Cloud Sphere: A 3D Shape Representation via Progressive Deformation

Zongji Wang
Key Laboratory of Network Information System Technology (NIST),
Aerospace Information Research Institute,
Chinese Academy of Sciences,
Beijing 100190, China.
wangzongji@aircas.ac.cn

Yunfei Liu
State Key Laboratory of Virtual Reality Technology and Systems,
School of Computer Science and Engineering,
Beihang University, Beijing 100191, China.
lyunfei@buaa.edu.cn

Feng Lu
State Key Laboratory of Virtual Reality Technology and Systems,
School of Computer Science and Engineering,
Beihang University, Beijing 100191, China.
Peng Cheng Laboratory, Shenzhen 518000, China.
lufeng@buaa.edu.cn

Abstract

In the area of 3D shape analysis, the geometric properties of a shape have long been studied. Instead of directly extracting representative features using expert-designed descriptors or end-to-end deep neural networks, this paper is dedicated to discovering distinctive information from the shape formation process. Concretely, a spherical point cloud served as the template is progressively deformed to fit the target shape in a coarse-to-fine manner. During the shape formation process, several checkpoints are inserted to facilitate recording and investigating the intermediate stages. For each stage, the offset field is evaluated as a stage-aware description. The summation of the offsets throughout the shape formation process can completely define the target shape in terms of geometry. In this perspective, one can derive the point-wise shape correspondence from the template inexpensively, which benefits various graphic applications. In this paper, the Progressive Deformation-based Auto-Encoder (PDAE) is proposed to learn the stage-aware description through a coarse-to-fine shape fitting task. Experimental results show that the proposed PDAE has the ability to reconstruct 3D shapes with high fidelity, and consistent topology is preserved in the multi-stage deformation process. Addi-

Figure 1: The coarse-to-fine progressive deformation process for shape reconstruction from a spherical template. The distinctive information of the shape is progressively added throughout the shape formation process.
tional applications based on the stage-aware description are performed, demonstrating its universality.

1. Introduction

Our key motivation comes from the sculpture creation process. Imagine the carving process, in which the artist progressively removes materials from a stone to create 3D visual arts. In different stages of carving, more and more details are embossed on top of low-fidelity surfaces. Therefore, one can intuitively assume that the distinctive information of the shape is progressively added throughout the sculpting. Fig. 1 illustrates the shape formation process, in which a spherical point cloud is progressively deformed to fit the target shape.

The coarse-to-fine manner has been widely used and proven to be effective in various 3D computer graphics tasks, such as 3D shape recognition, segmentation [19, 26], reconstruction [23], generation [14], etc.

For the recognition and segmentation problems, multi-scale features are extracted and integrated in a coarse-to-fine manner to enrich the descriptive ability for the encoding. For the reconstruction and generation problems, through a coarse-to-fine manner, shape details can be added progressively in the generation process. The original problem can be decomposed into multiple sub-problems, which are simpler to solve. Recent works in both problems usually combine features from different scales to form a relatively comprehensive latent code for specific tasks [14, 19, 23, 26]. In other words, they are task-driven methods.

Task-driven learning usually requires labour-intensive annotation such as object class label, point-wise semantic label, dense correspondence, etc. The speed of knowledge enchanting is much slower than that of data increasing. In addition to having high demand on data annotation, a task-driven method usually learns features that are highly relevant to the specific task, which decreases the universality of features. Thus, in a big data surge background, semi-supervised, self-supervised, or unsupervised general purpose representation learning is required. The learned representation can further be used for various applications directly or with slight fine-tuning.

In this paper, a novel 3D shape representation is introduced, which is named as Cloud Sphere according to the spherical point cloud template. We propose the Progressively Deformation-based Auto-Encoder (PDAE) which models the shape formation process. A spherical point cloud served as the template is progressively deformed to fit the target shape in a coarse-to-fine manner. Several checkpoints are inserted to facilitate recording and investigating the intermediate stages. For each stage, the offset field is evaluated as a stage-aware description. The set of per-stage offsets throughout the shape formation process can completely define the target shape in terms of geometry. The Cloud Sphere representation explicitly encodes the deformation residuals during shape formation process. Using this representation, any shape can be modelled by a series of deformation fields from the spherical point cloud template. The same template acts as a bridge connecting different target shapes, intuitively provides their dense correspondences.

There are several differences between the proposed the Cloud Sphere representation and the task-driven ones. First, this representation is learnt through an unsupervised self-reconstruction framework. Therefore, additional annotations are not required. Second, while learning the representation, multi-scale features act as conditions respectively in different reconstruction stages, rather than integrated as one latent code for a specific high-level 3D understanding task. This enables the learnt features to encode knowledge from corresponding reconstruction level. For interpretability, this method paves the way for studying the features’ properties at different scales. Third, the template deformation framework naturally provides point-wise correspondence between the template and the reconstructed point clouds. Note that point-wise correspondence is the low-level geometric information, which is of benefit to various high-level graphic applications. This is considered as the foundation of the universality of Cloud Sphere.

