GenAnalysis: Joint Shape Analysis by Learning Man-Made Shape Generators with Deformation Regularizations

YUEZHI YANG, The University of Texas at Austin, USA
HAITAO YANG, The University of Texas at Austin, USA
GEORGE NAKAYAMA, Stanford University, USA
XIANGRU HUANG, Westlake University, China
LEONIDAS GUIBAS, Stanford University, USA
QIXING HUANG, The University of Texas at Austin, USA

We present GenAnalysis, an implicit shape generation framework that allows joint analysis of man-made shapes, including shape matching and joint shape segmentation. The key idea is to enforce an as-affine-as-possible (AAAP) deformation between synthetic shapes of the implicit generator that are close to each other in the latent space, which we achieve by designing a regularization loss. It allows us to understand the shape variation of each shape in the context of neighboring shapes and also offers structure-preserving interpolations between the input shapes. We show how to extract these shape variations by recovering piecewise affine vector fields in the tangent space of each shape. These vector fields provide single-shape segmentation cues. We then derive shape correspondences by iteratively propagating AAAP deformations across a sequence of intermediate shapes. These correspondences are then used to aggregate single-shape segmentation cues into consistent segmentations. We conduct experiments on the ShapeNet dataset to show superior performance in shape matching and joint shape segmentation over previous methods.

CCS Concepts: • **Computing methodologies** → **Shape analysis**; • **Neural networks**;

Additional Key Words and Phrases: shape correspondence, joint shape segmentation, neural network regularization

ACM Reference Format:

Yuezhi Yang, Haitao Yang, George Nakayama, Xiangru Huang, Leonidas Guibas, and Qixing Huang. 2025. GenAnalysis: Joint Shape Analysis by Learning Man-Made Shape Generators with Deformation Regularizations. *ACM Trans. Graph.* 1, 1 (February 2025), 21 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Shape analysis, is a fundamental research area in geometry processing that enjoys many applications [Mitra et al. 2014; Xu et al. 2017].

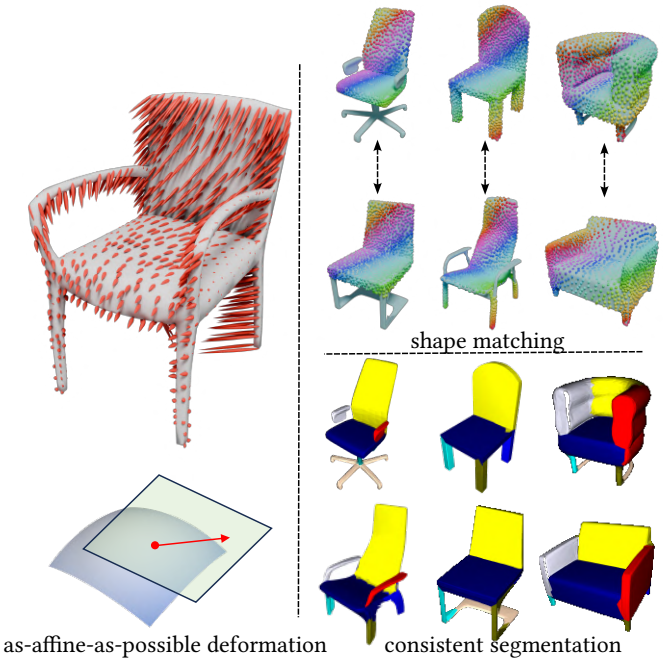

Fig. 1. GenAnalysis learns a shape manifold from a collection of man-made shapes with a novel as-affine-as-possible deformation regularization loss. The learned shape manifold supports shape analysis by analyzing the tangent space of each shape, shape matching through intermediate shapes on this manifold, and consistent segmentation by aggregating single-shape analysis results using inter-shape correspondences.

Existing shape analysis approaches fall into the categories of supervised and unsupervised methods. Although supervised methods achieved state-of-the-art results, they require user labels, which are costly to obtain and do not scale up. In contrast, unsupervised shape analysis methods [Chen et al. 2024; Deng et al. 2021; Shuai et al. 2023] benefit from learning from large-scale unlabeled shapes and have recently shown promising results in the large-scale ShapeNet-Part [Yi et al. 2016] dataset.

The fundamental challenge in developing unsupervised techniques is to accommodate significant structural and geometric variations across the input shapes. A popular paradigm is learning template-based models in which each template consists of a bag of (non-rigidly) deforming parts. The learning phase learns parameters of the templates to fit the training data. These include early works

Authors' addresses: Yuezhi Yang, The University of Texas at Austin, 2317 Speedway, Austin, Texas, 78712, USA; Haitao Yang, The University of Texas at Austin, 2317 Speedway, Austin, Texas, 78712, USA; George Nakayama, Stanford University, Palo Alto, California, USA; Xiangru Huang, Westlake University, Palo Alto, California, China; Leonidas Guibas, Stanford University, Palo Alto, California, USA; Qixing Huang, The University of Texas at Austin, 2317 Speedway, Austin, Texas, 78712, USA.

0730-0301/2025/2-ART $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

of [Averkiou et al. 2014; Huang et al. 2015a,b; Kim et al. 2013; Ovsjanikov et al. 2011] and recent deep learning methods [Chen et al. 2024; Deng et al. 2021; Deprelle et al. 2019; Ganapathi-Subramanian et al. 2018; Genova et al. 2019a; Shuai et al. 2023; Zheng et al. 2021]. The key advantage of template-based models is that during the testing phase it is only required to fit template to each shape, which is efficient and simple. However, template-based models exhibit two major limitations. First, template-based models rely on prior knowledge of shape parts (e.g., the number of parts and their spatial locations) either explicitly or implicitly. Second, the learning and inference phases can easily get stuck in local minimums.

In this paper, we introduce a novel shape analysis framework, named GenAnalysis, which learns an implicit shape generative model to fit the training shapes. We will show how to use this template-free shape generative model to perform shape analysis, hence circumventing the limitation of template-based models. Our approach is inspired by the success of studying organic shape collections using the lens of shape spaces as differential manifolds [Hartman et al. 2023; Kilian et al. 2007; Klassen et al. 2004; Srivastava et al. 2011; Yang et al. 2023a]. Moreover, recent advances in shape generative models [Atzmon and Lipman 2021; Chen and Zhang 2019; Mescheder et al. 2019; Park et al. 2019; Yang et al. 2024] can accurately fit man-made training shapes that exhibit much larger geometric variations than organic shapes. These learned continuous man-made shape spaces offer novel means of shape analysis. Specifically, they provide smooth interpolations between two shapes that can be used to compute shape correspondences. Moreover, when trained properly,these generative models enable us to understand the shape variations among the shape collection for shape segmentation by analyzing shape variations in the tangent space of each shape when viewing the learned generative models as shape manifolds. This continuous shape-space approach greatly alleviates the local minimum issue in template-based approaches.

However, learning a shape generative model is under-constrained, as there are many possible network parameters and latent codes of training shapes that fit the training data. Learning shape generative models under generic distribution alignment paradigms [Goodfellow et al. 2020; Kingma and Welling 2014; Zhang et al. 2023] do not result in shape generators that offer meaningful intermediate shapes and tangent spaces for shape matching and shape segmentation. Unlike deformable objects in which isometric deformations offer faithful deformation priors, the structural shape variations in man-made shapes are too complex to be summarized into a concise universal principle. The suitable regularization losses are heavily dependent on the application scenarios.

GenAnalysis introduces a novel regularization loss for learning the shape generator. This loss is tailored for shape analysis. The proposed loss builds on the popular piece-wise affine assumption [Chen et al. 2024; Huang et al. 2015a; Kim et al. 2013; Ovsjanikov et al. 2011; Xu et al. 2010], in which **if we abstract each shape part using a bounding box, then the deformations at the part level are affine** (see Figure 2). However, the challenge is how to model the significant shape variations within each part. GenAnalysis presents two strategies to address this challenge. First, we minimize the piece-wise affine deformation between adjacent synthetic shapes

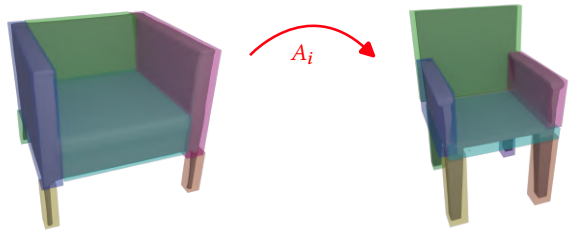

Fig. 2. **Piece-wise affine assumption.** Each shape part from source shape, approximated by its bounding box, undergoes a separate affine transformation A_i to the corresponding part in the target shape.

defined by the generator (i.e., they are close in the latent space). This approach is analogous to the smoothness loss in curve fitting, but we model strong prior knowledge about inter-shape deformations. In particular, we show how to employ an as-affine-as-possible deformation loss under a robust norm to fulfill the piece-wise affine assumption. Second, we describe a weighting scheme for test-time optimization [Chen et al. 2022; Niu et al. 2022b; Wang et al. 2021, 2023]. It prioritizes that the regularization loss distributes deviations of piece-wise affine deformations away from shapes of interest. This allows us to extract segmentation cues by analyzing the tangent space of each shape (i.e., vector fields that characterize differences between each shape and its adjacent shapes in the latent space).

We have evaluated the performance of GenAnalysis on ShapeNet for the task of shape matching and consistent shape segmentation. In both tasks, GenAnalysis outperforms state-of-the-art techniques by salient margins, justifying the power of performing shape analysis by learning shape generators. For example, GenAnalysis outperforms DAE-Net [Chen et al. 2024] by 3.2% in mean IOU score. An ablation study justifies the importance of each component of GenAnalysis.

2 RELATED WORK

2.1 Neural Generative Models for Man-Made Shapes

Recent advances in learning generative shape models have shown impressive results in shape synthesis. These approaches have focused on developing network architectures under different 3D representations [Achlioptas et al. 2018; Li et al. 2017; Longwen et al. 2024; Mo et al. 2019; Park et al. 2019; Siddiqui et al. 2024; Wu et al. 2016]. A common scheme is to align the distribution of the training shapes with that of the synthetic shapes [Arjovsky et al. 2017; Bouritsas et al. 2019; Goodfellow et al. 2020; Litany et al. 2018; Park et al. 2019; Ranjan et al. 2018; Song and Ermon 2019; Song et al. 2021; Yang et al. 2018; Zadeh et al. 2019]. However, these approaches are limited when the training shapes are sparse. An approach to address this data issue is to develop regularization losses that enforce prior knowledge about the underlying shape space [Atzmon et al. 2021; Gropp et al. 2020; Huang et al. 2021; Liu et al. 2022; Muralikrishnan et al. 2022; Yang et al. 2024]. The novelties of GenAnalysis are addressing man-made shapes that have large shape variations and learning the shape generator to promote simple extractions of shape segmentation cues in the tangent space of each shape.

2.2 Data-Driven Shape Segmentation

A common paradigm among non-deep learning-based joint shape segmentation approaches [Golovinskiy and Funkhouser 2009; Hu et al. 2012; Huang et al. 2011, 2019, 2014; Sidi et al. 2011; Wang et al. 2013, 2014] is to promote the segmentation of each shape to be consistent in a shape collection, using point-wise correspondences [Huang et al. 2011], primitive-level correspondences [Hu et al. 2012; Huang et al. 2011; Sidi et al. 2011], or functional maps [Huang et al. 2019, 2014; Ovsjanikov et al. 2012; Wang et al. 2013, 2014]. However, these approaches only exhibit limited performance in heterogeneous shape collections, where computing consistent correspondences between structurally dissimilar shapes is challenging.

Deep learning based methods are introduced to handle joint shape segmentation for its improved generalization ability on heterogeneous shape collection. Most of the methods for learning consistent segmentation are supervised learning [Kalogerakis et al. 2017; Qi et al. 2017a,b; Zhao et al. 2019]. To alleviate the need for 3D annotation, unsupervised methods learn deformable part templates to reconstruct input shapes for shape segmentation. These approaches use a variety of part abstractions, including cuboids [Sun et al. 2019; Tulsiani et al. 2017; Yang and Chen 2021], superquadratics [Paschalidou et al. 2019], part point clouds [Huang et al. 2015a], convex polygons [Chen et al. 2020; Deng et al. 2020], sphere templates [Paschalidou et al. 2021] and implicit templates [Chen et al. 2024, 2019; Niu et al. 2022a]. Part deformations can also be encoded explicitly [Huang et al. 2015a] or implicitly using neural network branches [Chen et al. 2024, 2019; Paschalidou et al. 2021; Shuai et al. 2023; Tertikas et al. 2023]. However, they still require some prior knowledge of the underlying part structure, e.g., the number of parts. The designed template often exhibits limited performance when expressing complex variations in each shape part. GenAnalysis circumvents the limitations of template-based approaches in two ways. First, as we will discuss next, GenAnalysis presents dense shape correspondences between shapes with large structural variations to aggregate shape segmentation cues. Second, GenAnalysis enforces the prior knowledge about shape parts using a regularization loss. The regularization loss is enforced on adjacent synthetic shapes and tolerates large inter-shape variations.

Several approaches, including STAR [Abdelreheem et al. 2023], 3D Highlighter [Decatur et al. 2023], and follow-up methods [Cen et al. 2023; Kontogianni et al. 2023; Lang et al. 2024; Yang et al. 2023b], have explored 3D segmentation using vision-language models (VLM) [Alayrac et al. 2022; Ravi et al. 2024; Zhang et al. 2022]. The key idea is to lift 2D features from off-the-shelf CLIP [Radford et al. 2021], GLIP [Li et al. 2022; Zhang et al. 2022], and SAM [Kirillov et al. 2023] models. One challenge of these approaches is to enforce multi-view consistency and address occlusions. Complementary to lifting 2D features, GenAnalysis is a 3D approach that focuses on consistent segmentation using a shape generative model.

2.3 Shape Matching and Deformation Models

Computing correspondences between geometric shapes is a long-standing problem in computer graphics. It is beyond the scope of this paper to provide a comprehensive review. We refer to [Sahillioglu 2020; Tam et al. 2013b; van Kaick et al. 2010] for surveys on this topic. Most approaches [Boscaini et al. 2016; Kim et al. 2011; Melzi et al. 2019] deal with deformable shapes assume that they have consistent topologies and full dense correspondences are well-defined. Such approaches do not apply to man-made shapes with complex structure variations where dense correspondences are partially defined.

A closely related problem to correspondence computation is non-rigid shape registration [Tam et al. 2013a], which deforms a source shape under some deformation model to match a target shape. In this context, the deformation model plays a key role. The most widely used deformation models are as-rigid-as-possible (ARAP) [Alexa et al. 2000; Huang et al. 2021; Sorkine and Alexa 2007] and as-conformal-as-possible (ACAP) [Yang et al. 2023a; Yoshiyasu et al. 2014] models. These models are mostly used for organic shapes but not for man-made shapes that exhibit sophisticated part deformations. For man-made shapes, a widely used assumption is piece-wise affine, which has been used in template-based models, that consists of rectangular bounding boxes. GenAnalysis shows how to apply the idea of non-rigid shape registration to find correspondences across man-made shapes. This is achieved by using a robust as-affine-as-possible deformation model to enforce the piece-wise affine deformation and performing non-rigid registrations between adjacent synthetic shapes of a man-made shape generator. The idea of optimizing an interpolation between two shapes for shape matching has also been explored in NeuroMorph [Eisenberger et al. 2021] and GenCorres [Yang et al. 2024], both of which study organic shapes. In contrast, GenAnalysis focuses on man-made shapes which have much more complex geometric variations, where the deformation models of NeuroMorph and GenCorres do not apply.

In contrast to template-based approaches [Deng et al. 2021; Genova et al. 2019b; Kim et al. 2023; Zheng et al. 2021] for made-made shapes, GenAnalysis exhibits improved performance on heterogeneous shape collections. This is because template-based models may not be expressive enough and/or the learning procedure can get trumped into local minimums.

3 PROBLEM STATEMENT AND APPROACH OVERVIEW

This section presents the problem statement (Section 3.1) and an overview of GenAnalysis (Section 3.2).

3.1 Problem Statement

The input to GenAnalysis is a collection of training man-made shapes $\mathcal{S} = \{S_i\} \subset \overline{\mathcal{S}}$ where $\overline{\mathcal{S}}$ denotes the ambient shape space. GenAnalysis aims to learn a generative model that enables two fundamental shape analysis tasks on a collection of test shapes $\mathcal{S}_{\text{test}}$:

- *Joint Shape Segmentation.* For each test shape $S \in \mathcal{S}_{\text{test}}$, we want to decompose it into parts based on the underlying shape variations provided by $\overline{\mathcal{S}}$.
- *Shape Correspondence.* Given a test source shape $S \in \mathcal{S}_{\text{test}}$, a test target shape $S' \in \mathcal{S}_{\text{test}}$, and a point $\boldsymbol{p}$ in S, we want to calculate its correspondence in S'. The output indicates whether this correspondence is well defined (due to partial similarities) and, if so, the corresponding point.

