Stable and Consistent Prediction of 3D Characteristic Orientation via Invariant Residual Learning

Seungwook Kim [*1] **Chunghyun Park** [*1] **Yoonwoo Jeong** [1] **Jaesik Park** [1 2] **Minsu Cho** [1 2]

Abstract

Learning to predict reliable characteristic orientations of 3D point clouds is an important yet challenging problem, as different point clouds of the same class may have largely varying appearances. In this work, we introduce a novel method to decouple the shape geometry and semantics of the input point cloud to achieve both stability and consistency. The proposed method integrates shape-geometry-based SO(3)-equivariant learning and shape-semantics-based SO(3)-invariant residual learning, where a final characteristic orientation is obtained by calibrating an SO(3)-equivariant orientation hypothesis using an SO(3)-invariant residual rotation. In experiments, the proposed method not only demonstrates superior stability and consistency but also exhibits state-of-the-art performances when applied to point cloud part segmentation, given randomly rotated inputs.

1. Introduction

With recent advances in deep learning, there have been successful attempts to reason about 3D geometric data such as point clouds or meshes (Qi et al., 2017a; Wang et al., 2019; Wu et al., 2019). These attempts have eschewed the need to rely on handcrafted features (Weinmann et al., 2015) to perform various tasks, including 3D object recognition, 3D semantic segmentation, and point cloud registration. However, a consistent inference with respect to variations of input point clouds remains challenging, especially when an arbitrary rotation is involved in the point clouds to handle.

While there exist methods to yield rotation-robust representations by employing rotation-equivariant (Shen et al., 2020; Deng et al., 2021) or rotation-invariant (Sun et al., 2019; Yu et al., 2020; Li et al., 2021b) networks, the yielded representations suffer from loss of expressivity in return for robustness to rotation. A more straightforward and intuitive alternative is to *canonicalize* the point clouds by determining their *characteristic* orientation (Fang et al., 2020; Katzir et al., 2022); the canonicalized point cloud, which is obtained by canceling out the characteristic orientation, preserves its shape information and is readily used for downstream tasks without loss of expressivity. Although it is non-trivial to define the canonical orientation of a point cloud, there exist its desiderata (Katzir et al., 2022): *stability* - the canonical orientation is invariant to any rotation transformations of the input point cloud - and *consistentency* - the canonical orientation is invariant to intra-class variations *i.e.*, point clouds of the same class should have the same canonical pose. Recent methods to canonicalize the point clouds commonly exploit these desiderata to train their network in a self-supervised manner (Sun et al., 2021; Sajnani et al., 2022; Katzir et al., 2022).

Existing methods to predict the characteristic orientation of point clouds, however, suffer from satisfying both of the criteria, stability and consistency, simultaneously. Earlier methods either rely on heavy rotation augmentation (Sun et al., 2021) to model equivariance, or rely on networks that are not purely equivariant (Spezialetti et al., 2020) to predict characteristic orientations, which therefore yield suboptimal stability. More recent methods propose to disentangle the shape and pose of point clouds by yielding rotation-invariant shape representation and rotation-equivariant pose information (Sajnani et al., 2022; Katzir et al., 2022). These methods are tightly coupled with a class-sensitive point cloud reconstruction pipeline (Groueix et al., 2018), consequently making them unsuitable for multi-class training. While they may be stable, such approaches often fail to obtain consistent orientation predictions.

In this work, we introduce an effective method to learn the characteristic orientation of point clouds achieving both stability and consistency. The idea is to decouple the shape information of a point cloud into two factors, pure geometry and semantics, and coordinate them by means of SO(3)-equivariant learning and SO(3)-invariant residual learning. Specifically, we predict a SO(3)-equivariant characteristic

[*]Equal contribution [1]Graduate School of Artificial Intelligence, POSTECH, Pohang, South Korea [2]Department of Computer Science and Engineering, POSTECH, Pohang, South Korea. Correspondence to: Minsu Cho <mscho@postech.ac.kr>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

orientation hypothesis and calibrate it using SO(3)-invariant residual rotation to yield a final prediction for stable and consistent canonicalization. We name the proposed CHaracteristic Orientation predictor with Invariant Residual learning as CHOIR. As it dispenses with the widely used point cloud reconstruction in learning (Sajnani et al., 2022; Katzir et al., 2022), CHOIR facilitates multi-class training, being more practical for downstream point cloud analysis tasks.