Our main contributions can be summarized as follows:

1) A novel 3D shape representation named Cloud Sphere which encodes knowledge from different stages of shape formation process.

2) A novel Progressively Deformation-based Auto-Encoder for point cloud feature encoding. This method enables the feature learning from each shape formation stage.

3) Experimental results demonstrate that the PDAE possesses better self-reconstruction ability compared to the state-of-the-arts, and the correspondence obtained from the Cloud Sphere representation preserves better local and global topology.

4) The point-wise correspondence between different shapes can be obtained from the Cloud Sphere representation. Based on our method, various computer graphics applications are performed.

2. Related Work

In the area of 3D shape analysis, the geometric properties of a shape have long been studied. The expert-designed descriptors [1, 9, 13, 21, 35] focus on specific geometric properties. However, a single descriptor is often insufficient to cover complex high-level parsing scenarios. More detailed discussion of these descriptors is given in [12]. Deep learning methods [11, 18, 19, 30, 34] usually map the shape into a vector in high-dimensional space. While the descriptive capacity of features increases, the interpretability decreases.
The interested readers are referred to an overview of deep geometry learning [27].

In this work, we use a progressive deformation framework for 3D shape self-reconstruction to learn the Cloud Sphere representation. Therefore, we mainly focus on the most related works which could be clustered into two classes: 1) multi-scale 3D shape representation and 2) deformation-based 3D shape modeling.

2.1. Multi-Scale 3D Shape Representation

Multi-scale features are widely used to enhance the features’ descriptive ability and robustness. Qin et al. proposed PointDAN [20], a 3D Domain Adaptation Network for point cloud data. This method jointly aligns the global and local features at multiple levels. Liu et al. proposed Point2Sequence [15] to learn 3D shape features from point clouds. This method employs an attention-based sequence learning model to utilize a sequence of multi-scale areas of each local region. In which the correlation of different areas are captured to enhance the discriminability of learned features. PointNet [18] operates on unordered point set directly. The independently processed points are aggregated into global feature by max-pooling. In the follow-up work PointNet++ [19], the authors improved PointNet by incorporating local dependencies and hierarchical feature learning in the network. This method is the first to extract features for multi-scale local regions individually and aggregates these features by concatenation, where the two steps are repeated to complete the hierarchical feature extraction.

The hierarchical multi-scale local region abstraction framework by PointNet++ is then followed by many works for point cloud understanding [4, 7, 31]. For example, Guo et al. [7] proposed a multi-scale deep network for semantic classification of terrestrial laser scanner (TLS) point clouds. This method subsamples the input point cloud at multiple scales, and then inputs each scale into the 3D CNN. The outputs are combined to form multi-scale and hierarchical deep features. Gadelha et al. presented MRTNet [4] (multi-resolution tree-structured networks) to process point clouds for 3D shape understanding and generation tasks. This method represents a 3D shape as a set of 1D ordered list of points. Yin et al. introduced LOGAN [31], which uses multi-scale features extracted from PointNet++ for unpaired 3D shape domain transfer. This method first embeds the shapes in a common latent space through a self-reconstruction task, and then trains an additional network for shape domain transfer. In the shape representation step, multi-scale features are concatenated as a comprehensive latent code.

2.2. Deformation-Based 3D Shape modeling

Deformation-based 3D modeling has been studied for a long time. Here we focus more on recent deep learning based methods. The deformation-based framework can be applied to various 3D data formats including mesh, voxel and point cloud. In the following, we give a fast review of these methods according to the data formats.

Mesh data. Mesh deformation has been used for 3D surface generation [22], single-view 3D mesh reconstruction [23], target-driven deformation [25], etc. Specifically, Sinha et al. proposed SurfNet [22] to use a parametrization of one or a few base shapes to represent training data as parametrized meshes. The vertex coordinates are mapped to the resulting UV space and fed into 2D neural networks for surface generation. Wang et al. proposed Pixel2Mesh [23], a graph-based convolutional neural network which directly produces a 3D shape in triangular mesh from a single color image. An ellipsoid is progressively deformed and remeshed to fit the target 3D shape described in the UV image. Wang et al. introduced the 3D Deformation Network (3DN) [25], an end-to-end network deforming a source 3D mesh to resemble a target shape in various kinds of modalities. They estimated the per-vertex offset displacements from the source mesh to the target shape. These methods usually require dense correspondence between the shapes [22], and generate results constrained by the input mesh topology.