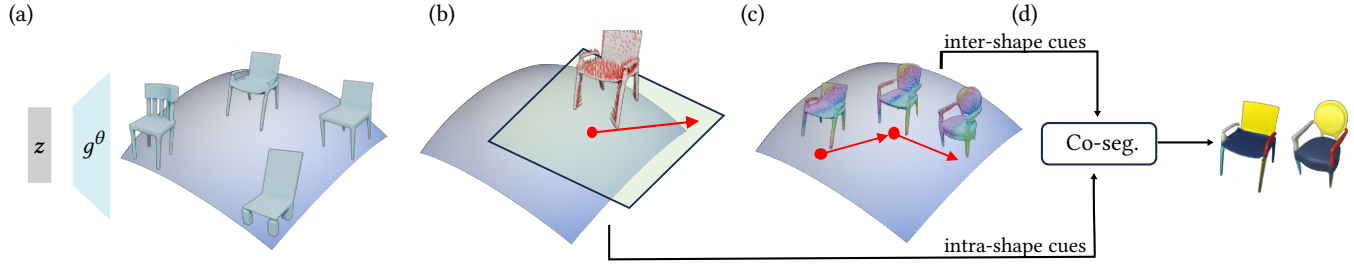

Fig. 3. **The pipeline of GenAnalysis pipeline**, which consists of four stages. (a) The first stage learns an implicit shape generator to fit the input shapes by combing a data loss and an as-affine-as possible deformation loss. (b) The second stage extracts piece-wise affine structures in vector fields of each shape derived from the tangent space of each shape. (c) The third stage computes pairwise shape correspondences obtained by composing correspondences along intermediate shapes defined by the generator. Points with similar color are in correspondence. (d) The last stage performs consistent segmentation using the correspondences obtained in stage three to integrate single-shape segmentation cues derived from stage two.

3.2 Approach Overview

GenAnalysis proceeds in four stages (see Figure 3 for an illustration).

3.2.1 Shape Generator Learning. The first stage learns an implicit shape generator from $\mathcal{S}$. We introduce a novel regularization loss to enforce piece-wise affine deformations between synthetic shapes that are close to each other in the latent space. The loss employs an as-affine-as-possible (AAAP) deformation model under the L^2 norm for correspondence computation and the same AAAP model under a robust norm to enforce piece-wise affine deformations. Network training integrates this regularization loss into a standard auto-decoder paradigm. In addition, we introduce a light-weight test-time optimization step to improve the alignment between the generative model and test shapes and distribute distortions of the piece-wise affine assumption away from the tangent spaces at test shapes. This weighting scheme facilitates the analysis of the tangent space at each test shape to extract shape segmentation cues.

3.2.2 Shape Variation Analysis. The second stage analyzes the tangent spaces of the learned shape manifold. The tangent space of each shape encodes the shape variations in its infinitesimal neighboring shape space. The regularization loss employed in stage one prioritizes that this tangent space contains variation vector fields that exhibit piece-wise affine structures. We present a spectral approach to extract such vector fields and show how to analyze their underlying piece-wise affine structures to derive a distance matrix, in which the distance between two points is small if they belong to the same part and vice versa. Figure 8 (Left) shows some examples of the extracted vector fields.

3.2.3 Shape Matching. The third stage of GenAnalysis computes shape correspondences between pairs of test shapes. We show that unlike matching pairs of shapes directly, which leads to poor results under large shape variations, GenAnalysis propagates correspondences derived from AAAP deformation between intermediate shapes interpolated from the learned deformation model. The propagation procedure combines displacement and projection operations to ensure that the propagated correspondences lie on intermediate shapes. The resulting correspondences will also be used to aggregate single-shape segmentation cues from the second stage.

3.2.4 Consistent Segmentation. The last stage of GenAnalysis performs a consistent segmentation. We formulate a spectral-based consistent segmentation approach that integrates the single-shape segmentations derived from shape variation analysis using inter-shape correspondences.

4 APPROACH

This procedure describes four stages of GenAnalysis in detail (from Section 4.1 to Section 4.4).

4.1 Shape Generator Learning

The main goal for learning the shape generative model is to enforce a deformation loss between adjacent synthetic shapes. Implicit deformation regularizations exist in the literature. One notable example is the Killing formulation [Kraevoy et al. 2008; Slavcheva et al. 2017] for as-rigid-as-possible (ARAP) deformation, in which the Jacobian of the implicit field is askew-symmetric. However, this approach does not apply to the affine setting, in which we do not have any constraint on the Jacobian. To address this issue, our approach turns an implicit generation model into an explicit one locally by computing dense correspondences between adjacent synthetic surfaces (see Section 4.1.1). These correspondences give us flexibility in enforcing the piece-wise affine assumption, and we define a regularization loss using a robust AAAP model to learn the shape generator (see Section 4.1.2). In addition, we present a lightweight test-time optimization strategy for test shapes in Section 4.1.3.

4.1.1 AAAP deformation induced correspondences. We first explain how to compute the correspondences between adjacent surfaces of an implicit generator model $g^\theta(\boldsymbol{x}, \boldsymbol{z}) : \mathbb{R}^3 \times \mathcal{Z} \rightarrow \mathbb{R}$ that outputs the signed distance function (SDF) value at $\boldsymbol{x}$ of the underlying 3D shape with the latent code $\boldsymbol{z} \in \mathcal{Z} \cong \mathbb{R}^q$ ($q = 256$ in our experiments). We focus on an AAAP model for man-made shapes.

To this end, we first discretize $g^\theta(\boldsymbol{x}, \boldsymbol{z}) = 0$ using a mesh with $n \approx 2000$ vertices $\boldsymbol{p}_i^\theta(\boldsymbol{z}), 1 \leq i \leq n$, e.g., via Marching Cube [Lorensen and Cline 1987]. To define the AAAP deformation model, our goal is to determine the corresponding location $\boldsymbol{p}_i^\theta(\boldsymbol{z} + \epsilon\boldsymbol{v})$ on the neighboring implicit surface $g^\theta(\boldsymbol{x}, \boldsymbol{z} + \epsilon\boldsymbol{v}) = 0$, where $\boldsymbol{v}$ is the direction of the perturbation and $\epsilon = 10^{-3}$ is an infinitesimal value. We approximate $\boldsymbol{p}_i^\theta(\boldsymbol{z} + \epsilon\boldsymbol{v})$ via the first-order Taylor expansion

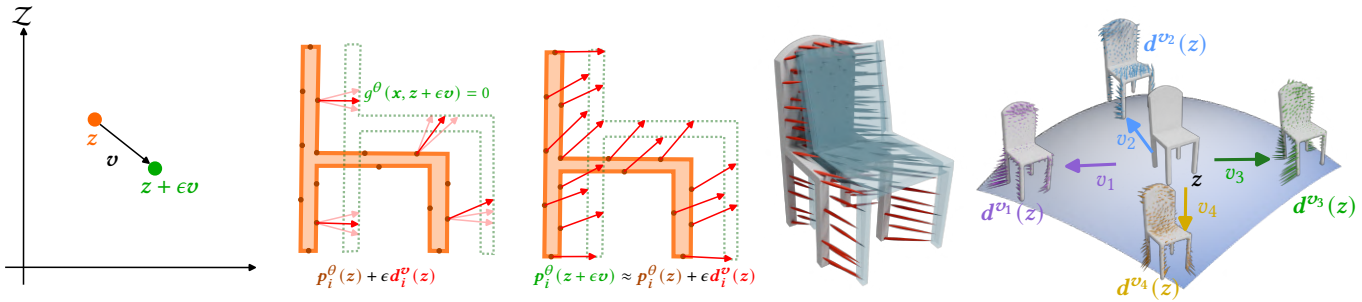

Fig. 4. **As-affine-as-possible (AAAP) regularization.** (a) We study infinitesimal perturbation $\boldsymbol{v}$ in the tangent space at each shape with latent code $\boldsymbol{z}$. (b) Due to constraint shown in Eq. (1), we can not determine the correspondence $\boldsymbol{d}_i^{\boldsymbol{v}}(\boldsymbol{z})$ that lies on $g^\theta(\boldsymbol{x}, \boldsymbol{z}+\epsilon\boldsymbol{v}) = 0$ directly. (c) We instead jointly compute all $\boldsymbol{d}_i^{\boldsymbol{v}}(\boldsymbol{z})$ by solving an constrained optimization problem using the objective function in Eq. (2). (d) We show resulting 3D correspondences between source shape colored in white, and a neighboring perturbed shape colored in transparent blue. (e) After derivation in section 4.1.1, we arrived at closed form solution shown in Eq. (5) where each perturbation v_i corresponds to variation at $\boldsymbol{d}^{\boldsymbol{v_i}}(\boldsymbol{z})$. We integrate over all directions to obtain the regularization term in Eq. (8).

$\boldsymbol{p}_i^\theta(\boldsymbol{z}+\epsilon\boldsymbol{v}) \approx \boldsymbol{p}_i^\theta(\boldsymbol{z}) + \epsilon\boldsymbol{d}_i^{\boldsymbol{v}}(\boldsymbol{z})$. By computing the derivative of the implicit function $g^\theta(\boldsymbol{p}_i^\theta(\boldsymbol{z}+\epsilon\boldsymbol{v}), \boldsymbol{z}+\epsilon\boldsymbol{v}) = 0$ with respect to $\boldsymbol{v}$ using the chain rule, we arrive at the following linear constraint of $\boldsymbol{d}_i^{\boldsymbol{v}}(\boldsymbol{z})$:

$$\frac{\partial g^\theta}{\partial \boldsymbol{x}}\left(\boldsymbol{p}_i^\theta(\boldsymbol{z}), \boldsymbol{z}\right)^T \boldsymbol{d}_i^{\boldsymbol{v}}(\boldsymbol{z}) + \frac{\partial g^\theta}{\partial \boldsymbol{z}}\left(\boldsymbol{p}_i^\theta(\boldsymbol{z}), \boldsymbol{z}\right)^T \boldsymbol{v} = 0. \tag{1}$$

As shown in Figure 4(b), the technical challenge is that the implicit surface representation only provides one constraint on the 3D displacement vector $\boldsymbol{d}_i^{\boldsymbol{v}}(\boldsymbol{z})$. To address the uniqueness problem, we solve a linearly constrained optimization problem [Tao et al. 2016; Yang et al. 2024] to find $\boldsymbol{d}_i^{\boldsymbol{v}}(\boldsymbol{z})$ jointly.

Specifically, we associate each vertex $\boldsymbol{p}_i^\theta(\boldsymbol{z})$ with a local transformation $I_3 + A_i \in \mathbb{R}^{3\times3}$, where A_i encodes the deviation of this transformation from the identity transformation. For the sake of optimization, we reparameterize $A_i = s_i I_3 + \boldsymbol{c}_i \times + R(\boldsymbol{a}_i)$, where $s_i I_3$ and $\boldsymbol{c}_i\times$ define scaling and rotation components, respectively; $R(\boldsymbol{a}_i)$ denotes the remaining component in the matrix space that is orthogonal to conformal matrices, and $\boldsymbol{a}_i \in \mathbb{R}^5$ is the coefficient vector under an orthonormal basis of this space (see Appendix A in the supp. material). A nice property of this parametrization is that it is easy to regularize A_i using quadratic objective terms.

Let $\mathcal{N}_i$ denote the neighbors of the vertex i and itself. Let vectors $\boldsymbol{p}^\theta(\boldsymbol{z}) \in \mathbb{R}^{3n}$ and $\boldsymbol{d}^{\boldsymbol{v}}(\boldsymbol{z}) \in \mathbb{R}^{3n}$ concatenate $\boldsymbol{p}_i^\theta(\boldsymbol{z})$ and $\boldsymbol{d}_i^{\boldsymbol{v}}(\boldsymbol{z})$, respectively. We model the deformation energy between them as

$$e\left(\boldsymbol{p}^\theta(\boldsymbol{z}), \boldsymbol{d}^{\boldsymbol{v}}(\boldsymbol{z})\right) := \min_{\{A_i\}} \sum_{i=1}^{n} \Big(\sum_{j\in\mathcal{N}_i} \|A_i(\boldsymbol{p}_i^\theta(\boldsymbol{z}) - \boldsymbol{p}_j^\theta(\boldsymbol{z})) - (\boldsymbol{d}_i^{\boldsymbol{v}}(\boldsymbol{z}) - \boldsymbol{d}_j^{\boldsymbol{v}}(\boldsymbol{z}))\|^2 + (\mu_r s_i^2 + \mu_s \|\boldsymbol{a}_i\|^2) \Big) \tag{2}$$

where $\mu_r = 1$ and $\mu_s = 1$ are regularization parameters. Our experiments show that the regularization effects remain similar in a large range of μ_r and μ_s. On the other hand, regularization is important since otherwise A_i maybe degenerate, e.g., on flat regions.

We introduce $\boldsymbol{y} \in \mathbb{R}^{9n}$ where $\boldsymbol{y}_i = (s_i; \boldsymbol{c}_i; \boldsymbol{a}_i)$ collects the affine transformation parameters for each A_i at $\boldsymbol{p}_i^\theta(\boldsymbol{z})$. It is clear that $e\left(\boldsymbol{p}^\theta(\boldsymbol{z}), \boldsymbol{d}^{\boldsymbol{v}}(\boldsymbol{z})\right)$ is quadratic in $\boldsymbol{y}$ and $\boldsymbol{d}^{\boldsymbol{v}}(\boldsymbol{z})$. Therefore, the resulting $e\left(\boldsymbol{p}^\theta(\boldsymbol{z}), \boldsymbol{d}^{\boldsymbol{v}}(\boldsymbol{z})\right)$ could be written as

$$e\left(\boldsymbol{p}^\theta(\boldsymbol{z}), \boldsymbol{d}^{\boldsymbol{v}}(\boldsymbol{z})\right) = \boldsymbol{d}^{\boldsymbol{v}}(\boldsymbol{z})^T L^\theta(\boldsymbol{z}) \boldsymbol{d}^{\boldsymbol{v}}(\boldsymbol{z}), \tag{3}$$

and the optimal solution

$$\boldsymbol{y}^\star = B^\theta(\boldsymbol{z}) \boldsymbol{d}^{\boldsymbol{v}}(\boldsymbol{z}) \tag{4}$$

where both $L^\theta(\boldsymbol{z}) \in \mathbb{R}^{3n\times3n}$ and $B^\theta(\boldsymbol{z}) \in \mathbb{R}^{9n\times3n}$ are sparse matrices (see Appendix A in the supp. material).

With this setup, we can then minimize the quadratic energy in Eq. (3) with respect to the linear constraints in Eq. (1). This leads to the optimal displacement vector

$$\boldsymbol{d}^{\boldsymbol{v}}(\boldsymbol{z}) = M^\theta(\boldsymbol{z}) \boldsymbol{v}. \tag{5}$$

where $M^\theta(\boldsymbol{z}) \in R^{3n\times q}$ can be computed efficiently by LU-factorization of a sparse matrix (see Appendix A in the supp. material).

Figure 5 shows the effects of this AAAP deformation model to find correspondences between shapes that exhibit large piece-wise affine deformations. Note that even if the term residuals of a piece-wise affine deformation exhibit a heavy-tail distribution and should be modeled using a robust norm, the resulting correspondences are accurate under the L^2 norm. However, we will see immediately that using a robust norm to formulate the deformation loss on the correspondences for learning the shape generator is important.

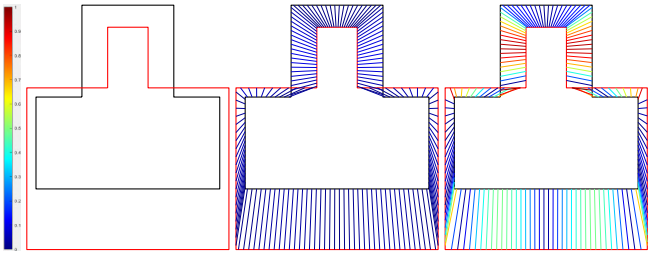

Fig. 5. Effectiveness of the AAAP formulation for identifying correspondences between two implicit shapes that have two parts, each of which undergoes an affine transformation. Correspondence errors are color-coded. (Left) The source shape is colored in black. The target shape is colored in red. (Middle) Deform the source shape to align with the target shape under the AAAP model, resulting in accurate correspondences. (Right) Correspondences drift under the ACAP model, i.e., $\boldsymbol{a}_i = 0$. Errors are color coded.

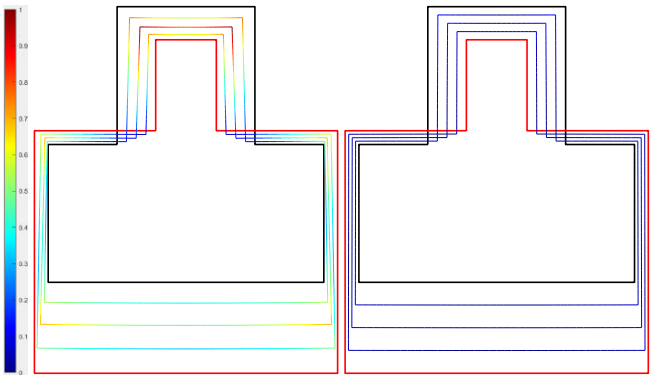

Fig. 6. Comparison between using different norms to optimize the interpolation between two implicit shapes in Figure 5. Deviations from the underlying piece-wise affine interpolation are color coded. (Left) Interpolation using the L^2 norm deviates from piece-wise affine. (Right) Interpolation using the robust norm is piecewise affine.