The contribution of our work is fourfold. **(1)** We introduce CHOIR, a strong characteristic orientation prediction method that decouples shape information of point clouds to geometry and semantics to achieve both stability and consistency. **(2)** We propose a novel residual predictor that learns to output a SO(3)-invariant residual rotation to calibrate our initial orientation hypotheses for better consistency. **(3)** In experiments, we not only demonstrate state-of-the-art stability and consistency, but also show superior performance when applied to the task of point cloud part segmentation. **(4)** We facilitate multi-class training by removing point cloud reconstruction, enhancing applicablity to downstream tasks, **(5)** We analyze the influence of consistency and stability on downstream tasks, which indicates that the balance between stability and consistency is the key to canonicalizing point clouds for downstream tasks.

2. Related Work

Learning based point cloud analysis. Over the recent years, various methods have been proposed to handle point clouds, which pose difficulties due to their irregularity and lack of order. PointNet (Qi et al., 2017a) and its follow-up methods (Qi et al., 2017b; Wu et al., 2019; Thomas et al., 2019; Wang et al., 2019) propose to input point clouds directly to networks to achieve permutation-invariant representations of the point cloud. For example, DGCNN (Wang et al., 2019) uses graph convolution using pointwise k-nearest-neighbors, while KPConv (Thomas et al., 2019) defines the convolution area using a set of kernel points. More recently, Point Transformer (Zhao et al., 2021) proposed to leverage the transformer architecture to process point clouds, while PointMixer (Choe et al., 2022) facilitates information sharing between points by replacing token-mixing MLPs with Softmax function. While these methods have shown impressive performances on point cloud analysis tasks including classification and segmentation, these methods rely on extensive train-time or test-time rotation augmentations to handle random rotations encountered during inference.

Rotation-robust point cloud analysis. To minimize the need for intensive rotation augmentations, one thread of work aims to design rotation-equivariant modules, such that the output representations will be equivariant with respect to the rotation of the input point cloud (Esteves et al., 2018; Rao et al., 2019; Shen et al., 2020; Fang et al., 2020; Thomas et al., 2018; Chen et al., 2021). To introduce a few recent successes, YOHO (Wang et al., 2022) exploits the symmetry of icosahedral group to extract group-equivariant representations. Luo et al. (2022) propose the idea of equivariant message passing by learning point-wise orientations. Vector Neuron Networks (Deng et al., 2021) lift the latent representations from scalar vectors to matrix vectors to implement a fully equivariant network.

Another thread of work explores achieving strict rotational invariance *i.e.*, yielding the same output representation regardless of the rotation of the input point cloud. Many existing work focus on handcrafting rotation-invariant features based on intrinsic geometries (Sun et al., 2019; Li et al., 2021b; Xiao & Wachs, 2021). While the above methods suffer from information loss, some approaches propose to leverage PCA-based canonical poses to achieve rotational invariance (Kim et al., 2020; Yu et al., 2020; Li et al., 2021a). However, PCA-based canonical poses suffer from sign and order ambiguities (Li et al., 2021a), and considering all the possible candidates for the canonical pose leads to high computational overhead.

In our work, we propose to leverage SO(3)-equivariant networks for learning the shape geometry, and SO(3)-invariant networks for learning the shape semantics, to predict the characteristic orientation of point clouds. Canonicalizing the input point cloud using our predicted characteristic orientation enables us to preserve the shape information without having to consider various ambiguities.