Voxel and point cloud data. Voxel data can be seen as the regularized down-sampling of dense point clouds. Therefore we review these two data formats together. Yumer et al. [33] introduced the first 3D volumetric generative network that learns to predict per-voxel dense 3D deformation flows using explicit high level deformation intentions. This method uses Free Form Deformation (FFD) to apply the deformation flow to the original 3D shape. Voxel-based methods usually have limited spatial resolution due to the high computational and memory cost. Yang et al. proposed FoldingNet [29], which uses a folding-based decoding operation for point cloud reconstruction. The folding operation that reconstructs a surface from a 2D grid essentially establishes a mapping from a 2D regular domain to a 3D point cloud. In the same year, Groueix et al. introduced AtlasNet [6], a 3D shape surface generation framework. They represented 3D shape as a collection of parametric surface elements. The primitive surface elements are projected into 3D space to fit the target 3D shape. Based on the idea of deforming 2D planes to 2-manifolds in 3D space used by FoldingNet and AtlasNet, Mehr et al. introduced DiscoNet [16] to learn a more complicated 3D shape embedding. Taking account of the topological variability in sets of 3D shapes, multiple branches of auto-encoders are trained to learn feature embeddings, which are not necessary to lie on a connected manifold. The decoding phase is performed by deforming a pre-learned 3D template representing the mean shape to the target instance. Yin et al. introduced P2P-NET [32], a bi-directional point displace-
ment network which learns geometric transformations between point clouds from two domains. In summary, these methods perform point/voxel-wise deformation to generate new shapes.

In recent years, there emerge methods modeling the shape deformation process as a continuous differential problem [10, 17, 28]. In these works, the continuous normalizing flow (CNF) [3, 5] is used to establish a continuous invertible transformation with the help of a continuous-time dynamic. The CNF’s outputs and the input gradients can be solved by ordinary differential equation (ODE) solvers. In this paper, we use a discrete deformation framework for the convenience to utilize multi-scale supervision.

3. Definition of Cloud Sphere

We propose Cloud Sphere, a novel 3D representation method for point clouds which explicitly encodes the deformation process during shape formation. This representation has some great properties for 3D shape analysis. First, all shapes are modeled by a deformation from the spherical template, thus point-wise correspondences can be acquired effortlessly. Second, point clouds are highly flexible. Without topological restrictions, the template is capable of fitting to arbitrary shapes with complex topological structure. Third, through this representation, it is easy to model the hierarchical, progressive shape generation process, which is of benefit to high-level applications such as shape attribute localization and editing.

In the following, the Cloud Sphere representation is defined mathematically. In the Cloud Sphere representation, a 3D shape \( S := \{ s_i \in \mathbb{R}^3 \}_{i=1}^n \) is modeled by a spherical point cloud template \( T := \{ q_i \in \mathbb{R}^3 \}_{i=1}^n \) and the point-wise displacements \( D^k := \{ d^k_i \in \mathbb{R}^3 \}_{i=1}^n \) from the sphere to the 3D shape surface. Note that \( S \) and \( T \) are point sets with the same cardinality, and \( D \) is the point-wise offsets corresponding to the points in \( T \). The stage of shape formation process is represented by \( k \). The larger \( k \) is, the higher abstract level of the shape is. The reconstructed shape of higher abstract level contains less geometric details. The target shape \( S \), spherical template \( T \) and the point-wise displacements \( D^k \) subject to the equation as:

\[
S = T + \sum_{k=0}^{K} D^k. \tag{1}
\]

Define \( D := \{ D^k | k \in \{ 0, ..., K \} \} \) as the set of the point-wise offsets in different shape formation stages. The deformation from the spherical point cloud \( T \) to the target point cloud \( S \) is decomposed into different deformation stages. The shape formation is the coarsest (close to \( T \)) while \( k = K \), and is the finest (close to \( S \)) while \( k = 0 \). In consequence, the shape \( S \) can be defined by \( \{ T, D \} \), which is named the Cloud Sphere representation.

This is a coarse-to-fine manner to add up a series of deformations \( D^k \) together, which allows the shape to be gradually refined in detail. In addition, the multi-stage deformation process directly encodes distinctive information, which can help shape analysis and high-level applications.

4. Method

In this section, we describe the Progressive Deformation-based Auto-Encoder (PDAE) for the Cloud Sphere representation learning. First, for computational feasibility, we model the ideal equation Eq. 1 as a practical optimization problem. Second, based on the practical problem, we introduce the network design for the PDAE. Third, we give a detailed description about the main constraints to train the network.