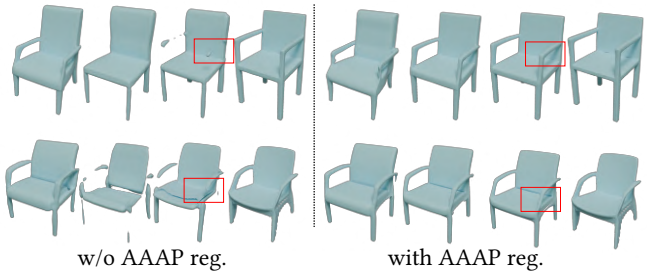

Fig. 7. Shape interpolations from the shape generator. (Left)Without AAAP regularization, the intermediate shapes may not preserve part structures. (Right) With AAAP regularization, the part structures are preserved.

4.1.2 Shape Generator Learning. The training objective for learning the implicit shape generator g^θ is given by

$$\min_\theta \quad \frac{1}{|\mathcal{S}|}\sum_{S\in\mathcal{S}_{\text{train}}} l_{\text{data}}\big(S, g^\theta(\cdot, z_S)\big) + \lambda_{\text{KL}} l_{\text{KL}}\big(\{z_S\}, \mathcal{N}(\mathbf{0}, I_d)\big) + \lambda_d \mathop{\mathbb{E}}_{z\sim\mathcal{N}(\mathbf{0},I_q)} r(\theta, z). \tag{6}$$

The first term $l_{\text{data}}(\cdot,\cdot)$ in Eq. (6) aligns the generator with the input shapes. Given an input shape S, let $\mathcal{P}_S = \{(\boldsymbol{p}, s)\}$ collect samples $\boldsymbol{p}$ and their corresponding SDF values s using the DeepSDF [Park et al. 2019] strategy. We define the data term as

$$l_{\text{data}}(S, g^\theta(\cdot, z_S)) = \sum_{(\boldsymbol{p},s)\in P_S} |g^\theta(\boldsymbol{p}, z_S) - s|^2. \tag{7}$$

The second term $l_{\text{KL}}(\cdot,\cdot)$ aligns the empirical latent distribution defined by the training shapes and the prior Gaussian distribution under the KL divergence measure. This enables us to define regularizations on latent codes sampled from the Gaussian distribution.

The third term $r(\cdot,\cdot)$, which is a key contribution of GenAnalysis, enforces AAAP deformations between adjacent synthetic shapes. To formulate $r(\cdot,\cdot)$, we use Eq. (5), which offers an explicit parameterization of $g^\theta(\boldsymbol{x}, \boldsymbol{z}') = 0$ for latent codes $\boldsymbol{z}'$ in the local neighborhood of $\boldsymbol{z}$. Let $\{A_i\}$ be the latent transformations decoded from Eq. (4).

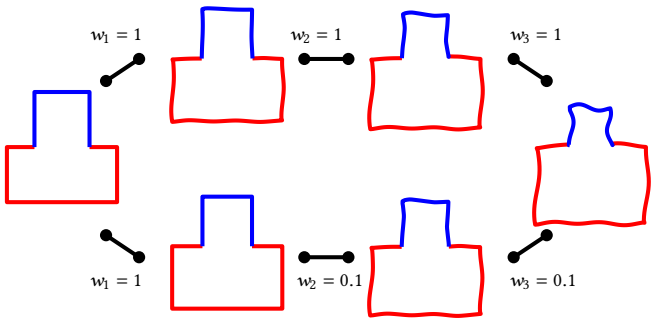

Fig. 8. Interpolations between the left shape and the right shape that using a path of two intermediate shapes. The underlying deformation deviates from the piece-wise affine assumption. We minimize AAAP deformations between adjacent shapes. (Top) Using uniform weights $(1, 1, 1)$ for three shape pairs. The deformations of all shape pairs deviate from piece-wise affine. (Bottom) The weights are $(1, 0.1, 0.1)$ where the first pair has a large weight. The deformation of the first pair is close to piece-wise affine.

We define the structure-preserving regularization term $r(\theta, \boldsymbol{z})$ to enforce that $\boldsymbol{d}^{\boldsymbol{v}}(\boldsymbol{z})$ admits a piece-wise affine structure via

$$r(\theta, \boldsymbol{z}) := \int_{\boldsymbol{v}\in\mathcal{B}^q} \sum_{i=1}^{n} \sum_{j\in\mathcal{N}_i} r_{ij}^\alpha(\boldsymbol{z}, \boldsymbol{v}) d\boldsymbol{v}, \tag{8}$$

$$r_{ij}(\boldsymbol{z}, \boldsymbol{v}) := \|A_i\big(\boldsymbol{p}_i^\theta(\boldsymbol{z}) - \boldsymbol{p}_j^\theta(\boldsymbol{z})\big) - \big(\boldsymbol{d}_i^{\boldsymbol{v}}(\boldsymbol{z}) - \boldsymbol{d}_j^{\boldsymbol{v}}(\boldsymbol{z})\big)\|.$$

where $\mathcal{B}^q$ is the unit ball in $\mathbb{R}^q$. We set $\alpha = 1$ to promote a heavy-tailed distribution in $r_{jk}^\alpha(\boldsymbol{z}, \boldsymbol{v})$, which models piece-wise affine deformations. Figure 6 shows that the robust norm is critical in obtaining the underlying piece-wise affine interpolations between two implicit shapes.

Figure 7 shows the effectiveness of this regularization term, which preserves part structures. In particular, it addresses the challenge of preserving thin-structures under implicit representations.

4.1.3 Test-Time Optimization. Inspired by the success of test-time optimization [Chen et al. 2022; Niu et al. 2022b; Wang et al. 2021, 2023], we present a test-time optimization approach on a collection of test shapes $\mathcal{S}_{\text{test}}$, so that 1) the generator offers improved reconstructions of these shapes, and 2) the tangent spaces at these shapes contain displacement fields that exhibit piece-wise affine structures suitable for shape analysis. Our key idea to achieve 2) is to introduce a weight in front of $r(\boldsymbol{z}, \theta)$ to distribute the distortions of piece-wise affine deformations. Specifically, for the shape S of interest, we set $w(z_S) = 1$ and use a small weight $w(\boldsymbol{z})$ when the distance between $\boldsymbol{z}$ and z_S becomes large. Figure 8 shows the effects of using this weighting scheme. When the deformation between two shapes has non piece-wise affine components, we can enforce that the deformation between the source shape and its adjacent interpolation is piece-wise affine by using a large weight.

Now, let us describe the test-time optimization loss. Let z_S^0 be the latent code of $S \in \mathcal{S}_{\text{test}}$ under the pre-trained generator. Test-time

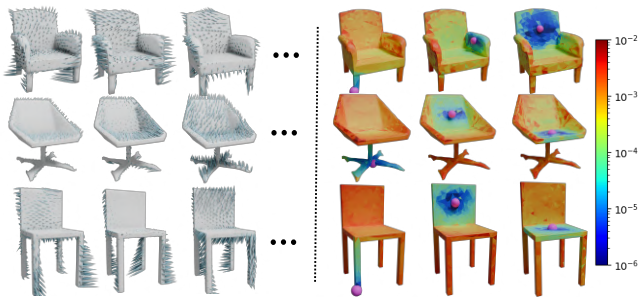

Fig. 9. (Left) Leading vector fields of example shapes. (Right) Visualizations of distance functions of samples colored in purple, The distance function shows the vector field is piece-wise part awareness. Samples from the same shape part has have smaller distance under affine fitting than samples from different parts.

optimization amounts to solve the following optimization problem:

$$\min_{\theta} \frac{1}{|\mathcal{S}_{\text{test}}|} \sum_{S \in \mathcal{S}_{\text{test}}} \Big(l_{\text{data}}(S, g^{\theta}(\cdot, z_S^0)) + \mu \int_{z \in \mathcal{B}^q(z_S^0, c_1)} \exp(-\frac{\|z - z_S^0\|}{2c_2^2}) r(\theta, z) dz \Big) \tag{9}$$

where $\mathcal{B}^q(z_S^0, c_1)$ is the ball centered at z_S^0 with radius r_1. We set $c_2 = \frac{1}{3}c_1$ and c_1 is the median of the shortest distance between the latent code of each test shape and the training shapes. In addition, $\mu = 10^{-1}$.

4.2 Shape Variation Analysis

The second stage of GenAnalysis analyzes the variations of each shape derived from the learned shape generator in the first stage. For each shape S, let $\mathcal{P}_S = \{\boldsymbol{p}_i\}$ be n samples on the surface of the corresponding reconstruction $g^{\theta}(\cdot, z_S)$. We summarize the analysis result in a distance matrix $D_{\mathcal{P}_S} \in \mathbb{R}^{n \times n}$ among all sample points where a small distance indicates that the two corresponding samples belong to the same underlying part.

We compute $D_{\mathcal{P}_S}$ by analyzing piece-wise affine structures in vector-fields $\boldsymbol{u}_l = M^{\theta}(z_S)\boldsymbol{v}_l, 1 \le l \le L$ of $\mathcal{P}_S$ ($L = 20$ in our experiments), where $M^{\theta}(z_S)$ is introduced in Eq.(5). Motivated by the modal analysis framework described in ([Huang et al. 2009]), we compute $\boldsymbol{v}_l, 1 \le l \le L$ as the smallest eigenvectors of

$$H^{\theta}(z_S) = M^{\theta}(z_S)^T L^{\theta}(z_S) M^{\theta}(z_S).$$

The behavior of $\boldsymbol{u}_l$ is similar to the spectral embedding characterized by leading eigenvectors of a graph Laplacian in which points of the same cluster stay close to each other in the embedding space. In our context, we observe that $\boldsymbol{u}_l$ are vector fields that possess affine deformation structures in parts.

For each sample $\boldsymbol{p}_i$ and each vector field $\boldsymbol{u}_l = (\boldsymbol{u}_{l1}; \cdots; \boldsymbol{u}_{ln})$, we fit an affine transformation $A_{li}, \boldsymbol{b}_{li}$ to $\boldsymbol{u}_{lj}, j \in \mathcal{N}_i$:

$$A_{li}, \boldsymbol{b}_{li} = \underset{A,\boldsymbol{b}}{\text{argmin}} \sum_{j \in \mathcal{N}_i} \|A\boldsymbol{p}_j + \boldsymbol{b} - \boldsymbol{u}_{lj}\|^2. \tag{10}$$

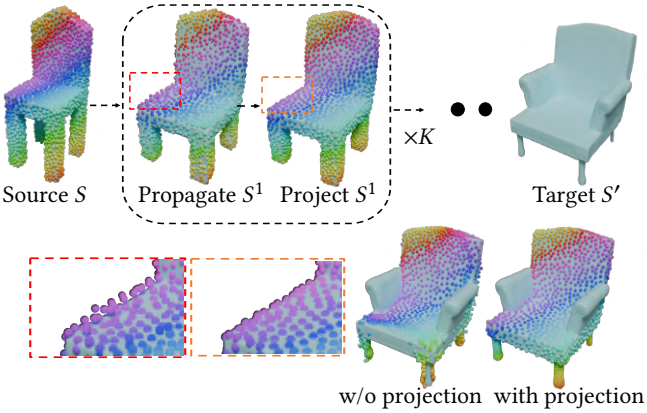

Fig. 10. Correspondences computation of samples from the source shape through intermediate shapes to the target shape. Corresponding points share the same colors. We alternate between a propagation step and a projection step. Without projection, the correspondences do not lie on the target shape and also drift away.

For any sample $\boldsymbol{p}_j$, the residual $\epsilon_{lij} = \|A_{li}\boldsymbol{p}_j + \boldsymbol{b}_{li} - \boldsymbol{u}_{lj}\|$ reveals whether $\boldsymbol{p}_j$ and $\boldsymbol{p}_i$ belong to the same underlying part or not. When ϵ_{lij} is small, $\boldsymbol{p}_j$ and $\boldsymbol{p}_i$ are probably in the same part. In contrast, they are likely in different parts when ϵ_{lij} is large. In light of this discussion, we define

$$D_{\mathcal{P}_S}(i, j) = \Big(\sum_{l=1}^{L} w_l \epsilon_{lij}^2 \Big)^{\frac{1}{2}}.$$

where $w_l = \frac{\lambda_1(H^{\theta}(z_S))}{\lambda_l(H^{\theta}(z_S))}$, and λ_l is the eigenvalue that corresponds to $\boldsymbol{u}_l$. In other words, vector fields with small deformation energies have larger weights. Figure 9 (Left) shows vector fields of an example shape. Figure 9 (Right) visualizes distances of $D_{\mathcal{P}}(i, j)$ when fixing i while varying j. We can see that the resulting distance field is indeed part-aware.

4.3 Shape Matching

The third stage of GenAnalysis computes inter-shape correspondences between two shapes with latent codes z_S and $z_{S'}$ by mapping samples $\mathcal{P}_S$ on $S := \{\boldsymbol{x} \mid g^{\theta}(\boldsymbol{x}, z_S) = 0\}$ to the surface S'. A naive approach is to apply Eq. (5) with the latent direction $\boldsymbol{v} = z_S - z_{S'}$. However, since Eq. (5) relies on a linear approximation of the implicit surface constraint, this approach does not result in points lying exactly on the target implicit surface. To address this, we use the generator $g^{\theta}(\cdot, \boldsymbol{z})$ to obtain $K = 5$ intermediate shapes $S^k := \{\boldsymbol{x} \mid g^{\theta}(\boldsymbol{x}, \boldsymbol{z}^k) = 0\}$, where $\boldsymbol{z}^k = z_S + \frac{k}{K+1}(z_{S'} - z_S)$. We propagate $\mathcal{P}_S$ to S' via $S^k, 1 \le k \le K$. Each propagation step begins by applying Eq. (5) to compute a displacement vector field for the current sample positions.

To further mitigate approximation errors at each propagation step, we introduce a projection step to snap the samples onto the surface. Rather than projecting each point in isolation, we solve an optimization problem to ensure consistent projection across all points. This projection step is similar to minimizing the quadratic energy in Eq.(3) subject to the linear constraints in Eq.(1). The differences are 1) the quadratic energy is defined using propagated

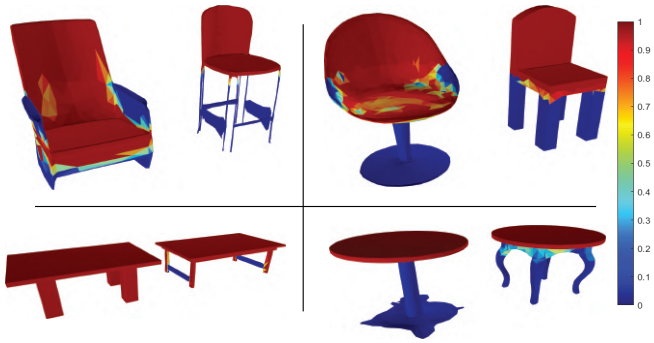

Fig. 11. Color-map of similarity weights $w(\boldsymbol{p}_i)$ between the source shape (left) and the target shape (right) defined by the local distortions e_i of the correspondences computed using our approach. We can see that the similarity weights characterize the structural similarities.

points, and 2) the linear constraints are defined using the next intermediate shape to be projected to.

In our implementation, we alternate between one step of propagation and one step of projection. The output of this procedure gives for each point $\boldsymbol{p} \in \mathcal{P}_S$ its corresponding point on S'. When S and S' are structurally similar, such correspondences are meaningful. When S and S' are only partially similar, some of these correspondences are not well defined. We identify such correspondences by calculating the distortion of the neighborhood of each point. In contrast to computing the distortion between S and S' directly, we find that it is more stable to accumulate the distortion during the propagation procedure. For each point $\boldsymbol{p}_i$, consider its neighborhood $\boldsymbol{p}_j^k$ and neighborhood $\boldsymbol{p}_j^{k+1}$ at step k and step $k+1$ (after one step of propagation and projection). We first calculate the underlying affine transformation A_{ik} between them by adopting Eq. (2):

$$A_{ik} = \operatorname*{argmin}_{A} \sum_{j \in \mathcal{N}_i} \|A(\boldsymbol{p}_j^k - \boldsymbol{p}_i^k) - (\boldsymbol{p}_j^{k+1} - \boldsymbol{p}_i^{k+1})\|^2 + (\mu_r s^2 + \mu_c \|\boldsymbol{a}\|^2).$$

We then define the distortion as

$$e_{ik} := \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} \|A(\boldsymbol{p}_j^k - \boldsymbol{p}_i^k) - (\boldsymbol{p}_j^{k+1} - \boldsymbol{p}_i^{k+1})\|.$$

The similarity weight of the i-th sample point is then given by

$$w(\boldsymbol{p}_i) = \exp\big(-e_i^2/2\sigma^2\big), \qquad e_i = \sum_{k=1}^{K} e_{ik} \tag{11}$$

where σ is the median of e_i. Intuitively, a point has high weight if its neighboring patch has small distortions during the propagation procedure. Figure 11 illustrates the similarity weights of the source shapes with respect to the target shapes. We can see that the distortions reveal structural similarities and differences.