Self-supervised category-level 6D pose estimation. The task of category-level pose estimation aims to find the 6D poses of unseen instances from known categories, without access to object CAD models. A similar approach called shape co-alignment (Averkiou et al., 2016; Mehr et al., 2018) proposes to obtain consistent category-level reference frames, which require clean mesh inputs or rely heavily on the symmetry of point clouds. QEC (Zhao et al., 2020) proposes a novel dynamic routing procedure on quaternions to establish end-to-end transformation equivariance, but QEC does not evaluate the predicted category-level pose, and assumes complete shapes. EquiPose (Li et al., 2021c) proposes to handle both complete and partial shapes, and introduces a self-supervised learning framework to estimate several category-level object pose candidates from single 3D point clouds by leveraging EPNs (Chen et al., 2021).

While characteristic orientation prediction and 6D pose estimation tasks are pertinent, 6D pose estimation, unlike characteristic orientation prediction, has ground-truth category-level pose annotations that uniquely define a category-level pose reference frame.

Estimation of characteristic orientation. Previously, some work proposed to learn the canonical frame of objects using

strong or weak supervision (Rempe et al., 2020; Novotny et al., 2019; Gu et al., 2020). More recent methods propose to determine the characteristic orientation of point clouds in a self-supervised manner. Compass (Spezialetti et al., 2020) projects point clouds to spherical signals to be processed by SO(3) group-equivariant spherical CNNs, and leverages the arg-max of SO(3)-equivariant representations to predict the characteristic orientation of point clouds. In Canonical Capsules (Sun et al., 2021), a network is trained to decompose point clouds into parts, on which invariance/equivariance is enforced through a Siamese training setup. The point clouds are then canonicalized to a learned frame of reference by reconstruction. ConDor (Sajnani et al., 2022) leverages Tensor Field Networks (TFNs) (Thomas et al., 2018), a class of permutation- and rotation equivariant, and translation-invariant 3D networks to learn to output an equivariant canonical pose. VN-SPD (Katzir et al., 2022) learns to disentangle shape and pose information from input point clouds by leveraging SO(3)-equivariant and SO(3)-invariant representations produced from Vector Neuron Networks (Deng et al., 2021). Note that Canonical Capsules, ConDor and VN-SPD rely on point cloud reconstruction pipelines to obtain the canonicalized point clouds, making their model suitable only for a single class of point clouds.

In this work, we eschew the point cloud reconstruction pipeline and rather focus on learning the residual rotation using SO(3)-invariant layers to calibrate the initial characteristic orientation hypothesis obtained from our SO(3)-equivariant networks (that directly takes point clouds as the input), enforcing better consistency.

3. Problem Statement

Given an input point cloud $\mathbf{P} \in \mathbb{R}^{N\times 3}$, which is centered at the origin, we aim to predict its *characteristic* orientation $R' \in \mathrm{SO}(3)$ such that point clouds of the same class with *arbitrary rotation* and *intra-class variation* are all aligned to the same coordinate frame, called the *canonical* orientation, by canceling out the characteristic orientation via rotation $R'^{\top}$. The resultant canonicalized point cloud $\mathbf{P}R'^{\top}$ provides a rotation-invariant representation, which is crucial for robust 3D perception in practical scenarios. Since the main challenge of the problem lies in achieving the robustness to both (1) arbitrary rotation and (2) intra-class variation, the characteristic orientation prediction is commonly evaluated with two metrics: *stability* and *consistency* (Sun et al., 2021; Katzir et al., 2022).

The stability metric quantifies how similar canonical orientations are recovered from point clouds of the same instance with different orientations, *i.e.*, the robustness to arbitrary rotation. Specifically, from a point cloud $\mathbf{P}$, randomly rotated ones $\{\mathbf{P}R_i\}$ are generated and then their characteristic orientations $\{R'_i\}$ are predicted. As canonicalized point clouds correspond to $\{\mathbf{P}R_iR_i'^{\top}\}$, the stability metric is measured by the standard deviation of the net rotations $\{R_iR_i'^{\top}\}$:

$$d_{\text{stability}}(\{R_iR_i'^{\top}\}_{i=1}^{K}) = \sqrt{\frac{1}{K}\sum_{i=1}^{K}\angle(R_iR_i'^{\top}, \bar{R})^2}, \quad (1)$$

where $\bar{R}$ is the chordal L_2-mean of $\{R_iR_i'^{\top}\}_{i=1}^{K}$ and $\angle(\cdot,\cdot)$ denotes the angle difference between two rotation matrices.