4.1. Problem modeling

As defined in Eq. 1, a shape \( S \) can be represented by the summation of a spherical template \( T \) and the corresponding displacements \( D \). In practice, it is not common that different shapes have the same point order in point cloud format; instead, the point set is naturally independent of the permutation order. Therefore, the equation \( S = T + \sum_{k=0}^{K} D^k \) cannot hold in common. To make this problem computationally feasible, it is modeled as an optimization objective function:

\[
\min_D \text{dist}(S, T + \sum_{k=0}^{K} (D^k)). \tag{2}
\]

In this objective function, the distance between two point clouds are minimized. We choose the Chamfer Distance (CD) between two point sets as the measurement for shape similarity. Detailed description can be found in section 4.3.

4.2. Network Design

In order to solve the optimization objective defined in Eq. 2, we design the Progressive Deformation-based Auto-Encoder, modeling the coarse-to-fine progressive shape formation process.

As shown in Fig. 2, a 3D shape \( S^0 \) in point cloud format is fed into the encoding phase, which consists of the multi-scale feature extraction module. In this module, features from different levels of abstraction are extracted using the set abstraction modules from PointNet++ [19]. The abstracted feature from the top abstraction level \( z^K \) is then fed into the corresponding coarsest (the \( K_{th} \) level, in short, \( LV.K \)) Point Cloud Reconstruction (PCR) module, in which the spherical template is deformed to fit the coarsest level of reference point cloud \( P^K \). Multiple PCRs in parallel with respect to the set abstraction modules are stacked to
Progressive Deformation-based Auto-Encoder

The key variables in the PDAE can be formulated as follows:

\[
\begin{align*}
z^k, S^k & = f(z^{k-1}, S^{k-1}), \\
D^k & = g(z^k, R^{k+1}), \\
R^k & = R^{k+1} + D^k,
\end{align*}
\]

where the superscript \( k \) represents the level of abstraction. The larger \( k \) is, the coarser shape is represented, and the less geometric details are depicted. Let \( z^k \) denote the feature extracted from the point cloud in the abstraction level \( k \). Let \( S^k \) denote the point cloud in abstraction level \( k \), which is sub-sampled \( k \) times from the input. Let \( D^k \) denote the predicted displacement field for shape reconstruction in level \( k \), and \( R^k \) denote the output point cloud of the reconstruction in level \( k \). Particularly, \( S^0 \) is the original input point cloud, \( T \) is the input spherical template. The spherical template is assigned to the initial reconstruction, i.e., \( R^{K+1} = T \). The output \( R^0 \) is the finest level of reconstruction. The set abstraction module from PointNet++ is formulated as \( f(\cdot, \cdot) \), which takes the lower (finer) level latent code and point cloud as inputs, and outputs the current level latent code as well as the abstracted point set. The point cloud reconstruction module is formulated as \( g(\cdot, \cdot) \), which takes the higher (coarser) level reconstruction \( R^{k+1} \) and the current level latent code \( z^k \) as inputs, and outputs the current level reconstruction \( R^k \).

After the PDAE is trained, the coarse-to-fine progressive representation Cloud Sphere can be extracted as:

\[
CS := \{T, D^k, [z^k] \{k = 0, \ldots, K\}\},
\]

in which the latent code \( z^k \) is optional to enrich the descriptive ability, but does not influence the definition completeness. In other words, the PDAE can be explained as a residual learning process for point-wise displacements to progressively fit the multi-resolution point clouds. The training constraints are introduced in the next subsection.

### 4.3. Multi-resolution Residual Learning

**Multi-resolution data pre-processing.** In order to guide the template-based progressive shape reconstruction task, the multi-resolution shapes from coarse to fine are required. This paper proposes a novel shape abstraction method to automatically extract multi-resolution point clouds from the original one. This would lower the demand.
for training data in this framework. The data preprocessing method is briefly introduced in below.

As shown in Fig. 3, a 3D point cloud is resampled into different levels of abstraction. Specifically, Farthest Point Sampling (FPS) is applied first for downsampling to get the shape abstraction. Then, for each sampled point, a subset of points are randomly splattered around it with Gaussian distribution. In particular, compared with other downsampling methods, FPS provides the point cloud abstraction with better coverage of the original one. The splattered points according to the abstracted point cloud helps provide more precise supervision.

The multi-resolution point clouds have following properties:

\[
|P_k^k| = |P_{k+1}^k|, \forall 0 \leq k < K, \\
P_k^k = \bigcup_{i=1}^{N_k} \{X | \text{gauss}(x | \mu = p_i^k, \sigma^2)\},
\]

where \(P_k^k = \{p_i^k \in \mathbb{R}^3 \}_{i=1}^{N_k}\) represents the preprocessed result in abstraction level-\(k\).