4.4 Consistent Segmentation

The last stage of GenAnalysis performs consistent segmentation among a collection of test shapes $\mathcal{S}_{\text{test}}$ by integrating single-shape segmentation cues formulated in stage II using correspondences obtained in stage III. For computational efficiency concern, we compute $m = 60$ over-segments for each shape (a widely used strategy in image/shape segmentation) and perform consistent segmentation on these over-segments. Figure 12 illustrates the consistent segmentation procedure.

The over-segments are computed for each shape S in isolation. We feed the distance matrix $D_{\mathcal{P}_S}$ into NormalizedCut [Shi and Malik 2000] to obtain the over-segments $\mathcal{O}_S$ for S. As shown in Figure 13, the resulting over-segments are better than those derived from angles between adjacent faces [Golovinskiy and Funkhouser 2008].

Consistent segmentation is achieved by performing spectral clustering on an affinity matrix $W \in \mathbb{R}^{(|\mathcal{S}_{\text{test}}|m)\times(|\mathcal{S}_{\text{test}}|m)}$, an $|\mathcal{S}_{\text{test}}| \times |\mathcal{S}_{\text{test}}|$ block matrix. Each diagonal block W_{ii} encodes the segmentation cue of $S_i \in \mathcal{S}$:

$$W_{ii}(s, s') = \exp\big(-\frac{D_{\mathcal{O}_i}(s, s')^2}{2\overline{\sigma}^2}\big), \qquad 1 \le s, s' \le m. \tag{12}$$

where the distance $D_{\mathcal{O}_i}(s, s')$ between over-segments s and s' is given by the mean value of $D_{\mathcal{P}_{S_i}}(i, j)$ where $i \in s$ and $j \in s'$. $\overline{\sigma}$ is the median of $D_{\mathcal{O}_i}(s, s')$ when s and s' are adjacent over-segments.

The off-diagonal blocks of W are constructed using a similarity graph whose edges $\mathcal{E}$ connect the nearest neighbors of each testing shape $S \in \mathcal{S}_{\text{test}}$. In our implementation, we connect 10 most similar shapes, in which the similar score between two shapes is defined as the average of the correspondence weights between them defined in Eq. (11). We set $W_{ij} = 0, \forall (i, j) \notin \mathcal{E}$. Each non-empty off diagonal block $W_{ij}, (i, j) \in \mathcal{E}$ encodes the correspondences between over-segments of S_i and over-segments of S_j. Given two over-segments $s_i \in \mathcal{O}_i$ and $s_j \in \mathcal{O}_j$, we define their affinity score as

$$W_{ij}(s_i, s_j) = \lambda \frac{|o(s_i, s_j)|}{\max(|s_i|, |s_j|)} \cdot \operatorname*{mean}_{(\boldsymbol{p}_i, \boldsymbol{p}_j) \in o(s_i, s_j)} \frac{(w(\boldsymbol{p}_i) + w(\boldsymbol{p}_j))}{2} \tag{13}$$

where $o(s_i, s_j)$ collects point pairs from s_i and s_j that are in correspondences; the second term in (13) calculates the mean of similarity scores of the correspondences in $o(s_i, s_j)$; λ is a hyper-parameter that balances single shape segmentation cues and consistency in segmentations. As the quality of our inter-shape correspondences is high, we set $\lambda = 2$ in our experiments.

Given W, we employ a variant of spectral graph cut to compute clusters of over-segments across all input shapes, which lead to consistent segmentation. Please refer to Appendix B for details of this spectral clustering procedure. Appendix C presents our consistent segmentation algorithm in algorithm block.

5 EXPERIMENTAL RESULTS

We demonstrate the effectiveness of GenAnalysis and provide both quantitative and qualitative results. We begin with the experimental setup in Section 5.1. We then evaluate the shape matching and shape segmentation results of GenAnalysis in Section 5.2 and Section 5.3, respectively. Section 5.4 presents an ablation study.

5.1 Experimental Setup

5.1.1 Dataset. We train and evaluate GenAnalysis and baselines on ShapeNet [Chang et al. 2015]. We follow the training and testing split of ShapeNetPart [Yi et al. 2016]. To evaluate shape matching, we conduct various surrogate tasks including part label transfer

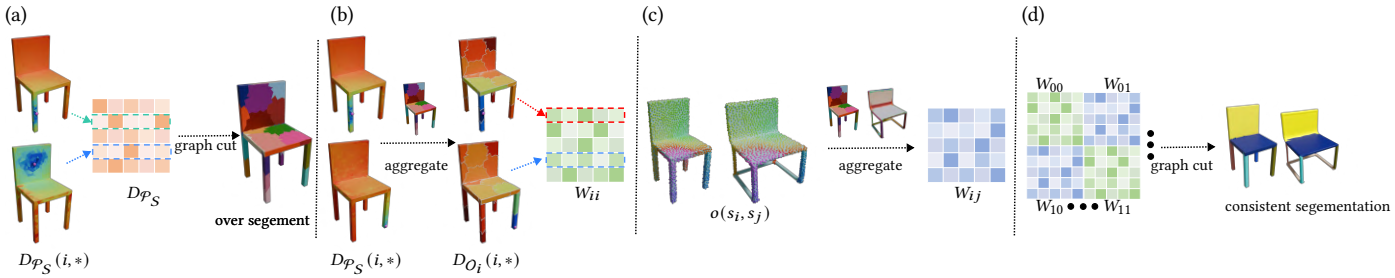

Fig. 12. **Overview of our consistent-segmentation algorithm.** (a) We compute over-segments for each shape S_i using its distance matrix $D_{\mathcal{P}_{S_i}}$. We aggregate (b) affine fitting distance function in each shape and (c) correspondence between each shape pair by over segments.(d) We perform spectral clustering among over-segments of all shapes. Spectral clustering employs an block-wise affinity matrix that encodes each single-shape segmentation cues in its diagonal blocks and correspondences between structurally similar shapes in its off-diagonal blocks.

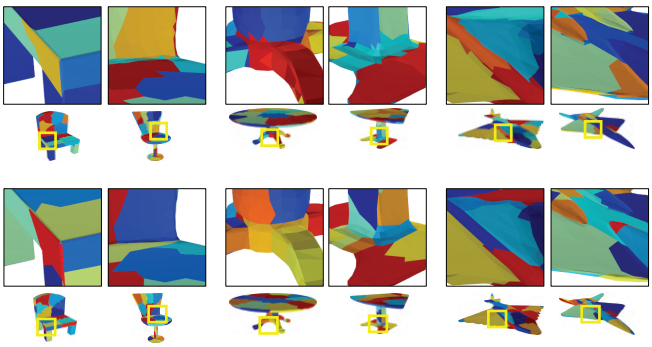

Fig. 13. Comparison between over-segments derived from different strategies. (Top) Over-segments derived from angles between adjacent faces [Golovinskiy and Funkhouser 2008]. (Bottom) Over-segments using our approach, which offer more meaningful segment boundaries.

and keypoint transfer due to the lack of labeled datasets providing ground truth dense correspondence for direct evaluation. We use labels from ShapeNetPart [Yi et al. 2016] and KeypointNet [You et al. 2020] for the part label transfer and key point transfer tasks, respectively. We evaluate shape matching on three popular categories, i.e., chair, table, and airplane. For shape co-segmentation, we also use labels from ShapeNetPart [Yi et al. 2016] for evaluation. We report the performance throughout the entire dataset that has 15 categories. The ablation study is performed in the categories of Chair, Table, and Airplane, as do most baseline approaches.

5.1.2 Baseline approaches. For shape matching, we compare our result with the template learning methods, including DIT [Zheng et al. 2021], DIF [Deng et al. 2021] and Semantic DIF [Kim et al. 2023] since they achieve the state-of-the-art performance in finding dense correspondences on ShapeNet. We also include AtlasNetV2 [Deprelle et al. 2019] and SIF [Genova et al. 2019b] as additional baselines.

We compare our shape co-segmentation results with BAE-Net [Chen et al. 2019], RIM-Net [Niu et al. 2022a] and DAE-Net [Chen et al. 2024], which perform unsupervised shape co-segmentation using branched auto-encoding implicit network. There are some other methods on structure learning that could also perform unsupervised shape co-segmentation. However, they split each input shape into an excessive set of small parts [Chen et al. 2020; Deng et al. 2020], so they are not compared in our evaluation.

mIOU	Chair	Table	Airplane
AtlasNetV2 [Deprelle et al. 2019]	67.1	59.6	56.8
SIF [Genova et al. 2019b]	61.5	62.7	54.3
DIT [Zheng et al. 2021]	79.6	68.7	64.4
DIF [Deng et al. 2021]	80.4	68.6	71.9
SemanticDIF [Kim et al. 2023]	80.6	69.5	71.8
Ours-NR	80.3	70.0	70.7
Ours	**82.6**	**73.0**	**73.3**

Table 1. **Label transfer results on three categories of ShapeNetPart.** We report mean IOU. Higher is better. Ours-NR means GenAnalysis without AAAP regularization.

5.1.3 Implementation Detail. We train our model on a machine with 8 Nvidia Quadro RTX 6000 GPU, and Intel(R) Gold 6230 CPU (40-Core) CPU. We use adam optimizer [Kingma and Ba 2015] and train our model for 4000 epoch using base learning rate of 1e-4. The training takes approximately 3 days to complete. We use the same 8 -layer MLP network as SALD[Atzmon and Lipman 2021] and set λ_d to 1e-4 and λ_{KL} to 1e-5 respectively. Test-time optimization takes 30 minutes on 700 test shapes. Stage II, stage III, and stage IV of GenAnalysis take 1 hour, 4 hours, and 10 minutes, respectively.

5.2 Analysis of Shape Matching

5.2.1 Part label transfer. We transfer the part segment labels from the source shape to the target shape using our correspondence results. Specifically, for each of the three categories, we select five labeled shapes as the source shape, transfer their labels to target shapes and compare with ground truth part labels to record the best results. This task can be viewed as 5-shot 3D shape segmentation using 5 samples as training data. Table 1 presents quantitative results on the 5-shot part label transfer task, measured by average per-part IOU. Our method achieves the best performance in all three categories compared to all the baseline methods. Figure 15 illustrates the qualitative results of part label transfer. AtlasNetV2 and SIF often generate inconsistent correspondences among small regions such as the engine region of airplane and arm region of chair due to the limitation of their shape representations. Template-based learning methods [Deng et al. 2021; Kim et al. 2023; Zheng et al. 2021] suffer from finding accurate correspondences among

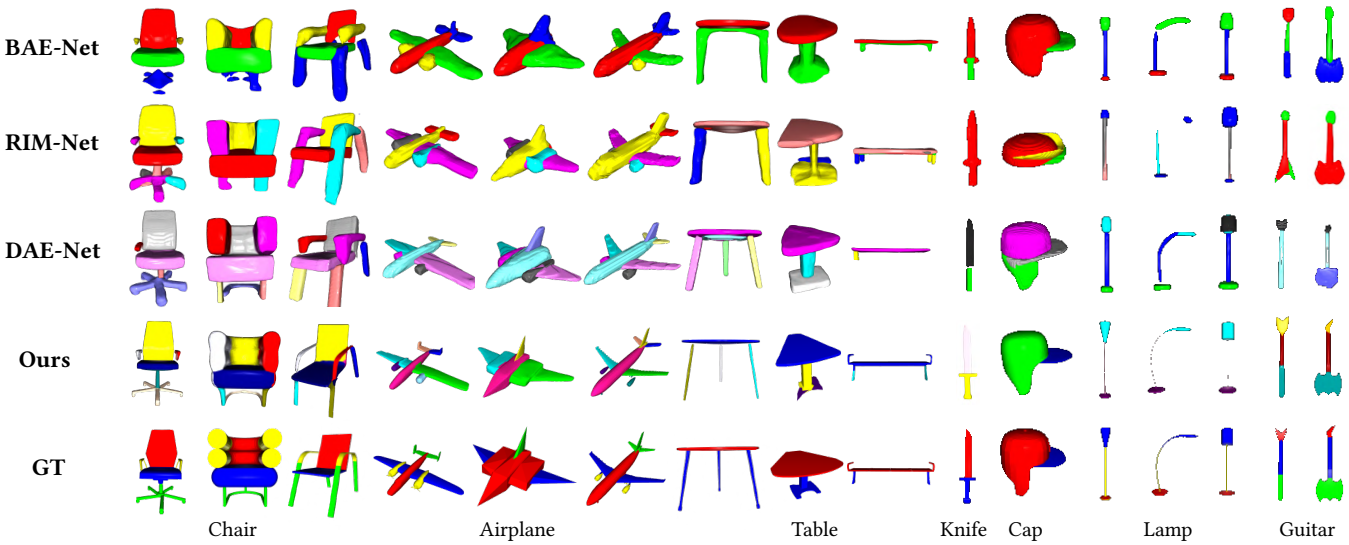

Fig. 14. **Qualitative evaluation of shape co-segmentation quality on ShapeNet.** We compare co-segmentation results with BAE-Net[Chen et al. 2019], RIM-Net[Niu et al. 2022a] and DAE-Net[Chen et al. 2024]. The colored parts visualize segmentation consistency across different shapes in the same category

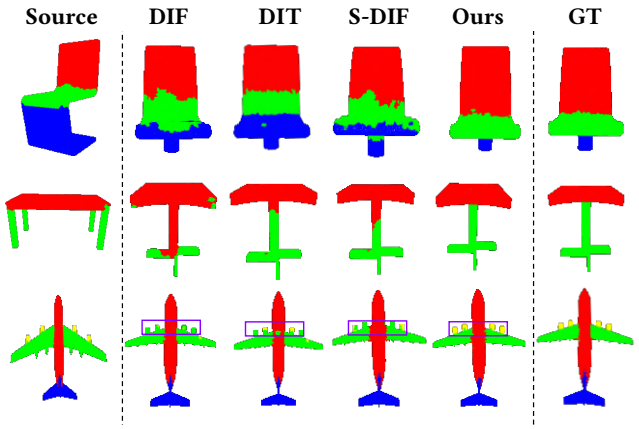

Fig. 15. **Label transfer results on ShapeNet.** We transfer ShapeNet-Part [Yi et al. 2016] labels from source shapes to target shapes and compare our result with DIF [Deng et al. 2021], DIT [Zheng et al. 2021] and Semantic DIF [Kim et al. 2023].

PCK	Chair	Table	Airplane
AtlasNetV2 [Deprelle et al. 2019]	16.6/37.1	24.5/45.3	25.7/42.4
SIF [Genova et al. 2019b]	20.1/40.7	28.6/47.2	28.1/46.7
DIT [Zheng et al. 2021]	24.6/45.3	38.9/54.2	31.7/52.0
DIF [Deng et al. 2021]	32.9/52.5	40.5/61.4	36.9/54.7
SemanticDIF [Kim et al. 2023]	25.9/44.5	26.9/47.9	20.0/31.1
Ours-NR	28.9/49.7	37.3/57.2	35.7/61.4
Ours	**34.9**/**58.6**	43.1/**64.2**	**40.5**/**67.8**

Table 2. **Keypoint transfer results on three categories of ShapeNetPart.** We report the PCK scores with thresholds of 0.01 and 0.02. Higher is better. Ours-NR means GenAnalysis without AAAP regularization.

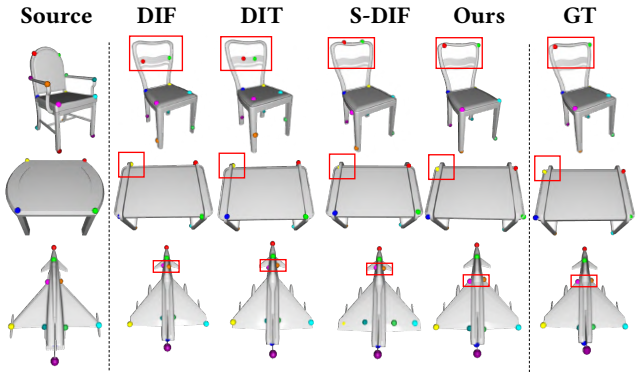

Fig. 16. **Keypoint transfer results on ShapeNet.** We transfer keypoint labels from source shapes to target shapes and compare our result with DIF [Deng et al. 2021], DIT [Zheng et al. 2021] and Semantic DIF [Kim et al. 2023].

structurally distinct and less common shapes due to the limited flexibility of the shared template to represent those shapes. In contrast, GenAnalysis performs much better on those less common shapes with large geometric variations, such as a high-base chair or table. More qualitative visualization results can be found in Figure 24 in the supp. material.