On the other hand, the consistency metric quantifies how similar canonical orientations are recovered from different instances of the same category, *i.e.*, the robustness to intra-class variation. Specifically, given N different point cloud instances from the same class, we measure the standard deviation of their predicted orientations as follows:

$$d_{\text{consistency}}(\{R'_j\}_{j=1}^{N}) = \sqrt{\frac{1}{N}\sum_{j=1}^{N}\angle(R'_j, \bar{R})^2}, \quad (2)$$

where $\bar{R}$ denotes the chordal L_2-mean of the predicted characteristic orientations $\{R'_j\}_{j=1}^{N}$.

4. Our Approach

We introduce a characteristic orientation predictor, CHOIR, that effectively minimizes both stability and consistency.

The main idea of our approach is to identify two distinctive factors in determining a characteristic orientation and decouple them in learning and prediction. The first and primary factor is the pure shape geometry: a characteristic orientation may be assigned purely based on its shape geometry such that the rotated shape of the object has a consistent characteristic orientation corresponding to the rotation intact. The second factor is the shape semantics: a characteristic orientation may be affected by recognizing a specific semantic property, *e.g.*, the signature of a specific object instance or class. Note that the shape semantics is of course related to the shape geometry but is distinct in the sense that it involves additional semantic information in learning. As can easily be noticed, these two factors are closely connected to the two criteria of characteristic orientation, stability and consistency, respectively, which may cause conflict with each other in learning. To achieve both stability and consistency, we decouple the two factors, shape geometry and shape semantics, and integrate them in a compatible manner by combining shape-geometry-based SO(3)-equivariant learning and shape-semantics-based SO(3)-invariant residual learning. Figure 1 illustrates the overview of our approach.

To guarantee the stability of predictions, we leverage SO(3)-equivariance in our network to predict a shape-geometry-based characteristic orientation hypothesis, such that it remains *equivariant* to the rotation of the input (Sec. 4.2). To