As revealed in the first equation in Eq. 5, the generated point sets are with the same cardinality with the input, which helps the cascade of multi-stage deformation. The second equation in Eq. 5 means that, the \(k_{th}\) level resampled point cloud \(P_k^k\) consists of several point subsets. Each of the subset is modeled by a set of random variable \(\{X\}\) with \(|\{X\}| = |P_k^k|/k\). The random variable \(X\) obeys Gaussian distribution with the sampled point as mean, and \(\sigma^2\) as the variation. The Gaussian distribution models the point existence probability. The closer to the sampled centroid, the more likely there exists points.

As a result, the generated multi-resolution point clouds resemble the original point cloud from coarse to fine. After point set downsampling and points splattering using Gaussian distribution, the high-level abstracted point clouds can cover low-level ones, but depict more about the shape structure. In the high-level abstraction stage with a relatively low sample-rate, compared to point clouds with merely downsampling, the augmented ones with splattered points benefit from better balance for the distance metrics.

**Shape similarity constraint.** We use Chamfer Distance to measure the difference between two point sets:

\[
L_{CD}(k) = \sum_{p \in P_k^k} \min_{q \in \mathbb{R}^3} ||p-q||^2_2 + \sum_{q \in \mathbb{R}^3} \min_{p \in P_k^k} ||q-p||^2_2,
\]

where \(k\) is the level of abstraction. This Chamfer Distance is invariant to the ordering of the output point cloud. The multi-stage shape similarity loss can be written as:

\[
L_{CD} = \sum_{k=0}^{K} \alpha_k L_{CD}(k),
\]

where \(\alpha_k\) represents the weight of the shape similarity on the abstraction level-\(k\).

**Point set topology regularization.** To regularize the point-wise deformation, we use the local topology preserving term to encourage the deformed template to preserve its own topology.

\[
\omega_{i,j} = \exp(-||q_i - q_j||_2), \forall q \in \mathbb{T}, \\
L_{reg}(k) = \sum_{0 < i \neq j \leq N} \omega_{i,j} ||d_k^k(i) - d_k^k(j)||_2,
\]

in which \(\omega_{i,j}\) is the pairwise weight between two points \((q_i, q_j)\) from the spherical template \(\mathbb{T}\). The closer they are, the bigger \(\omega_{i,j}\) is. Let \(k\) denote the shape abstraction level, therefore \(d_k^k(i)\) is the displacement of the point \(q_i\) in \(R^{k+1}\). Note that the displacement vector \(d^k\) has the same order with the spherical template \(\mathbb{T}\). The multi-stage topology constraints can be written as:

\[
L_{reg} = \sum_{k=0}^{K} \beta_k L_{reg}(k),
\]

in which \(\beta_k\) represents the weight of the topology regularization on the abstraction level-\(k\).

The total loss \(L_{total}\) is:

\[
L_{total} = \sum_{k=K}^{0} (\alpha_k L_{CD}(k) + \beta_k L_{reg}(k)).
\]

We set the weights as \(\alpha_k = [0.5, 0.2, 0.2, 0.2, 0.2]\) and \(\beta_k = [0, 0, 0, 1.0, 10.0]\), with \(k = [0, 1, 2, 3, 4]\).
5. Experimental Results

In this section, we first present the data we used to train and test the proposed PDAE framework. Two experiments are conducted to evaluate the performance of the PDAE. First, self-reconstruction results are qualitatively and quantitatively compared with the state-of-the-art methods. Second, the correspondence results achieved by deformation-based reconstruction are evaluated. Next, an ablation study is conducted to verify the effectiveness of the progressive deformation framework. Finally, several interesting applications based on the Cloud Sphere representation are demonstrated.

5.1. Dataset

Our data is rearranged from the standard ShapeNet Core dataset (v1) [2]. Specifically, we select 4 major semantic sets containing more than 1000 instances in total: airplane, car, chair and table. For each semantic set, we analyse the fine-grained classes. Finally, 4 different fine-grained classes from each semantic class are achieved. After the rearrangement of the ShapeNet Core (v1) dataset, we obtain a challenging dataset for fine-grained shape analysis, on which generalized correspondence among shapes within a coarse class can be defined. The point clouds are generated using the virtual scanner from O-CNN [24], and then for each shape, 4096 points are sampled using FPS.

5.2. Self-reconstruction Evaluation

We compare the proposed PDAE with LOGAN [31] and AtlasNet [6] on the shape self-reconstruction task. In AtlasNet, the same spherical point cloud template as ours is used as input. Table 1 shows the quantitative results. Our method produces point clouds reconstruction with the lowest Chamfer distance and earth mover distance, and highest intersection over union on solid voxelized grids.

Fig. 4 illustrates the qualitative results. The PDAE produces results with more accurate geometry and topology. We can see in the top two rows, the chair leg and back from the PDAE have more geometric details and less noisy points. Similar effects can be observed in the other examples including the table (the third row), the plane (the forth row) and the car (the bottom row).