5.2.2 Keypoint transfer. Table 2 presents the quantitative result in the keypoint transfer task. We transfer keypoints from the source shape to the target shape using our correspondence results and measure performance by the percentage of correct keypoints (PCK) [Yi et al. 2017]. We compute the geodesic distance between the transferred keypoints and the ground truth points and report the PCK score under a distance threshold of 0.01/0.02. Table 2 shows that our method outperforms baseline approaches. Note that the relevant

improvements in keypoint transfer are higher than those in segmentation transfer. This can be understood as the fact that GenAnalysis optimizes point-wise correspondences, while segmentation transfer accuracy does not measure the quality of the correspondences in each segment. Figure 16 shows the qualitative keypoint transfer results. Our method shows better results in finding the correspondences between structurally distinct shapes, while template-based learning methods [Deng et al. 2021; Kim et al. 2023; Zheng et al. 2021] fail to find a consistent scale to fit the template and produce inaccurate correspondence results. More qualitative visualization results can also be found in Figure 25 in the supp. material.

5.2.3 Texture transfer. We can also transfer mesh attributes such as texture from the source shape to the target shape using the dense correspondence generated by GenAnalysis. Figure 17 shows texture transfer result between shapes in ShapeNet. Our method preserves complex patterns in texture and transfers them to semantically consistent regions.

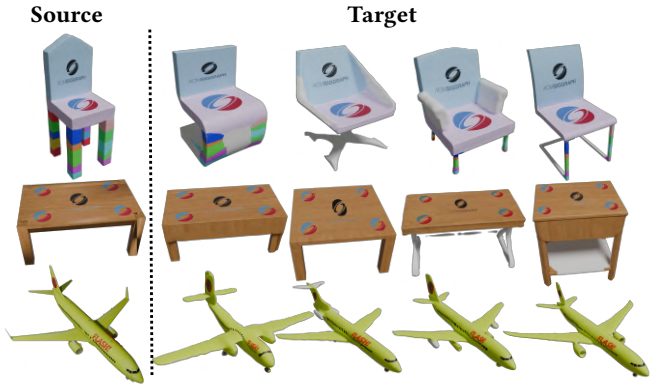

Fig. 17. **Texture transfer result on ShapeNet.** We transfer texture from source shapes to target shapes with different structure on ShapeNet objects using coorrespondences generated by GenAnalysis.

5.3 Analysis of Shape Segmentation

We then evaluate the performance of consistent shape segmentation. The same as DAE-Net [Chen et al. 2024], we quantitatively evaluate the consistent segmentation results by mean IOU in 15 categories of ShapeNetPart.

Table 3 shows that GenAnalysis achieves the highest accuracy in almost all categories except Bag, in which RIM-Net outperforms GenAnalysis by 0.1%. In particular, GenAnalysis achieves salient improvements in Chair, Table, Guitar, Knife, Lamp, Laptop, Rocket, and Skateboard, in which part variations exhibit strong piece-wise affine structures. In Laptop, which exhibits perfect piece-wise affine part variations, GenAnalysis reduces the gap to the ground-truth from DAE-Net by 40% (from 95.0% to 97.1%). For other categories including Airplane, Bag, Cap, Earphone, Motorcycle, Mug, and Pistol, in which part variations are more complex, the improvements of GenAnalysis are still noticeable. These results show the robustness of GenAnalysis in different cases.

We also present qualitative consistent segmentation results in Figure 14. Many baseline methods, especially BAE-Net fail to produce fine-grained segmentations, e.g., four legs of a chair and two back wings of an airplane. These are not reflected in the mean IOU numbers, as these individual segments are grouped into one category in the ground-truth labels. We also notice that RIM-Net struggles to segment shapes with rare and distinct structures, such as the three-leg table, due to the limited flexibility of the binary tree network. DAE-Net presents inconsistent segmentation results between structurally less similar shapes. In contrast, GenAnalysis produces more consistent and fine-grained segmentation than other baseline methods. Furthermore, our method shows much better reconstruction quality compared to all the baseline methods. In this way, we do not need to perform additional post-processing steps used by other baselines such as projecting labels to ground truth shapes for evaluation. This is because instead of finding a consistent partition of the shape in the template using a bottleneck network, as other baseline methods do, we analyze the piecewise affine variation of the shape to produce consistent segmentation, which does not hurt network reconstruction ability. We show that the piecewise affine transformation assumption fits part variation among a wide range of shape category, which supports the powerful generalization ability of our method to more complex shape collections.

5.4 Ablation Study

This section presents an ablation study of GenAnalysis. Table 4 presents quantitative results.

5.4.1 Without regularization. As show in Table 2 and Table 1, the correspondence quality of GenAnalysis in terms of mIOU in the part label transfer task and PCK scores in the keypoint transfer task decreases without the AAAP reguarization loss. This is expected because, without the AAAP regularization, the interpolations from the generator do not recover the underlying deformations between shape pairs. The resulting correspondences easily drift away.

In terms of consistent segmentation, Table 3 shows that performance drops in mIOU values are much more significant when the AAAP regularization term is withdrawn, that is, by 30.3%, 33.4% and 26.7% on the Chair, Table, and Airplane. In addition to having low-quality shape correspondences in this setting, another important factor is that the shape variation segmentation cues become ineffective in this setting.

5.4.2 Without structural variation cue. Next, we drop the structural variation cue on each input shape and use the edge angle formulation in [Golovinskiy and Funkhouser 2008] to generate the over-segments and single shape segmentation cues. In this case, the mIOU values on the Chair, Table, and Airplane drop by 16.1%, 13.4%, and 16.7%, respectively. On the one hand, this shows the effectiveness of the shape variation segmentation cues. However, the decrements are not glaring. This can be understood as the power of consistent shape segmentation using inter-shape correspondences.

5.4.3 Without test-time optimization. Without the test-time optimization step, the mIOU values drop by a few percentage points. We find that the main issue comes from the fact that the learned shape generator does not offer effective reconstructions of test shapes, although the segmentations on the reconstructed shapes are still

mIOU	Chair	Table	Plane	Bag	Cap	Earph.	Guitar	Knife	Lamp	Laptop	Motor.	Mug	Pistol	Rocket	Skateb.	Mean
BAE-Net	56.1	58.4	74.3	84.4	84.9	44.1	51.0	32.5	74.7	27.1	27.5	94.4	29.0	40.9	63.3	56.2
RIM-Net	80.2	54.9	76.0	**86.1**	62.6	72.9	25.7	29.5	68.7	33.2	28.5	48.6	36.2	39.5	64.9	53.6
DAE-Net	85.5	75.5	78.0	84.4	86.3	77.2	88.4	85.8	73.2	95.0	48.1	94.2	74.6	38.7	68.2	76.9
GenAnalysis	**88.4**	**82.6**	**79.1**	86.0	**87.3**	**78.4**	**92.2**	**89.8**	**77.6**	**97.1**	**49.6**	**95.3**	**74.9**	**52.7**	**70.4**	**80.1**

Table 3. **Shape co-segmentation result on ShapeNetPart.** We report the mean IOU. Higer is better. Baseline approaches include BAE-Net [Chen et al. 2019],RIM-Net [Niu et al. 2022a], and DAE-Net [Chen et al. 2024].

mIOU	Chair	Table	Airplane
GenAnalysis-No-AAAPReg	58.1	49.2	52.4
GenAnalysis-No-TanAnal.	72.3	69.2	62.4
GenAnalysis-No-TestTime.	85.7	79.2	74.3
GenAnalysis-No-Weighting.	87.6	81.5	77.6
GenAnalysis	**88.4**	**82.6**	**79.1**

Table 4. **Shape co-segmentation result on ShapeNetPart.** We report the mean IOU. Higer is better. GenAnalysis-AAAPReg means GenAnalysis without AAAP regularization. GenAnalysis-No-TanAnal means no segmentation cues from tangent space analysis. GenAnalysis-No-Testime stands for dropping test-time optimization. GenAnalysis-No-Weighting means dropping the latent code weighting term in test time optimization.

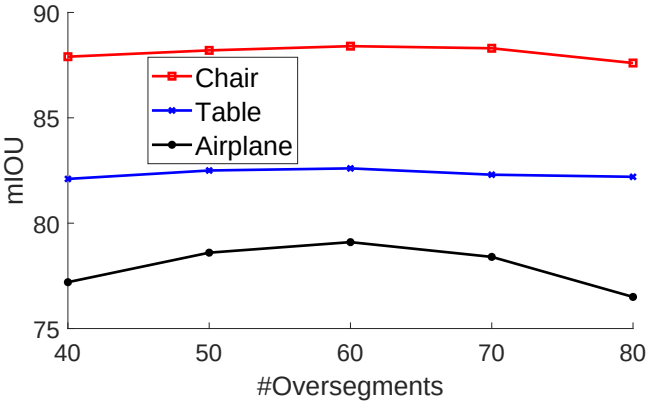

Fig. 18. Mean IOU values when varying the number of over segments. We show three categories, i.e., Chair, Table, and Airplane.

good. In this case, performance drops mainly come from projecting the segmentations back to the test shapes.

5.4.4 Without latent code weighting. The weighting scheme in test-time optimization improves the mIOU values by 0.8% to 1.5% in these three categories. These improvements are consistent, showing its effectiveness.

5.4.5 Varying the number of over-segments m. We proceed to analyze the effects when varying the number of over-segments m per shape. Figure 18 shows the mIOU values of Chair, Table, and Airplane, when varying the number of over-segments m. We can see that the values peak around $m = 60$ and drop slightly when m becomes too small or too large. We can understand this behavior as follows. When m is small, the best shape segmentation offered by the over-segments is of low quality. However, the consistent shape segmentation problem, which employs spectral relaxation, is easier to solve, leading to a good approximate solution. In contrast,

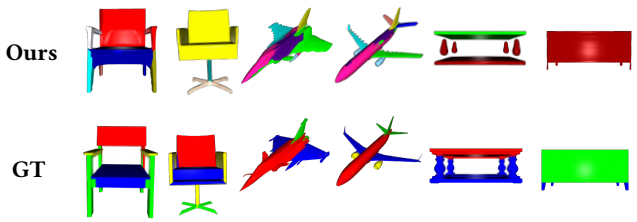

Fig. 19. **Unsuccessful co-segmentation results.**

when m is large, the best shape segmentation offered by the over-segments is of high quality. Yet, it is difficult to obtain the optimal solution, because of the approximation nature of spectral clustering. Nevertheless, the variations in mIOU are not significantly, showing the robustness of GenAnalysis.

6 CONCLUSIONS, LIMITATIONS, AND FUTURE WORK

In this paper, we have introduced GenAnalysis, a novel framework for performing joint shape analysis by learning implicit shape generators. The key idea is to enforce an as-affine-as-possible (AAAP) deformation prior among neighboring synthetic shapes by establishing correspondences between them. This allows us to establish inter-shape correspondences, to extract structurally similar shapes, and to understand shape variations. We show how to extract single-shape segmentations by recovering piece-wise affine structures from the vector fields in the tangent space of each shape. We also show how to perform consistent shape segmentation by integrating segmentation cues from single-shapes using consistent shape correspondences derived from the shape generator. For both shape matching and shape segmentation, GenAnalysis has achieved state-of-the-art results on ShapeNetPart.

The design of GenAnalysis allows efficient test-time optimization that achieves noticeable performance gains. This strength comes from learning an implicit generative model, which can be fine-tuned to fit any test shapes. In contrast, template-based models employed by existing approaches do not possess this property. On the other hand, test-time optimization comes with the cost of optimizing network weights and is computationally more expensive than simply applying learned neural networks during test time.

Since our method relies on spectral clustering to perform consistent segmentation, for some isolated shapes where shape correspondence quality is poor, our method could perform poorly. See the chair example in Figure 19 where the arm of the chair is not distinguished from the back of the chair and the table example. In addition, due to the nature of spectral clustering, we occasionally

observe over-segmentations and under-segmentations, as shown in Airplane and Table in Figure 19.

GenAnalysis opens the door to use shape generators to understand shape variations and establish inter-shape correspondences. There are ample opportunities for future research. On the analysis side, we would like to detect clusters of structurally similar shapes and jointly learn a shape generator of each cluster. This can be achieved by decoupling the latent space into a geometry latent code and a structure latent code. The structure latent code can model structural variations, avoiding explicit clustering. Another direction is to explore shape generation. We propose to introduce another latent code to model shape details and enforce AAAP when varying geometry and structure latent codes.

REFERENCES

Ahmed Abdelreheem, Ivan Skorokhodov, Maks Ovsjanikov, and Peter Wonka. 2023. Satr: Zero-shot semantic segmentation of 3d shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. IEEE Computer Society, Washington, DC, USA, 15166–15179.

Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas J. Guibas. 2018. Learning Representations and Generative Models for 3D Point Clouds. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018 (Proceedings of Machine Learning Research, Vol. 80)*, Jennifer G. Dy and Andreas Krause (Eds.). PMLR, Stockholm, Sweden, 40–49. http://proceedings.mlr.press/v80/achlioptas18a.html

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. 2022. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems* 35 (2022), 23716–23736.

Marc Alexa, Daniel Cohen-Or, and David Levin. 2000. As-rigid-as-possible shape interpolation. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques (SIGGRAPH '00)*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 157–164. https://doi.org/10.1145/344779.344859

Martín Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein Generative Adversarial Networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017 (Proceedings of Machine Learning Research, Vol. 70)*, Doina Precup and Yee Whye Teh (Eds.). PMLR, Sydney, NSW, Australia, 214–223. http://proceedings.mlr.press/v70/arjovsky17a.html

Matan Atzmon and Yaron Lipman. 2021. SALD: Sign Agnostic Learning with Derivatives. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, Online, 17 pages. https://openreview.net/forum?id=7EDgLu9reQD

Matan Atzmon, David Novotný, Andrea Vedaldi, and Yaron Lipman. 2021. Augmenting Implicit Neural Shape Representations with Explicit Deformation Fields.

Melinos Averkiou, Vladimir G. Kim, Youyi Zheng, and Niloy J. Mitra. 2014. ShapeSynth: Parameterizing model collections for coupled shape exploration and synthesis. *Comput. Graph. Forum* 33, 2 (2014), 125–134.

Davide Boscaini, Jonathan Masci, Emanuele Rodoià, and Michael Bronstein. 2016. Learning Shape Correspondence with Anisotropic Convolutional Neural Networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems* (Barcelona, Spain) *(NIPS'16)*. Curran Associates Inc., Red Hook, NY, USA, 3197–3205.

Giorgos Bouritsas, Sergiy Bokhnyak, Stylianos Ploumpis, Michael Bronstein, and Stefanos Zafeiriou. 2019. Neural 3D Morphable Models: Spiral Convolutional Networks for 3D Shape Representation Learning and Generation. In *The IEEE International Conference on Computer Vision (ICCV)*. IEEE Computer Society, Washington, DC, USA, 7213–7222.

Jiazhong Cen, Zanwei Zhou, Jiemin Fang, Wei Shen, Lingxi Xie, Dongsheng Jiang, Xiaopeng Zhang, Qi Tian, et al. 2023. Segment anything in 3d with nerfs. *Advances in Neural Information Processing Systems* 36 (2023), 25971–25990.

Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. 2015. ShapeNet: An Information-Rich 3D Model Repository.

Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. 2022. Contrastive Test-Time Adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*. IEEE, Washington, DC, USA, 295–305. https://doi.org/10.1109/CVPR52688.2022.00039

Zhiqin Chen, Qimin Chen, Hang Zhou, and Hao Zhang. 2024. DAE-Net: Deforming Auto-Encoder for fine-grained shape co-segmentation. In *ACM SIGGRAPH 2024 Conference Papers* (Denver, CO, USA) *(SIGGRAPH '24)*. Association for Computing Machinery, New York, NY, USA, Article 82, 11 pages. https://doi.org/10.1145/3641519.3657528

Zhiqin Chen, Andrea Tagliasacchi, and Hao Zhang. 2020. Bsp-net: Generating compact meshes via binary space partitioning.. In *CVPR*. Computer Vision Foundation / IEEE, Washington, DC, USA, 45–54.

Zhiqin Chen, Kangxue Yin, Matthew Fisher, Siddhartha Chaudhuri, and Hao Zhang. 2019. BAE-Net: Branched autoencoder for shape co-segmentation.. In *CVPR*. Computer Vision Foundation / IEEE, Washington, DC, USA, 8490–8499.

Zhiqin Chen and Hao Zhang. 2019. Learning Implicit Fields for Generative Shape Modeling. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, Long Beach, CA, USA, 5939–5948. https://doi.org/10.1109/CVPR.2019.00609

Dale Decatur, Itai Lang, and Rana Hanocka. 2023. 3d highlighter: Localizing regions on 3d shapes via text descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, Washington, DC, USA, 20930–20939.