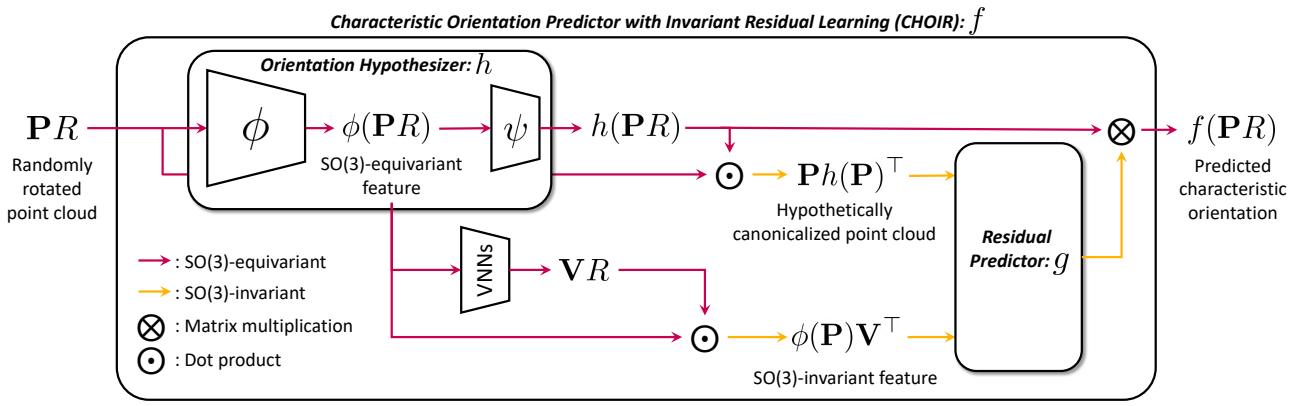

Figure 1. **Overview of CHOIR.** CHOIR (f) predicts the characteristic orientation using an SO(3)-equivariant orientation hypothesizer (h) and an SO(3)-invariant residual predictor (g). The orientation hypothesizer consists of two modules: SO(3)-equivariant encoder (ϕ) and SO(3)-equivariant rotation predictor (ψ). The hypothesizer first predicts an equivariant characteristic orientation hypothesis, which is calibrated using the rotation-invariant residual rotation predicted by the residual predictor. With this novel rotation-invariant residual learning, CHOIR predict the characteristic orientation of a rotated point cloud that is both stable under arbitrary rotations and consistent under intra-class shape deformations.

enforce consistency of predictions, we introduce a SO(3)-invariant residual rotation predictor (Sec. 4.3), which refines the characteristic orientation hypothesis by predicting a shape-semantics-based residual rotation from SO(3)-invariant features. Note that our residual strategy for consistency allows the final prediction to preserve the stability since SO(3)-invariant rotation does not hurt SO(3)-equivariance of the orientation hypothesis. In training CHOIR, the adoption of SO(3)-equivariant network for the characteristic orientation hypothesis intrinsically reflects the shape geometry, enforcing the stability by design, and thus we focus on learning the consistency based on shape semantics. To this end, we use a Siamese training setup on aligned point cloud pairs of different point clouds within the same class, using self-supervised losses to facilitate consistency between the predicted characteristic orientations of the point cloud pair (Sec. 4.4).

4.1. Preliminary: Vector Neuron Networks

In designing CHOIR, one of our aims is to improve the stability of its characteristic orientation predictions. To this end, we employ SO(3)-equivariant layers into our network architectures, inspired by Vector Neuron Networks (VNNs) (Deng et al., 2021) for their simplicity and generalizability. In VNNs, the sequence of scalar values, which represents a single neuron, is lifted to a sequence of 3D vectors, *i.e.*, a *vector-list* feature $\mathbf{V} \in \mathbb{R}^{C\times 3}$. The layers of VNNs map from and to batches of such vector-list features such that $f(\mathbf{V}R) = f(\mathbf{V})R$ satisfying the equivariance to the rotation $R \in \mathrm{SO}(3)$. SO(3)-invariant features can also be obtained using the product of an equivariant vector-list feature $\mathbf{V}R \in \mathbb{R}^{C\times 3}$ and the transpose of another equivariant vector-list feature $\mathbf{U}R \in \mathbb{R}^{C'\times 3}$: $(\mathbf{V}R)(\mathbf{U}R)^\top = \mathbf{V}RR^\top\mathbf{U}^\top = \mathbf{V}\mathbf{U}^\top$. We refer the readers to the original paper (Deng et al., 2021) for more details.

4.2. SO(3)-equivariant Orientation Hypothesizer

The characteristic orientation hypothesizer builds on the top of SO(3)-equivariant VNNs (Deng et al., 2021) so that it initially predicts a SO(3)-equivariant orientation in our approach. Specifically, the characteristic orientation hypothesizer ($h := \psi \circ \phi$) consists of two SO(3)-equivariant modules, an encoder $\phi : \mathbb{R}^{N\times 3} \mapsto \mathbb{R}^{N\times C\times 3}$ and a rotation predictor $\psi : \mathbb{R}^{N\times C\times 3} \mapsto \mathrm{SO}(3)$. The encoder processes a rotated point cloud $\mathbf{P}R \in \mathbb{R}^{N\times 3}$ using VNNs and extracts a SO(3)-equivariant feature $\phi(\mathbf{P}R)$. Then, the SO(3)-equivariant rotation predictor, ψ, predicts an orientation hypothesis by reducing the channel of $\phi(\mathbf{P}R)$ from C to 2 to estimate two basis vectors of the orientation and orthonormalizing the vectors by Gram-Schmidt process. Therefore, our hypothesizer maintains SO(3)-equivariance as:

$$h(\mathbf{P}R) = \psi(\phi(\mathbf{P}R)) = \psi(\phi(\mathbf{P}))R = h(\mathbf{P})R. \quad (3)$$

Attributing to these properties of our hypothesizer, our characteristic orientation hypotheses are stable.