5.3. Correspondence Evaluation

In a deformation-based reconstruction framework, the template is deformed to fit the target shape. Therefore, there exists point-wise correspondence between the original template and the reconstructed shape. We obtain the bi-projection between the spherical template $T$ and the reconstructed point cloud $P^0$. The spherical template plays the role of a bridge through which various shapes are connected. In this experiment, we evaluate the template-to-shape correspondence.

Table 1: Self-reconstruction results on test sets. Chamfer Distance (CD \times 1000), Earth Mover Distance (EMD \times 100), and Intersection over Union (IoU) on solid voxelized grids are estimated. The ↓ means the lower the better. The ↑ means the higher the better.

| Metrics    | LOGAN [31] | AtlasNet [6] | PDAE (ours) |
|------------|------------|--------------|-------------|
| CD ↓       | 0.23       | 0.82         | 1.60        |
| EMD ↓      | 1.07       | 1.93         | 3.20        |
| IoU ↑      | 0.77       | 0.61         | 0.40        |
| CD ↓       | 0.17       | 0.64         | 1.31        |
| EMD ↓      | 0.94       | 1.67         | 2.39        |
| IoU ↑      | 0.81       | 0.62         | 0.55        |
| CD ↓       | 0.15       | 0.40         | 0.55        |
| EMD ↓      | 0.90       | 1.43         | 1.98        |
| IoU ↑      | 0.81       | 0.69         | 0.61        |
| Mean ↓     | 0.80       | 0.62         | 0.58        |
| Mean ↑     | 1.06       | 0.54         | 0.40        |

In Fig. 5, the template sphere is color-coded according to the coordinate values along three axes respectively. The color is then directly projected to the reconstructed shapes. For each row in Fig. 5, the left column is the color coding direction, the middle is the correspondence result from our PDAE, and the right is that from AtlasNet. We observed that in the rightmost column (AtlasNet results), the color in the chair back is discontinued. Which implies that the near region on the chair back is constructed by faraway points from the template. Moreover, from all the three rows, we can see the middle column results are with the same color direction ($x,y,or z$) as the template, while the right column results are with different color direction. This is because our method progressively deforms the template to fit the target, while AtlasNet directly projects the point’s latent code into 3D space, without coarse-to-fine constraints.

We also provide the numerical results in Table 2. Consider-
Figure 4: Reconstruction comparison with two methods. The proposed PDAE produces shapes with more geometric details and less noisy points. In the top two rows, the comparison areas are highlighted with boxes. The red boxes represent the output results, while the green boxes represent the ground-truth comparison.

Considering the absence of ground-truth dense correspondence, we use \( \text{spread} \) to represent the extent of reconstruction distraction among the nearby points grouped by voxelized grids. Specifically, \( \text{spread} \) is the variance of the reconstructed point locations indexed from the same voxelized grid in the template. \( \text{spread} \) is the average of \( \text{spread} \) throughout the shape. The larger \( \text{spread} \) is, the messier template points are used to reconstruct the shape. We use \( \text{shift} \) to measure the average displacements from the template to the target shape. Our method achieves lower mean \( \text{spread} \) and \( \text{shift} \), which suggests that the correspondence obtained by PDAE preserves better local and global topology.

### 5.4. Ablation Study

In order to verify the effects of the multi-stage supervision on feature learning, the ablation study on the number of deformation stages is conducted. The experiment is carried out in the Chair A dataset. Different number of supervision stages are used for training the network. Detailed settings are as follows:

1) Let \{\} represent the training without multi-stage su-

| Metrics | AtlasNet [6] | PDAE (ours) |
|---------|--------------|-------------|
|         | \( \text{spread} \downarrow \text{shift} \downarrow \) | \( \text{spread} \downarrow \text{shift} \downarrow \) |
| Plane A | 0.40 1.21    | 0.29 0.79   |
| Car A   | 0.17 1.13    | 0.19 0.61   |
| Chair A | 0.48 0.98    | 0.23 0.68   |
| Table A | 0.30 1.07    | 0.16 0.60   |
| Mean    | 0.34 1.10    | 0.22 0.67   |

Table 2: Correspondence evaluation. We denote \( \text{spread} \) as the extent of reconstruction distraction from the template point clouds. We use \( \text{shift} \) to measure the average displacements from the template to the target. For both metrics, the lower the better.
Figure 5: Correspondence comparison between our method and AtlasNet. In AtlasNet, the same spherical point cloud template as ours is used as input. The colors in the spherical template are transferred into the reconstructed chairs. It can be observed that our correspondence derived from PDAE preserves better local and global topology from the template.