Boyang Deng, Kyle Genova, Soroosh Yazdani, Sofien Bouaziz, Geoffrey E. Hinton, and Andrea Tagliasacchi. 2020. CvxNet: Learnable Convex Decomposition. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE, Washington, DC, USA, 31–41. https://doi.org/10.1109/CVPR42600.2020.00011

Yu Deng, Jiaolong Yang, and Xin Tong. 2021. Deformed Implicit Field: Modeling 3D Shapes With Learned Dense Correspondence. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, Washington, DC, USA, 10286–10296. https://doi.org/10.1109/CVPR46437.2021.01015

Theo Deprelle, Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan C. Russell, and Mathieu Aubry. 2019. Learning elementary structures for 3D shape generation and matching. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (Eds.). Curran Associates, New York, NY, USA, 7433–7443. https://proceedings.neurips.cc/paper/2019/hash/d360a502598a4b64b936683b44a5523a-Abstract.html

Marvin Eisenberger, David Novotny, Gael Kerchenbaum, Patrick Labatut, Natalia Neverova, Daniel Cremers, and Andrea Vedaldi. 2021. NeuroMorph: Unsupervised Shape Interpolation and Correspondence in One Go. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, Washington, DC, USA, 7473–7483.

Vignesh Ganapathi-Subramanian, Olga Diamanti, Sören Pirk, Chengcheng Tang, Matthias Nießner, and Leonidas J. Guibas. 2018. Parsing Geometry Using Structure-Aware Shape Templates. In *2018 International Conference on 3D Vision, 3DV 2018, Verona, Italy, September 5-8, 2018*. IEEE Computer Society, Washington, DC, USA, 672–681. https://doi.org/10.1109/3DV.2018.00082

Kyle Genova, Forrester Cole, Daniel Vlasic, Aaron Sarna, William T. Freeman, and Thomas A. Funkhouser. 2019a. Learning Shape Templates With Structured Implicit Functions. In *ICCV*. IEEE, Washington, DC, USA, 7153–7163.

Kyle Genova, Forrester Cole, Daniel Vlasic, Aaron Sarna, William T. Freeman, and Thomas A. Funkhouser. 2019b. Learning Shape Templates With Structured Implicit Functions. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, Washington, DC, USA, 7153–7163. https://doi.org/10.1109/ICCV.2019.00725

Aleksey Golovinskiy and Thomas Funkhouser. 2008. Randomized cuts for 3D mesh analysis. In *ACM SIGGRAPH Asia 2008 papers* (Singapore) *(SIGGRAPH Asia '08)*. ACM, New York, NY, USA, Article 145, 12 pages. https://doi.org/10.1145/1457515.1409098

Aleksey Golovinskiy and Thomas Funkhouser. 2009. Consistent Segmentation of 3D Models. *Computers and Graphics (Shape Modeling International 09)* 33, 3 (June 2009), 262–269.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative Adversarial Networks. *Commun. ACM* 63, 11 (oct 2020), 139–144. https://doi.org/10.1145/3422622

Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. 2020. Implicit Geometric Regularization for Learning Shapes. In *ICML (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, Virtual, 3789–3799.

Emmanuel Hartman, Yashil Sukurdeep, Eric Klassen, Nicolas Charon, and Martin Bauer. 2023. Elastic Shape Analysis of Surfaces with Second-Order Sobolev Metrics: A Comprehensive Numerical Framework. *Int. J. Comput. Vis.* 131, 5 (2023), 1183–1209. https://doi.org/10.1007/S11263-022-01743-0

Ruizhen Hu, Lubin Fan, and Ligang Liu. 2012. Co-Segmentation of 3D Shapes via Subspace Clustering. *Comput. Graph. Forum* 31, 5 (aug 2012), 1703–1713. https://doi.org/10.1111/j.1467-8659.2012.03175.x

Haibin Huang, Evangelos Kalogerakis, and Benjamin Marlin. 2015a. Analysis and Synthesis of 3D Shape Families via Deep-Learned Generative Models of Surfaces. In

Proceedings of the Eurographics Symposium on Geometry Processing (Graz, Austria) *(SGP '15)*. Eurographics Association, Goslar, DEU, 25–38.

Haibin Huang, Evangelos Kalogerakis, and Benjamin Marlin. 2015b. Analysis and Synthesis of 3D Shape Families via Deep-learned Generative Models of Surfaces. *Comput. Graph. Forum* 34, 5 (Aug. 2015), 25–38. https://doi.org/10.1111/cgf.12694

Qixing Huang, Xiangru Huang, Bo Sun, Zaiwei Zhang, Junfeng Jiang, and Chandrajit Bajaj. 2021. ARAPReg: An As-Rigid-As Possible Regularization Loss for Learning Deformable Shape Generators. In *ICCV*. IEEE, Washington, DC, USA, 5795–5805.

Qixing Huang, Vladlen Koltun, and Leonidas Guibas. 2011. Joint Shape Segmentation with Linear Programming. *ACM Trans. Graph.* 30, 6 (dec 2011), 1–12. https://doi.org/10.1145/2070781.2024159

Qixing Huang, Zhenxiao Liang, Haoyun Wang, Simiao Zuo, and Chandrajit Bajaj. 2019. Tensor Maps for Synchronizing Heterogeneous Shape Collections. *ACM Transaction on Graphics* 38 (2019), 106:1–106:18. Issue 4.

Qixing Huang, Fan Wang, and Leonidas Guibas. 2014. Functional Map Networks for Analyzing and Exploring Large Shape Collections. *ACM Trans. Graph.* 33, 4, Article 36 (July 2014), 11 pages. https://doi.org/10.1145/2601097.2601111

Qi-Xing Huang, Martin Wicke, Bart Adams, and Leonidas J. Guibas. 2009. Shape Decomposition using Modal Analysis. *Comput. Graph. Forum* 28, 2 (2009), 407–416.

Evangelos Kalogerakis, Melinos Averkiou, Subhransu Maji, and Siddhartha Chaudhuri. 2017. Deep convolutional priors for indoor scene synthesis. In *CVPR*. Computer Vision Foundation / IEEE, Washington, DC, USA, 3779–3788.

Martin Kilian, Niloy J. Mitra, and Helmut Pottmann. 2007. Geometric modeling in shape space. *ACM Trans. Graph.* 26, 3 (July 2007), 64–es. https://doi.org/10.1145/1276377.1276457

Sihyeon Kim, Juyeon Ko, Minseok Joo, Juhan Cha, Jaewon Lee, and Hyunwoo J. Kim. 2023. Semantic-Aware Implicit Template Learning via Part Deformation Consistency. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*. IEEE, Washington, DC, USA, 593–603. https://doi.org/10.1109/ICCV51070.2023.00061

Vladimir G. Kim, Wilmot Li, Niloy J. Mitra, Siddhartha Chaudhuri, Stephen DiVerdi, and Thomas A. Funkhouser. 2013. Learning part-based templates from large collections of 3D shapes. *ACM Trans. Graph.* 32, 4 (2013), 70:1–70:12.

Vladimir G. Kim, Yaron Lipman, and Thomas Funkhouser. 2011. Blended Intrinsic Maps. *ACM Trans. Graph.* 30, 4, Article 79 (jul 2011), 12 pages. https://doi.org/10.1145/2010324.1964974

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. http://arxiv.org/abs/1412.6980

Diederik P. Kingma and Max Welling. 2014. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). OpenReview.net, Online, 14 pages. http://arxiv.org/abs/1312.6114

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. 2023. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. IEEE Computer Society, Washington, DC, USA, 4015–4026.

Eric Klassen, Anuj Srivastava, Washington Mio, and Shantanu H. Joshi. 2004. Analysis of Planar Shapes Using Geodesic Paths on Shape Spaces. *IEEE Trans. Pattern Anal. Mach. Intell.* 26, 3 (2004), 372–383. https://doi.org/10.1109/TPAMI.2004.1262333

Theodora Kontogianni, Ekin Celikkan, Siyu Tang, and Konrad Schindler. 2023. Interactive object segmentation in 3d point clouds. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, Washington, DC, USA, 2891–2897.

Vladislav Kraevoy, Alla Sheffer, Ariel Shamir, and Daniel Cohen-Or. 2008. Non-homogeneous resizing of complex models. In *ACM SIGGRAPH Asia 2008 papers* (Singapore) *(SIGGRAPH Asia '08)*. ACM, New York, NY, USA, Article 111, 9 pages. https://doi.org/10.1145/1457515.1409064

Itai Lang, Fei Xu, Dale Decatur, Sudarshan Babu, and Rana Hanocka. 2024. iSeg: Interactive 3D Segmentation via Interactive Attention. In *SIGGRAPH Asia 2024 Conference Papers (SA '24)*. Association for Computing Machinery, New York, NY, USA, Article 139, 11 pages. https://doi.org/10.1145/3680528.3687605

Jun Li, Kai Xu, Siddhartha Chaudhuri, Ersin Yumer, Hao (Richard) Zhang, and Leonidas J. Guibas. 2017. GRASS: generative recursive autoencoders for shape structures. *ACM Trans. Graph.* 36, 4 (2017), 52:1–52:14.

Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. 2022. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, Washington, DC, USA, 10965–10975.

Or Litany, Alexander M. Bronstein, Michael M. Bronstein, and Ameesh Makadia. 2018. Deformable Shape Completion With Graph Convolutional Autoencoders. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. IEEE Computer Society, Salt Lake City, UT, USA, 1886–1895. https://doi.org/10.1109/CVPR.2018.00202

Hsueh-Ti Derek Liu, Francis Williams, Alec Jacobson, Sanja Fidler, and Or Litany. 2022. Learning Smooth Neural Functions via Lipschitz Regularization. In *ACM SIGGRAPH 2022 Conference Proceedings* (Vancouver, BC, Canada) *(SIGGRAPH '22)*. Association for Computing Machinery, New York, NY, USA, Article 31, 13 pages. https://doi.org/10.1145/3528233.3530713

Zhang Longwen, Wang Ziyu, Zhang Qixuan, Qiu Qiwei, Pang Anqi, Jiang Haoran, Yang Wei, Xu Lan, and Yu Jingyi. 2024. CLAY: A Controllable Large-scale Generative Model for Creating High-quality 3D Assets. *ACM Trans. Graph.* 43, 6 (2024), 1–14.

William E. Lorensen and Harvey E. Cline. 1987. Marching Cubes: A High Resolution 3D Surface Construction Algorithm. *SIGGRAPH Comput. Graph.* 21, 4 (Aug. 1987), 163–169. https://doi.org/10.1145/37402.37422

Simone Melzi, Jing Ren, Emanuele Rodolà, Abhishek Sharma, Peter Wonka, and Maks Ovsjanikov. 2019. ZoomOut: Spectral Upsampling for Efficient Shape Correspondence. *ACM Trans. Graph.* 38, 6, Article 155 (nov 2019), 14 pages. https://doi.org/10.1145/3355089.3356524

Lars M. Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. 2019. Occupancy Networks: Learning 3D Reconstruction in Function Space. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, Washington, DC, USA, 4460–4470. https://doi.org/10.1109/CVPR.2019.00459

Niloy J. Mitra, Michael Wand, Hao Zhang, Daniel Cohen-Or, Vladimir Kim, and Qi-Xing Huang. 2014. Structure-Aware Shape Processing. In *ACM SIGGRAPH 2014 Courses* (Vancouver, Canada) *(SIGGRAPH'14)*. Association for Computing Machinery, New York, NY, USA, Article 13, 21 pages. https://doi.org/10.1145/2614028.2615401

Kaichun Mo, Paul Guerrero, Li Yi, Hao Su, Peter Wonka, Niloy J. Mitra, and Leonidas J. Guibas. 2019. StructureNet: hierarchical graph networks for 3D shape generation. *ACM Trans. Graph.* 38, 6 (2019), 242:1–242:19.

Sanjeev Muralikrishnan, Siddhartha Chaudhuri, Noam Aigerman, Vladimir G. Kim, Matthew Fisher, and Niloy J. Mitra. 2022. GLASS: Geometric Latent Augmentation for Shape Spaces. In *CVPR*. IEEE, Washington, DC, USA, 470–479.

Chengjie Niu, Manyi Li, Kai Xu, and Hao Zhang. 2022a. RIMNet: Recursive implicit fields for unsupervised learning of hierarchical shape structures.. In *CVPR*. Computer Vision Foundation / IEEE, Washington, DC, USA, 11769–11778.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. 2022b. Efficient Test-Time Model Adaptation without Forgetting. , 16888–16905 pages. https://proceedings.mlr.press/v162/niu22a.html

Maks Ovsjanikov, Mirela Ben-Chen, Justin Solomon, Adrian Butscher, and Leonidas J. Guibas. 2012. Functional maps: a flexible representation of maps between shapes. *ACM Trans. Graph.* 31, 4 (2012), 30:1–30:11. https://doi.org/10.1145/2185520.2185526

Maks Ovsjanikov, Wilmot Li, Leonidas J. Guibas, and Niloy J. Mitra. 2011. Exploration of continuous variability in collections of 3D shapes. *ACM Trans. Graph.* 30, 4 (2011), 33.

Jeong Joon Park, Peter Florence, Julian Straub, Richard A. Newcombe, and Steven Lovegrove. 2019. DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, Washington, DC, USA, 165–174. https://doi.org/10.1109/CVPR.2019.00025

Despoina Paschalidou, Angelos Katharopoulos, Andreas Geiger, and Sanja Fidler. 2021. Neural Parts: Learning Expressive 3D Shape Abstractions With Invertible Neural Networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, Washington, DC, USA, 3204–3215. https://doi.org/10.1109/CVPR46437.2021.00322

Despoina Paschalidou, Ali Osman Ulusoy, and Andreas Geiger. 2019. Superquadrics Revisited: Learning 3D Shape Parsing Beyond Cuboids. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, Washington, DC, USA, 10344–10353. https://doi.org/10.1109/CVPR.2019.01059

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. 2017a. Pointnet: Deep learning on point sets for 3d classification and segmentation.. In *CVPR*. Computer Vision Foundation / IEEE, Washington, DC, USA, 652–660.

Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space.. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2017*. Curran Associates, New York, NY, USA, 5099–5108.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. , 8748–8763 pages.

Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J Black. 2018. Generating 3D faces using convolutional mesh autoencoders. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer Science+Business Media, Berlin/Heidelberg, Germany, 704–720.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. 2024. SAM 2: Segment Anything in Images and Videos. https://arxiv.org/abs/2408.00714

Yusuf Sahillioglu. 2020. Recent advances in shape correspondence. *Vis. Comput.* 36, 8 (2020), 1705–1721. https://doi.org/10.1007/s00371-019-01760-0

Jianbo Shi and Jitendra Malik. 2000. Normalized Cuts and Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 22 (August 2000), 888–905. Issue 8. https://doi.org/10.1109/34.868688

Qingyao Shuai, Chi Zhang, Kaizhi Yang, and Xuejin Chen. 2023. DPF-Net: Combining Explicit Shape Priors in Deformable Primitive Field for Unsupervised Structural Reconstruction of 3D Objects. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*. IEEE, Washington, DC, USA, 14275–14283. https://doi.org/10.1109/ICCV51070.2023.01317

Yawar Siddiqui, Antonio Alliegro, Alexey Artemov, Tatiana Tommasi, Daniele Sirigatti, Vladislav Rosov, Angela Dai, and Matthias Nießner. 2024. MeshGPT: Generating Triangle Meshes with Decoder-Only Transformers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*. IEEE, Washington, DC, USA, 19615–19625. https://doi.org/10.1109/CVPR52733.2024.01855

Oana Sidi, Oliver van Kaick, Yanir Kleiman, Hao Zhang, and Daniel Cohen-Or. 2011. Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering. *ACM Trans. Graph.* 30, 6 (2011), 126.

Miroslava Slavcheva, Maximilian Baust, Daniel Cremers, and Slobodan Ilic. 2017. KillingFusion: Non-rigid 3D Reconstruction without Correspondences. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, Washington, DC, 5474–5483. https://doi.org/10.1109/CVPR.2017.581

Yang Song and Stefano Ermon. 2019. Generative Modeling by Estimating Gradients of the Data Distribution. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (Eds.). Curran Associates Inc., Red Hook, NY, USA, 11895–11907. https://proceedings.neurips.cc/paper/2019/hash/3001ef257407d5a371a96dcd947c7d93-Abstract.html

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2021. Score-Based Generative Modeling through Stochastic Differential Equations.

Olga Sorkine and Marc Alexa. 2007. As-rigid-as-possible surface modeling. In *Proceedings of the fifth Eurographics symposium on Geometry processing* (Barcelona, Spain). Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 109–116. http://dl.acm.org/citation.cfm?id=1281991.1282006

Anuj Srivastava, Eric Klassen, Shantanu H. Joshi, and Ian H. Jermyn. 2011. Shape Analysis of Elastic Curves in Euclidean Spaces. *IEEE Trans. Pattern Anal. Mach. Intell.* 33, 7 (2011), 1415–1428. https://doi.org/10.1109/TPAMI.2010.184

Chun-Yu Sun, Qian-Fang Zou, Xin Tong, and Yang Liu. 2019. Learning adaptive hierarchical cuboid abstractions of 3d shape collections. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 13.