4.3. SO(3)-invariant Residual Predictor

Due to the presence of intra-class shape variations within a class, our hypothesizer network alone is insufficient to obtain consistent class-specific characteristic orientations in spite of our usage of cross-shape point cloud pairs. To this end, we introduce a residual orientation predictor that aims to calibrate the class-specific characteristic orientation hypothesis $h(\mathbf{P}R)$ such that it will be consistent within semantic classes. Formally put, we assume there exists

a true class-specific canonical orientation, and there is a residual between the true class-specific canonical orientation and the coordinate frame aligned by the predicted SE(3)-equivariant characteristic orientation hypothesis due to its SO(3)-invariant properties such as shape or partiality. Therefore, we introduce a SO(3)-invariant residual predictor ($g : \mathbb{R}^{N\times 3} \times \mathbb{R}^{N\times C\times 3} \rightarrow \mathrm{SO}(3)$) to predict the residual and to align point clouds from the same class into the true class-specific canonical orientation.

We employ Point Transformer (Zhao et al., 2021), which takes 3D points and their corresponding features as inputs, as our SO(3)-invariant residual predictor denoted by $g(\cdot,\cdot)$; and because we want the predicted residual to be invariant to rotations, we provide rotation-invariant features and a *hypothetically* canonicalized point cloud as inputs. We first generate the rotation-invariant feature using the intermediate SO(3)-equivariant feature $\phi(\mathbf{P}R) \in \mathbb{R}^{N\times C\times 3}$ from the hypothesizer network h. Specifically, we pass $\phi(\mathbf{P}R)$ through more SO(3)-equivariant layers to yield another SO(3)-equivariant representation $\mathbf{V}R \in \mathbb{R}^{N\times 3\times 3}$. We then compute their inner product along the batch dimension N to obtain rotation-invariant features of the input point cloud:

$$\phi(\mathbf{P}R)(\mathbf{V}R)^\top = \phi(\mathbf{P})RR^\top\mathbf{V}^\top = \phi(\mathbf{P})\mathbf{V}^\top. \quad (4)$$

To obtain the *hypothetically* canonicalized point cloud, we apply our orientation hypothesis onto the input:

$$\mathbf{P}Rh(\mathbf{P}R)^\top = \mathbf{P}RR^\top h(\mathbf{P})^\top = \mathbf{P}h(\mathbf{P})^\top. \quad (5)$$

Since all inputs of the residual predictor $g(\cdot,\cdot)$ are SO(3)-invariant, we can predict a SO(3)-invariant residual orientation nonetheless even though we use Point Transformer (Zhao et al., 2021), which is not SO(3)-invariant, as our residual predictor.

With the SO(3)-equivariant orientation hypothesizer h and the SO(3)-invariant residual predictor g, we finally construct a SO(3)-equivariant network $f : \mathbb{R}^{N\times 3} \mapsto \mathrm{SO}(3)$ to predict the characteristic orientation of the point cloud $\mathbf{P}R$ as:

$$\begin{aligned} f(\mathbf{P}R) &= g(\mathbf{P}Rh(\mathbf{P}R)^\top, \phi(\mathbf{P}R)(\mathbf{V}R)^\top)h(\mathbf{P}R) \\ &= g(\mathbf{P}h(\mathbf{P})^\top, \phi(\mathbf{P})\mathbf{V}^\top)h(\mathbf{P}R) \\ &= g(\mathbf{P}h(\mathbf{P})^\top, \phi(\mathbf{P})\mathbf{V}^\top)h(\mathbf{P})R \\ &= f(\mathbf{P})R. \end{aligned} \quad (6)$$

Figure 1 illustrates the overview of f.