2) Let \{16\} represent the two-stage supervised training. In this setting, the coarsest shape \(P^4\) is used as the intermediate supervision for the progressive deformation. The spherical template is deformed to fit the abstract coarse shape first, and then to fit the original shape.

3) Let \{16, 256\} represent the three-stage supervised training. In this setting, the shapes \(P^4\) and \(P^2\) are used as the intermediate supervision for the progressive deformation.

4) Let \{16, 64, 256, 1024\} represent the multi-stage supervision used in the proposed PDAE framework. In this setting, the shapes \(P^4, P^3, P^2\) and \(P^1\) are used as the intermediate supervision for the progressive deformation.

As is shown in Table 3, the reconstruction task benefits from the proposed multi-stage supervision. It can be observed that, with the number of stages increases, there are better numerical evaluations of the reconstruction results.

### 5.5. Additional Applications

#### 5.5.1 Class-defined Region Localization

The Cloud Sphere representation can help localize the class-defined regions. Specifically, the Cloud Sphere of the 3D shape is fed into a PointNet \([18]\) like encoder, through which multi-level features are extracted and concatenated into a comprehensive one. The combined feature is then passed through an squeeze and excitation module \([8]\). Finally a fully-connected layer is used to perform classification.

Fig. 6 shows the class-defined region localization obtained by the classification task. The regions are detected by the point-wise activation in the high attention feature channel. Three fine-grained classes of chairs are illustrated. We highlight the most distinctive regions compared to each class. This will be great convenient for multiple high-level graphic tasks such as style localization, shape transfer, etc.
5.5.2 Deformation-based Shape Editing

Shape Geometry Transfer. This application is based on the inter-class correspondence. We perform shape geometry transfer between different fine-grained classes. Once the distinctive regions are detected, shape interpolation is then conducted based on the detection results. Specifically, only the class-distinctive regions are deformed towards the target. The remaining parts are maintained. Through this manner, fine-grained class or style transfer can be achieved.

The geometry transfer results are illustrated in Fig. 7. We can see arms are added to the proper position of the straight chairs. Note that the straight legs can be deformed to star-like ones.

Co-editing. Using the inner-class correspondence, we can modify the shapes in the same class simultaneously. Fig. 8 illustrates the co-editing results. The arms are added to a set of straight chairs simultaneously.

6. Conclusion

This paper has introduced a novel 3D shape representation method which explicitly encodes the deformation residuals from a spherical surface. The Cloud Sphere representation constructs point-wise correspondence between the shape and the template. Therefore it can serve as a general-purpose representation for various graphic applications. In order to learn this representation, the progressive deformation-based auto-encoder has been proposed. Specifically, a spherical point cloud is deformed progressively from coarse to fine to fit the multi-scale counterparts of the target shape, with the corresponding multi-scale features as guidance. Experimental results show that the proposed PDAE has the ability to reconstruct 3D shapes with high fidelity, and consistent topology is preserved while deformation. Additional applications based on the Cloud Sphere representation have been performed, demonstrating its universality. In future, we plan to extend the multi-stage supervision to the continuous deformation framework.

References

[1] M. Ben-Chen and C. Gotsman. Characterizing shape using conformal factors. In Proceedings of the 1st Eurographics Conference on 3D Object Retrieval, 3DOR ’08, pages 1–8, Aire-la-Ville, Switzerland, Switzerland, 2008. Eurographics Association.

[2] 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.

[3] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud. Neural ordinary differential equations. arXiv preprint arXiv:1806.07366, 2018.

[4] M. Gadelha, R. Wang, and S. Maji. Multiresolution tree networks for 3d point cloud processing. In Proceedings of the European Conference on Computer Vision (ECCV), pages 103–118, 2018.

[5] W. Grathwohl, R. T. Chen, J. Bettencourt, I. Sutskever, and D. Duvenaud. Ffjord: Free-form continuous dynamics for scalable reversible generative models. arXiv preprint arXiv:1810.01367, 2018.

[6] T. Groueix, M. Fisher, V. G. Kim, B. C. Russell, and M. Aubry. A papier-mâché approach to learning 3d surface generation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 216–224, 2018.