Gary K. Tam, Zhi-Quan Cheng, Yu-Kun Lai, Frank Langbein, Yonghuai Liu, A. David Marshall, Ralph Martin, Xianfang Sun, and Paul Rosin. 2013a. Registration of 3D Point Clouds and Meshes: A Survey from Rigid to Nonrigid. *IEEE Transactions on Visualization and Computer Graphics* 19, 7 (July 2013), 1199–1217. https://doi.org/10.1109/TVCG.2012.310

Gary K. L. Tam, Zhi-Quan Cheng, Yu-Kun Lai, Frank C. Langbein, Yonghuai Liu, A. David Marshall, Ralph R. Martin, Xianfang Sun, and Paul L. Rosin. 2013b. Registration of 3D Point Clouds and Meshes: A Survey from Rigid to Nonrigid. *IEEE Trans. Vis. Comput. Graph.* 19, 7 (2013), 1199–1217. https://doi.org/10.1109/TVCG.2012.310

Michael Tao, Justin Solomon, and Adrian Butscher. 2016. Near-Isometric Level Set Tracking. *Comput. Graph. Forum* 35, 5 (2016), 65–77.

Konstantinos Tertikas, Despoina Paschalidou, Boxiao Pan, Jeong Joon Park, Mikaela Angelina Uy, Ioannis Z. Emiris, Yannis Avrithis, and Leonidas J. Guibas. 2023. Generating Part-Aware Editable 3D Shapes without 3D Supervision. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*. IEEE, Washington, DC, USA, 4466–4478. https://doi.org/10.1109/CVPR52729.2023.00434

Shubham Tulsiani, Hao Su, Leonidas J Guibas, Alexei A Efros, and Jitendra Malik. 2017. Learning shape abstractions by assembling volumetric primitives.. In *CVPR*. Computer Vision Foundation / IEEE, Washington, DC, USA, 2635–2643.

Oliver van Kaick, Hao Zhang, Ghassan Hamarneh, and Daniel Cohen-Or. 2010. A Survey on Shape Correspondence. In *31st Annual Conference of the European Association for Computer Graphics, Eurographics 2010 - State of the Art Reports, Norrköping, Sweden, May 3-7, 2010*, Helwig Hauser and Erik Reinhard (Eds.). Eurographics Association, Eindhoven, Netherlands, 61–82. https://doi.org/10.2312/egst.20101062

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno A. Olshausen, and Trevor Darrell. 2021. Tent: Fully Test-Time Adaptation by Entropy Minimization. https://openreview.net/forum?id=uXl3bZLkr3c

Fan Wang, Qixing Huang, and Leonidas J. Guibas. 2013. Image Co-segmentation via Consistent Functional Maps. In *IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013*. Computer Vision Foundation / IEEE, Washington, DC, USA, 849–856.

Fan Wang, Qixing Huang, Maks Ovsjanikov, and Leonidas J. Guibas. 2014. Unsupervised Multi-class Joint Image Segmentation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*. Computer Vision Foundation / IEEE, Washington, DC, USA, 3142–3149.

Renhao Wang, Yu Sun, Yossi Gandelsman, Xinlei Chen, Alexei A. Efros, and Xiaolong Wang. 2023. Test-Time Training on Video Streams. https://doi.org/10.48550/ARXIV.2307.05014 arXiv:2307.05014

Jiajun Wu, Chengkai Zhang, Tianfan Xue, William T. Freeman, and Joshua B. Tenenbaum. 2016. Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling. In *Proceedings of the 30th International Conference on Neural Information Processing Systems* (Barcelona, Spain) *(NIPS'16)*. Curran Associates Inc., Red Hook, NY, USA, 82–90.

Kai Xu, Vladimir G. Kim, Qixing Huang, and Evangelos Kalogerakis. 2017. Data-Driven Shape Analysis and Processing. *Comput. Graph. Forum* 36, 1 (2017), 101–132. https://doi.org/10.1111/cgf.12790

Kai Xu, Honghua Li, Hao Zhang, Daniel Cohen-Or, Yueshan Xiong, and Zhi-Quan Cheng. 2010. Style-content separation by anisotropic part scales. In *ACM SIGGRAPH Asia 2010 Papers* (Seoul, South Korea) *(SIGGRAPH ASIA '10)*. Association for Computing Machinery, New York, NY, USA, Article 184, 10 pages. https://doi.org/10.1145/1866158.1866206

Haitao Yang, Xiangru Huang, Bo Sun, Chandrajit Bajaj, and Qixing Huang. 2024. GenCorres: Consistent Shape Matching via Coupled Implicit-Explicit Shape Generative Models. In *12th International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024, Conference Track Proceedings*. OpenReview.net, Online, 1–20.

Haitao Yang, Bo Sun, Liyan Chen, Amy Pavel, and Qixing Huang. 2023a. GeoLatent: A Geometric Approach to Latent Space Design for Deformable Shape Generators. *ACM Trans. Graph.* 42, 6, Article 242 (dec 2023), 20 pages. https://doi.org/10.1145/3618371

Kaizhi Yang and Xuejin Chen. 2021. Unsupervised learning for cuboid shape abstraction via joint segmentation from point clouds. *ACM Transactions on Graphics (TOG)* 49, 152 (2021), 11.

Yunhan Yang, Xiaoyang Wu, Tong He, Hengshuang Zhao, and Xihui Liu. 2023b. Sam3d: Segment anything in 3d scenes.

Zhenpei Yang, Lihang Liu, and Qixing Huang. 2018. Learning Generative Neural Networks for 3D Colorization. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*. AAAI, New York, USA, 2580–2587. https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17379

Li Yi, Vladimir G. Kim, Duygu Ceylan, I-Chao Shen, Mengyan Yan, Hao Su, Cewu Lu, Qixing Huang, Alla Sheffer, and Leonidas Guibas. 2016. A Scalable Active Framework for Region Annotation in 3D Shape Collections. *ACM Trans. Graph.* 35, 6, Article 210 (dec 2016), 12 pages. https://doi.org/10.1145/2980179.2980238

Li Yi, Hao Su, Xingwen Guo, and Leonidas J. Guibas. 2017. SyncSpecCNN: Synchronized Spectral CNN for 3D Shape Segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, Washington, DC, USA, 6584–6592. https://doi.org/10.1109/CVPR.2017.697

Yusuke Yoshiyasu, Wan-Chun Ma, Eiichi Yoshida, and Fumio Kanehiro. 2014. As-Conformal-As-Possible Surface Registration. *Comput. Graph. Forum* 33, 5 (2014), 257–267. https://doi.org/10.1111/CGF.12451

Yang You, Yujing Lou, Chengkun Li, Zhoujun Cheng, Liangwei Li, Lizhuang Ma, Cewu Lu, and Weiming Wang. 2020. KeypointNet: A Large-Scale 3D Keypoint Dataset Aggregated From Numerous Human Annotations. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE, Washington, DC, USA, 13644–13653. https://doi.org/10.1109/CVPR42600.2020.01366

Amir Zadeh, Yao-Chong Lim, Paul Pu Liang, and Louis-Philippe Morency. 2019. Variational Auto-Decoder.

Biao Zhang, Jiapeng Tang, Matthias Nießner, and Peter Wonka. 2023. 3DShape2VecSet: A 3D Shape Representation for Neural Fields and Generative Diffusion Models. *ACM Trans. Graph.* 42, 4, Article 92 (July 2023), 16 pages. https://doi.org/10.1145/3592442

Haotian Zhang, Pengchuan Zhang, Xiaowei Hu, Yen-Chun Chen, Liunian Li, Xiyang Dai, Lijuan Wang, Lu Yuan, Jenq-Neng Hwang, and Jianfeng Gao. 2022. Glipv2: Unifying localization and vision-language understanding. *Advances in Neural Information Processing Systems* 35 (2022), 36067–36080.

Yongheng Zhao, Tolga Birdal, Haowen Deng, and Federico Tombari. 2019. 3d point capsule networks. In *CVPR*. Computer Vision Foundation / IEEE, Washington, DC, USA, 1009–1018.

Zerong Zheng, Tao Yu, Qionghai Dai, and Yebin Liu. 2021. Deep Implicit Templates for 3D Shape Representation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, Washington, DC, USA, 1429–1439. https://doi.org/10.1109/CVPR46437.2021.00148

A EXPRESSION OF THE QUADRATIC ENERGY

We can express $e\big(\boldsymbol{g}^\theta(z), \boldsymbol{d}^{\boldsymbol{v}}(z)\big)$

$$:= \min_{\boldsymbol{y}} \begin{pmatrix} \boldsymbol{d}^{\boldsymbol{v}}(z) \\ \boldsymbol{y} \end{pmatrix}^T \begin{pmatrix} K & E^\theta(z) \\ E^\theta(z)^T & G^\theta(z) \end{pmatrix} \begin{pmatrix} \boldsymbol{d}^{\boldsymbol{v}}(z) \\ \boldsymbol{y} \end{pmatrix}. \tag{14}$$

Here $K = \mathrm{diag}(\{|\mathcal{N}_i|\}) \otimes I_3$; $E^\theta(z)$ and $G^\theta(z)$ are $n \times n$ sparse and diagonal block matrices, whose expressions are deferred to Appendix A.1.

As discussed in Appendix A.2, (14) is a quadratic form:

$$e\big(\boldsymbol{g}^\theta(z), \boldsymbol{d}^{\boldsymbol{v}}(z)\big) = \boldsymbol{d}^{\boldsymbol{v}}(z)^T \overline{L}^\theta(z) \boldsymbol{d}^{\boldsymbol{v}}(z). \tag{15}$$

where

$$\overline{L}^\theta(z) := K - E^\theta(z) G^\theta(z)^{-1} E^\theta(z)^T. \tag{16}$$

Moreover, the optimal transformations are given by

$$\boldsymbol{y}^\star = -G^\theta(z)^{-1} E^\theta(z)^T \boldsymbol{d}^{\boldsymbol{v}}(z). \tag{17}$$

A.1 Expressions of Matrices in (14)

We can parameterize the elements of A_i as follows:

$$\mathrm{vec}(A_i) = \overline{J} \cdot \begin{pmatrix} s_i \\ \boldsymbol{c}_i \\ \boldsymbol{a}_i \end{pmatrix} = \overline{J} \boldsymbol{y}_i$$

where

$$\overline{J} := \begin{pmatrix} 1 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{6}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & \frac{2}{\sqrt{6}} & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{6}} & 0 & 0 & 0 \end{pmatrix}.$$

Let

$$R = \begin{pmatrix} \mu_r & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \mu_s I_5 \end{pmatrix}.$$

It follows that

$$e\big(\boldsymbol{g}^\theta(z), \boldsymbol{d}^{\boldsymbol{v}}(z)\big) = \sum_{i=1}^n \Big(\sum_{j \in \mathcal{N}_i} \|((\boldsymbol{p}_i^\theta(z) - \boldsymbol{p}_j^\theta(z)) \otimes)^T \overline{J} \boldsymbol{y}_i - (\boldsymbol{d}_i^{\boldsymbol{v}}(z) - \boldsymbol{d}_j^{\boldsymbol{v}}(z))\|^2 + \boldsymbol{y}_i^T R \boldsymbol{y}_i \Big)$$

It follows that K is the Laplacian matrix of the graph whose edges are $\{(i,j) | j \in \mathcal{N}_i, 1 \le i \le n\}$; the diagonal blocks of G^θ are

$$G_{ii}^\theta(z) = \overline{J}^T \Big(\sum_{j \in \mathcal{N}_i} \big(\boldsymbol{p}_i^\theta(z) - \boldsymbol{p}_j^\theta(z)\big)\big(\boldsymbol{p}_i^\theta(z) - \boldsymbol{p}_j^\theta(z)\big)^T \otimes I_3 \Big) \overline{J} + R;$$

The ij-th block of $E^\theta(z)$ are given by

$$E_{ij}^\theta(z) = \begin{cases} -\sum_{j' \in \mathcal{N}_i} (\boldsymbol{p}_i^\theta(z) - \boldsymbol{p}_{j'}^\theta(z))^T \otimes I_3 & j = i \\ (\boldsymbol{p}_i^\theta(z) - \boldsymbol{p}_j^\theta(z))^T \otimes I_3 & j \in \mathcal{N}_i \setminus \{i\} \\ 0 & \text{otherwise} \end{cases}$$

A.2 Quadratic Norm Expression of Eq.(2)

Given $\boldsymbol{d}^{\boldsymbol{v}}(z)$, the optimal solution to

$$\min_{\boldsymbol{y}} (\boldsymbol{d}^{\boldsymbol{v}}(z)^T, \boldsymbol{y}^T) \begin{pmatrix} K & E^\theta(z) \\ E^\theta(z)^T & G^\theta(z) \end{pmatrix} (\boldsymbol{d}^{\boldsymbol{v}}(z); \boldsymbol{y})$$

is given by

$$\boldsymbol{y} = -G^\theta(z)^{-1} E^\theta(z)^T.$$

Therefore,

$$L^\theta(z) = K - E^\theta(z)^{-1} E^\theta(z)^T \tag{18}$$

A.3 Correspondence Computation

We solve the following quadratic minimization problem with linear constraint to find $\boldsymbol{d}^{\boldsymbol{v}}(z)$:

$$\boldsymbol{d}^{\boldsymbol{v}}(z) := \lim_{\mu \to 0} \operatorname*{argmin}_{\boldsymbol{d}} \quad \boldsymbol{d}^T L^\theta(z) \boldsymbol{d} + \mu \|\boldsymbol{d}\|^2 \quad s.t. \quad C^\theta(z) \boldsymbol{d} = -F^\theta(z) \boldsymbol{v} \tag{19}$$

where $C^\theta(z)\boldsymbol{d} = -F^\theta(z)\boldsymbol{v}$ is the matrix representation of Eq.(1). $\mu\|\boldsymbol{d}\|^2$ avoids degenerate solutions.

It is easy to check the optimal solution to (19) is given by

$$\boldsymbol{d}^{\boldsymbol{v}}(z) = M^\theta(z) \boldsymbol{v}. \tag{20}$$

$$M^\theta(z) = -(I_{3n}, 0) \begin{pmatrix} L^\theta(z) & C^\theta(z)^T \\ C^\theta(z) & 0 \end{pmatrix}^\dagger F^\theta(z) \tag{21}$$

where $A^\dagger$ denotes the pseudo inverse of A. Eq.(21) only needs the pseudo-inverse of a sparse matrix, which can be computed efficiently by LU pre-factorization.

B SPECTRAL CONSISTENT SEGMENTATION

This section presents details of the spectral consistent segmentation procedure used in GenAnalysis. The procedure takes a generalized adjacency matrix $A \in \mathbb{R}^{nm \times nm}$ as input and outputs K partitions $\{1, \cdots, nm\} = \mathcal{I}_1 \cdots \mathcal{I}_M$. The optimal value for M is determined later.

Denote $\boldsymbol{d} = A\mathbf{1}$ and $D = \mathrm{diag}\boldsymbol{d}$. We first consider the normalized adjacency matrix

$$\overline{L} = I_{nm} - D^{-\frac{1}{2}} A D^{-\frac{1}{2}}.$$

With $\boldsymbol{u}_l$ and $\overline{\lambda}_l$ as the l-th leading eigenvector and eigenvalue of $\overline{L}$, where $1 \le l \le L$. In our experiments, we choose $L = 10$.

We form an embedding space

$$U = (\boldsymbol{u}_2, \cdots, \sqrt{\frac{\overline{\lambda}_2}{\overline{\lambda}_L}} \boldsymbol{u}_L) \in \mathbb{R}^{nm \times (L-1)}.$$

where each row of U provides the coordinate of the corresponding over-segment. We then perform probabilistic K-means with an isotropic covariance matrix to compute the resulting clusters. The initial centers are determined by farthest point sampling (FPS).

We determined the optimal value K with the maximum value in $\frac{r_{K-1}}{r_K}$ where r_K is the maximum distance between each point of the cluster center in FPS with K clusters.

ALGORITHM 1: Consistent Segmentation in GenAnalysis

Input : A collection of test shapes $\mathcal{S}_{\text{test}} = \{S_i\}$, each with a distance matrix D_{PS_i}.
Number of over-segments m
Similarity graph $\mathcal{E}$ connecting each test shape to its k most similar shapes (e.g., $k = 10$).
Hyper-parameter λ (e.g., $\lambda = 2$).
Median σ for distance normalization.

Output : Consistent segmentation of the test shapes $\mathcal{S}_{\text{test}}$.