4.4. Self-Supervised Objective

Given two rotated point clouds of the same class, $\mathbf{P}_1R_1$ and $\mathbf{P}_2R_2$, their respective outputs from CHOIR *i.e.,* characteristic orientations would be $R'_1 = f(\mathbf{P_1}R_1)$ and $R'_2 = f(\mathbf{P_2}R_2)$, respectively. To achieve the consistency of the predicted characteristic orientations, we formulate our self-supervised objective ($\mathcal{L}$) as follows:

$$\begin{aligned} \mathcal{L} &= \|{R'_1}^\top R'_2 - R_1^\top R_2\|_F^2 \\ &= \|f(\mathbf{P}_1R_1)^\top f(\mathbf{P}_2R_2) - R_1^\top R_2\|_F^2 \\ &= \|R_1^\top f(\mathbf{P}_1)^\top f(\mathbf{P}_2)R_2 - R_1^\top R_2\|_F^2 \\ &= \|R_1^\top (f(\mathbf{P}_1)^\top f(\mathbf{P}_2) - I)R_2\|_F^2 \\ &= \|f(\mathbf{P}_1)^\top f(\mathbf{P}_2) - I\|_F^2 \\ &= \|f(\mathbf{P}_2) - f(\mathbf{P}_1)\|_F^2, \end{aligned} \quad (7)$$

where $I \in \mathbb{R}^{3\times 3}$ is the identity matrix. In essence, minimizing $\mathcal{L}$ enforces f to predict *consistent* characteristic orientations from two different point clouds $\mathbf{P}_1$ and $\mathbf{P}_2$. It is worth noting that we do not use the pre-aligned coordinate frame of a dataset, which is aligned by *humans*, as a ground truth canonical orientation to *fully* supervise f to align input point clouds to the ground truth. Instead, we construct a pair of point clouds using either augmentation (*e.g.*, kNN patch removal, resampling) or cross-instance sampling within the same class, and *self*-supervise the class-specific canonical orientation, which may differ from the pre-aligned coordinate frame.

5. Experiments

We perform qualitative and quantitative experiments to demonstrate the efficacy of our proposed method on the task of predicting the characteristic orientations of point clouds. Also, we analyze the significance of stability and consistency of characteristic orientation estimation on a downstream point cloud analysis task, the ShapeNet (Yi et al., 2016) part segmentation task.

5.1. Characteristic Orientation Prediction

Dataset and implementation details. We use the ShapeNet dataset (Yi et al., 2016) to train and test our model for characteristic orientation prediction of point clouds. Following VN-SPD (Katzir et al., 2022), we focus on four classes of the ShapeNet dataset: airplanes, chairs, tables, and cars. Note that airplanes and cars classes are more semantically consistent *i.e.*, contain less intra-class shape variation, while chairs and tables are less semantically consistent. We sample 1024 points randomly for each point cloud. We train our CHOIR network using the Adam optimizer (Kingma & Ba, 2015) at a learning rate of 0.01 for 3000 epochs. We use the checkpoint which shows the best validation sum of stability and consistency (lower the better) as our model for evaluation i.e., the model with the best compromise between stability and consistency on the validation set. To facilitate the analysis of the influence of stability and consistency on downstream tasks, we also propose to train a model that places more focus on consistency than stability,