[7] Z. Guo and C.-C. Feng. Using multi-scale and hierarchical deep convolutional features for 3d semantic classification of...
ts point clouds. *International Journal of Geographical Information Science*, 34(4):661–680, 2020. 3

[8] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 9

[9] Q. Huang, M. Wicke, B. Adams, and L. Guibas. Shape decomposition using modal analysis. In *Computer Graphics Forum (Proceedings of Eurographics 2009)*, volume 28, pages 407–416, Munchen, Germany, April 2009. 2

[10] C. Jiang, J. Huang, A. Tagliasacchi, L. Guibas, et al. Shapeflow: Learnable deformations among 3d shapes. *arXiv preprint arXiv:2006.07982*, 2020. 4

[11] E. Kalogerakis, M. Averkiou, S. Maji, and S. Chaudhuri. 3D shape segmentation with projective convolutional networks. In *Proc. IEEE Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[12] E. Kalogerakis, A. Hertzmann, and K. Singh. Learning 3d mesh segmentation and labeling. *ACM Trans. Graph.*, 29(4):102:1–102:12, July 2010. 2

[13] S. Katz and A. Tal. Hierarchical mesh decomposition using fuzzy clustering and cuts. *ACM Trans. Graph.*, 22(3):954–961, July 2003. 2

[14] J. Li, K. Xu, S. Chaudhuri, E. Yumer, H. Zhang, and L. Guibas. Grass: Generative recursive autoencoders for shape structures. *ACM Transactions on Graphics (Proc. of SIGGRAPH 2017)*, 36(4):to appear, 2017. 2

[15] X. Liu, Z. Han, Y.-S. Liu, and M. Zwicker. Point2sequence: Learning the shape representation of 3d point clouds with an attention-based sequence to sequence network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8778–8785, 2019. 3

[16] E. Mehr, A. Jourdan, N. Thome, M. Cord, and V. Guittene. Disconet: Shapes learning on disconnected manifolds for 3d editing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3474–3483, 2019. 3

[17] M. Niemeyer, L. Mescheder, M. Oechsle, and A. Geiger. Occupancy flow: 4d reconstruction by learning particle dynamics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5379–5389, 2019. 4

[18] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 2, 3, 9

[19] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in neural information processing systems*, pages 5099–5108, 2017. 2, 3, 4

[20] C. Qin, H. You, L. Wang, C.-C. J. Kuo, and Y. Fu. Pointdan: A multi-scale 3d domain adaption network for point cloud representation. In *Advances in Neural Information Processing Systems*, pages 7192–7203, 2019. 3

[21] L. Shapira, S. Shalom, A. Shamir, D. Cohen-Or, and H. Zhang. Contextual part analogies in 3d objects. *Int. J. Comput. Vision*, 89(2-3):309–326, Sept. 2010. 2

[22] A. Sinha, A. Umesh, Q. Huang, and K. Ramani. Surfnet: Generating 3d shape surfaces using deep residual networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6040–6049, 2017. 3

[23] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y.-G. Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *ECCV*, 2018. 2, 3

[24] P.-S. Wang, Y. Liu, Y.-X. Guo, C.-Y. Sun, and X. Tong. O-cnn: Octree-based convolutional neural networks for 3d shape analysis. *ACM Transactions on Graphics (TOG)*, 36(4):1–11, 2017. 7

[25] W. Wang, D. Ceylan, R. Mech, and U. Neumann. 3dn: 3d deformation network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1038–1046, 2019. 3

[26] Z. Wang and F. Lu. Voxsegnet: Volumetric cnns for semantic part segmentation of 3d shapes. *IEEE transactions on visualization and computer graphics*, 2019. 2

[27] Y.-P. Xiao, Y.-K. Lai, F.-L. Zhang, C. Li, and L. Gao. A survey on deep geometry learning: From a representation perspective. *Computational Visual Media*, 6(2):113–133, 2020. 3

[28] G. Yang, X. Huang, Z. Hao, M.-Y. Liu, S. Belongie, and B. Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4541–4550, 2019. 4

[29] Y. Yang, C. Feng, Y. Shen, and D. Tian. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 206–215, 2018. 3

[30] L. Yi, H. Su, X. Guo, and L. Guibas. Syncspectncn: Synchronized spectral cnn for 3d shape segmentation. *arXiv preprint arXiv:1612.00606*, 2016. 2

[31] K. Yin, Z. Chen, H. Huang, D. Cohen-Or, and H. Zhang. Logan: Unpaired shape transform in latent overcomplete space. *ACM Transactions on Graphics (TOG)*, 38(6):1–13, 2019. 3, 7

[32] K. Yin, H. Huang, D. Cohen-Or, and H. Zhang. P2p-net: Bidirectional point displacement net for shape transform. *ACM Transactions on Graphics (TOG)*, 37(4):1–13, 2018. 3

[33] M. E. Yumer and N. J. Mitra. Learning semantic deformation flows with 3d convolutional networks. In *European Conference on Computer Vision*, pages 294–311. Springer, 2016. 3

[34] A. K. F. Y. L. Z. X. T. J. X. Z. Wu, S. Song. 3d shapenets: A shape representation for 3d object analysis. *Computational Visual Media*, 6(2):113–133, 2020. 3

[35] J. Zhang, J. Zheng, C. Wu, and J. Cai. Variational mesh editing on point cloud auto-encoder via deep grid deformation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 206–215, 2018. 3