Step 1: Over-segmentation

foreach *test shape* $S_i \in \mathcal{S}_{\text{test}}$ **do**
  Apply NormalizedCut [Shi and Malik 2000] to D_{PS_i} to obtain m over-segments O_{S_i}.

$$O_{S_i} = NormalizedCut(D_{PS_i}).$$

end

Step 2: Construct the Block Affinity Matrix A

Initialize a block matrix

$$\mathbf{W} \in \mathbb{R}^{(|\mathcal{S}_{\text{test}}|\cdot m)\times(|\mathcal{S}_{\text{test}}|\cdot m)}.$$

(a) Diagonal Blocks:

foreach *shape* $S_i \in \mathcal{S}_{\text{test}}$ **do**
  foreach *over-segments* $s,\ s' \in O_{S_i}$ **do**
    Compute

$$\mathbf{W}_i(s, s') \;=\; \exp\Big(-\frac{D_{O_i}(s,s')^2}{2\,\sigma^2}\Big),$$

    where

$$D_{O_i}(s, s') \;=\; \frac{1}{|s|\cdot|s'|}\sum_{i\in s}\sum_{j\in s'} D_{PS_i}(i, j).$$

  end
end

(b) Off-Diagonal Blocks:

foreach $(i, j) \in \mathcal{E}$ *with* $S_i, S_j \in \mathcal{S}_{\text{test}}$ **do**
  foreach *over-segments* $s_i \in O_{S_i}, s_j \in O_{S_j}$ **do**
    Let $o(s_i, s_j)$ be the set of corresponding point pairs. Then

$$\mathbf{W}_{i,j}(s_i, s_j) = \lambda\,\frac{|\,o(s_i, s_j)\,|}{\max(|s_i|, |s_j|)}\;\cdot\;\text{mean}\Big(\frac{w(\mathbf{p}_i)+w(\mathbf{p}_j)}{2}\Big)_{(\mathbf{p}_i,\mathbf{p}_j)\,\in\, o(s_i,s_j)},$$

    where $w(\mathbf{p})$ is the correspondence weight for point $\mathbf{p}$.
  end
end

Set $\mathbf{W}_{i,j} = \mathbf{0}$ for all $(i, j) \notin \mathcal{E}$.

Step 3: Spectral Clustering

Perform spectral graph cut on $\mathbf{A}$ to cluster the over-segments across all shapes in $\mathcal{S}_{\text{test}}$. The resulting clusters yield consistent segment labels for each shape.

return Consistent segmentation of all shapes in $\mathcal{S}_{\text{test}}$.

C CONSISTENT SEGMENTATION ALGORITHM DETAIL

We summarize the algorithm presented in section 4.4 in algorithm block.

D MORE QUALITATIVE RESULTS

We show additional consistent segmentation results on ShapeNet in Figure 20, 21, 22, 23 respectively. We show more results on part label transfer and key point transfer in Figure 24 and 25.

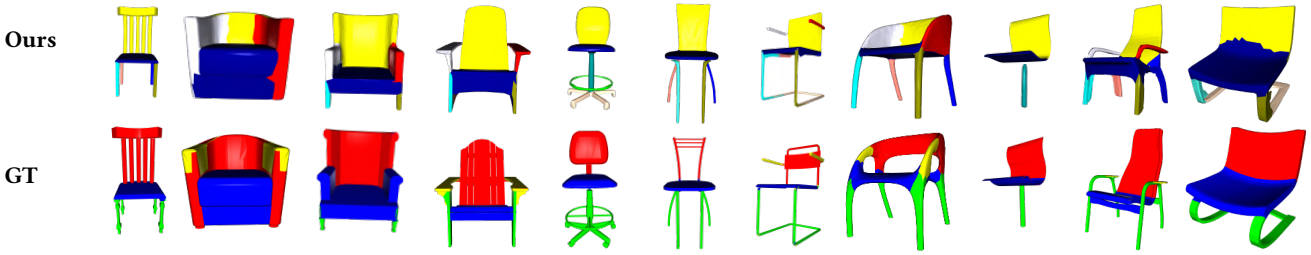

Fig. 20. **More qualitative results of shape co-segmentation on chair category in ShapeNet.**

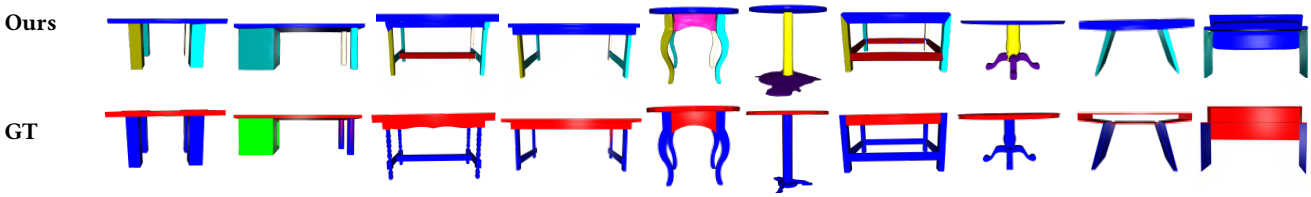

Fig. 21. **More qualitative results of shape co-segmentation on table category in ShapeNet.**

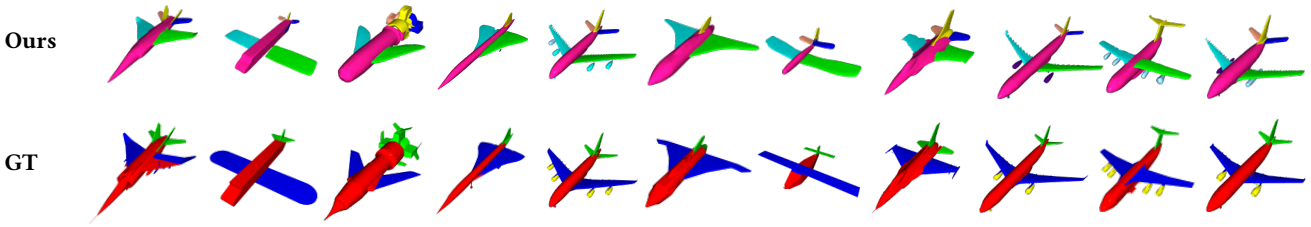

Fig. 22. **More qualitative results of shape co-segmentation on airplane category in ShapeNet.**

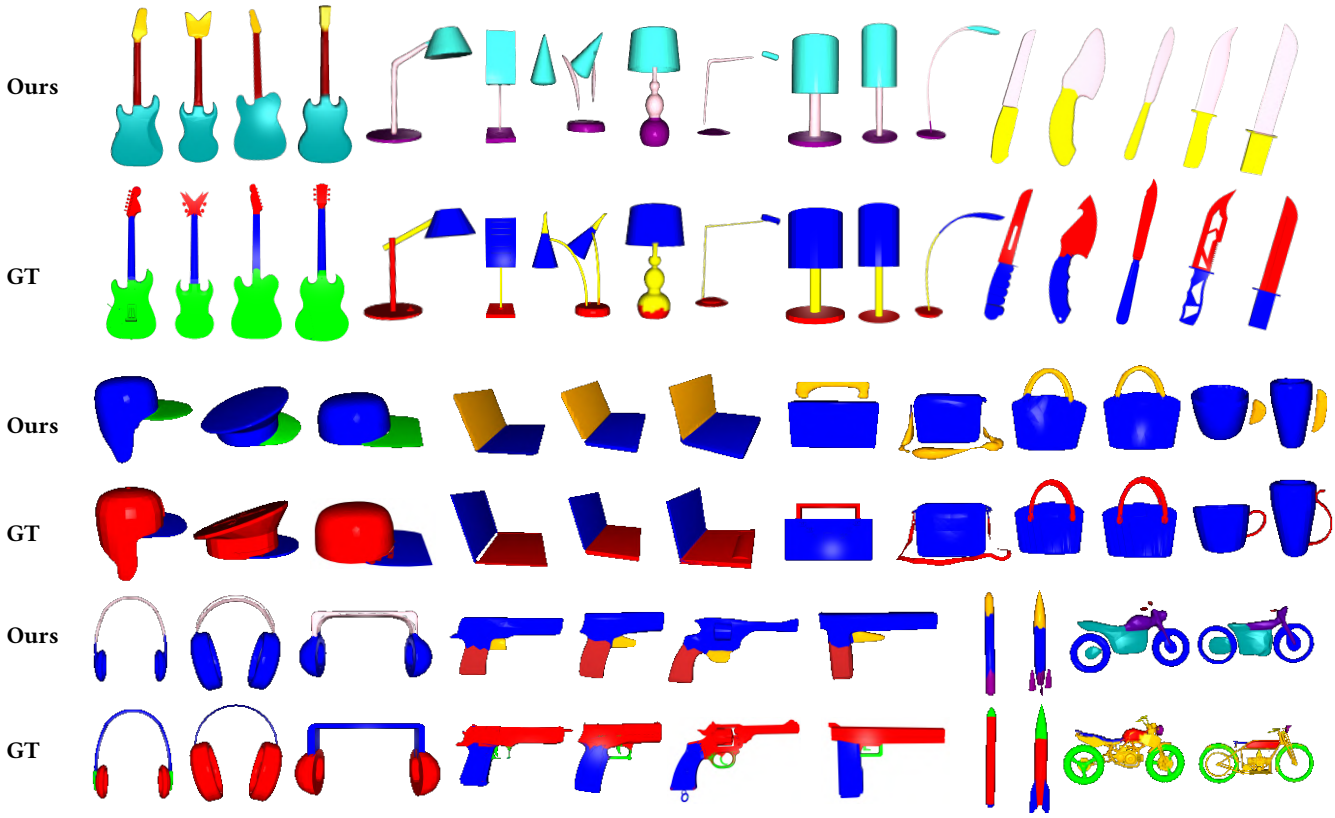

Fig. 23. **More qualitative results of shape co-segmentation on other category in ShapeNet.**

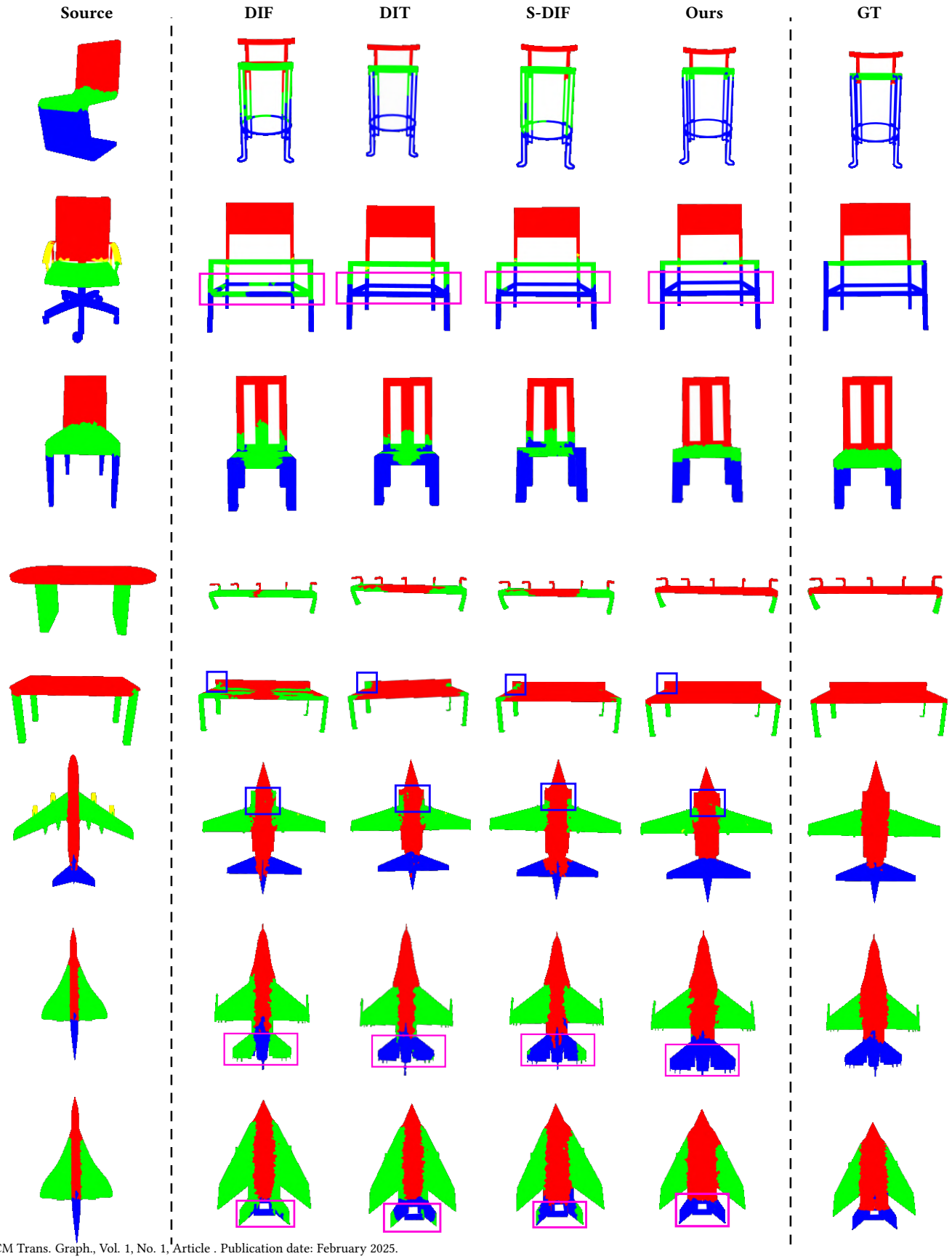

Fig. 24. **Additional comparison on label transfer in ShapeNet.**

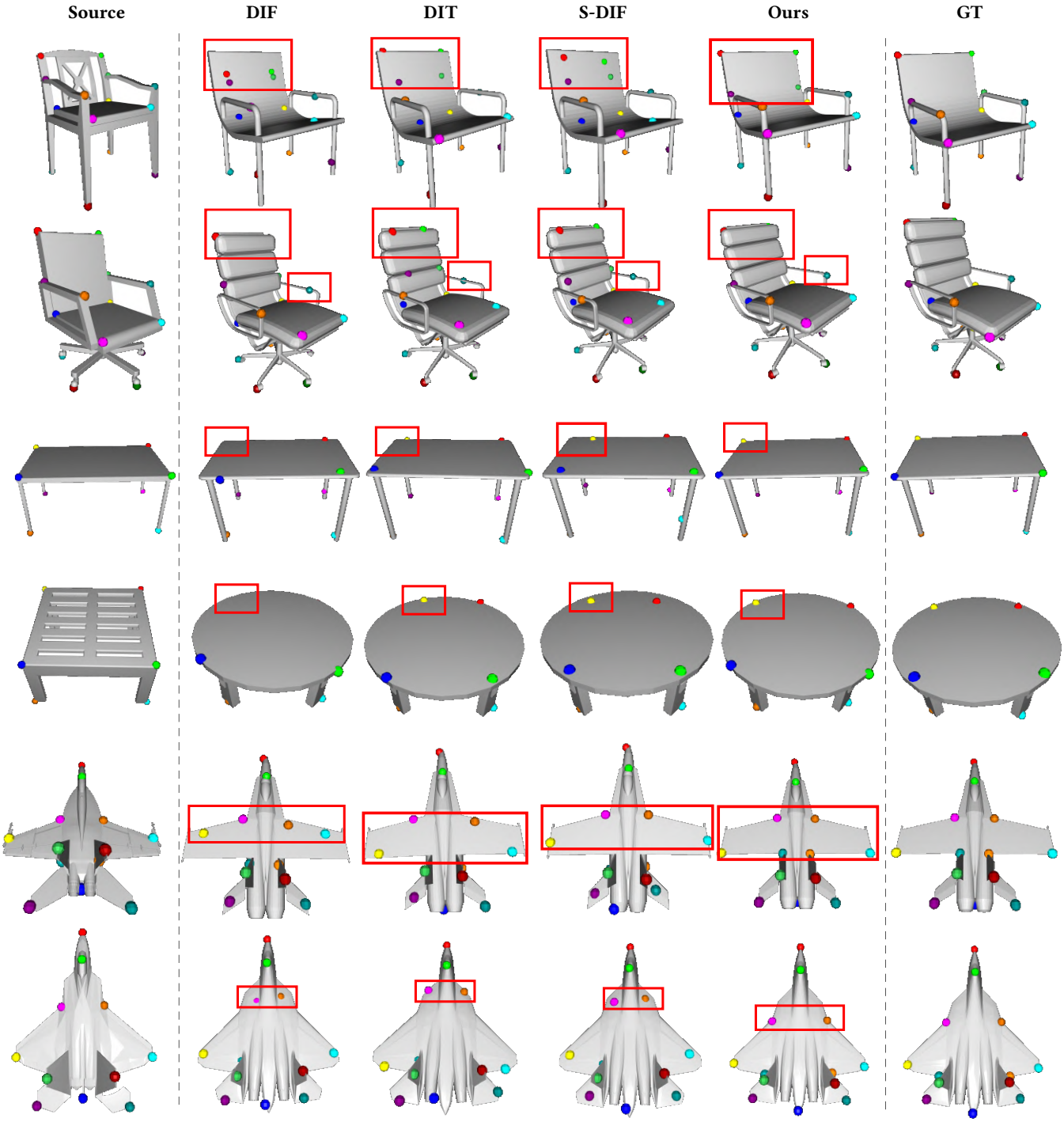

Fig. 25. **Additional comparison on keypoint transfer in ShapeNet.**