F.P., Kosinka, J., Dodgson, N., 2015. Cluster-based point set saliency, in: IEEE International Conference on Computer Vision, pp. 163–171.
Tasse, F.P., Kosinka, J., Dodgson, N.A., 2016. Quantitative analysis of saliency models, in: SIGGRAPH ASIA 2016 Technical Briefs, ACM, New York, USA. pp. 19:1–19:4.
Tian, Y., Krishnan, D., Isola, P., 2019. Contrastive multiview coding. ArXiv abs/1906.05849.
Wang, Y., Asafi, S., van Kaick, O., Zhang, H., Cohen-Or, D., Chen, B., 2012. Active co-analysis of a set of shapes. ACM Transactions on Graphics 31, 1–10.
Wang, Y., Gong, M., Wang, T., Cohen-Or, D., Zhang, H., Chen, B., 2013. Projective analysis for 3D shape segmentation. ACM Transactions on Graphics 32, 1–12.
Wu, Z., Xiong, Y., Yu, S.X., Lin, D., 2018. Unsupervised feature learning via non-parametric instance discrimination. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition , 3733–3742.
Xie, Z., Xu, K., Liu, L., Xiong, Y., 2014. 3D shape segmentation and labeling via extreme learning machine. Computer Graphics Forum 33.
Xie, Z., Xu, K., Shan, W., Liu, L., Xiong, Y., Huang, H., 2015. Projective feature learning for 3D shapes with multi-view depth images. Computer Graphics Forum 34, 1–11.
Yang, Y., Pan, H., Liu, S., Liu, Y., Tong, X., 2018. PFCNN: Convolutional neural networks on 3D surfaces using parallel frames. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition , 13575–13584.
Yee, Y.H., Pattanaik, S.N., Greenberg, D.P., 2005. Spatiotemporal sensitivity and visual attention for efficient rendering of dynamic environments. ACM Transactions on Graphics 20, 39–65.
Zhuang, S., Wei, G., Cui, Z., Zhou, Y., 2025. Robust hybrid learning for automatic teeth segmentation and labeling on 3D dental models. IEEE Transactions on Multimedia 27, 792–803. doi:10.1109/TMM.2023.3289760.