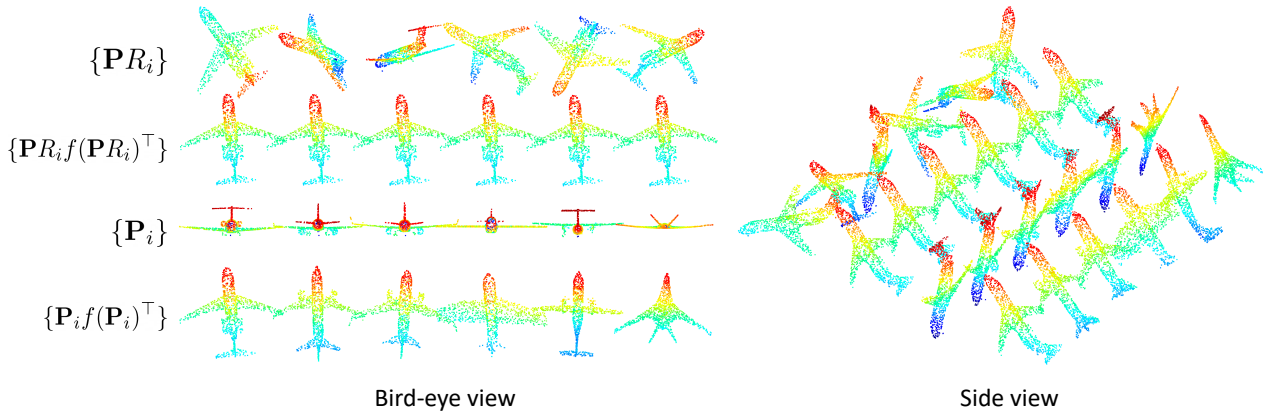

Figure 2. **Qualitative visualization of canonicalized Airplane class on the ShapeNet dataset.** The first two rows visualize the stability of our characteristic orientation predictions. Given differently rotated point clouds of the same instance $\{\mathbf{P}R_i\}$, CHOIR yields stable canonicalized point clouds $\{\mathbf{P}R_i f(\mathbf{P}R_i)^\top\}$ such that they are all well aligned. The lower two rows visualize the consistency of our characteristic orientation predictions. Given initially aligned different point clouds of the same class $\{\mathbf{P}_i\}$, CHOIR is able to obtain consistent canoncalizations $\{\mathbf{P}_i f(\mathbf{P}_i)^\top\}$ such that they are aligned. Note that the pre-aligned coordinate frame of the dataset differs from the learned canonical orientation for we use a self-supervised loss $\mathcal{L}$ to learn to predict characteristic orientations.

Table 1. **Single-class stability (°) and consistency (°) comparison on the ShapeNet dataset.** Lower is better. While VN-SPD demonstrates the best stability, CHOIR achieves the best consistency overall, and is just marginally behind VN-SPD in terms of stability.

Method	Airplane		Car		Chair		Table	
	Stability	Consistency	Stability	Consistency	Stability	Consistency	Stability	Consistency
Compass (2020)	13.81	71.43	12.01	68.20	19.20	87.50	74.80	115.3
Canonical Capsules (2021)	7.42	45.76	4.79	68.13	81.9	**11.1**	14.7	119.3
ConDor (2022)	35.93	118.00	34.52	109.55	25.98	122.08	29.68	**77.99**
VN-SPD (2022)	**0.02**	49.97	**0.04**	24.31	**0.02**	35.6	**0.02**	106.3
CHOIR (Ours)	0.67	**32.07**	0.24	**13.52**	0.33	18.91	0.85	101.23

Table 2. **Multi-class stability (°) and consistency (°) comparison on the ShapeNet dataset** Lower is better. It can be seen that CHOIR demonstrates the best stability and consistency overall, outperforming existing methods by a significant margin especially in terms of stability. Point Transformer* is a SO(3)-**variant** characteristic orientation predictor based on Point Transformer (Zhao et al., 2021), optimized in favor of consistency while neglecting stability; this model will be used to analyze the effect of stability and consistency in downstream point cloud analysis tasks.

Method	Airplane		Car		Chair		Table	
	Stability	Consistency	Stability	Consistency	Stability	Consistency	Stability	Consistency
Canonical Capsules (2021)	20.96	129.03	7.08	78.29	9.11	109.04	18.62	123.25
ConDor (2022)	31.05	122.86	34.17	113.83	27.07	118.89	31.22	128.89
VN-SPD (2022)	97.88	104.82	96.99	95.63	97.96	94.91	97.19	**97.72**
CHOIR (Ours)	**0.67**	**48.72**	**0.40**	**21.77**	**0.42**	**25.65**	**4.80**	103.64
Point Transformer* (2021)	86.12	16.00	85.89	21.19	85.44	13.07	87.13	18.36

which is referred to as Point Transformer*. Point Transformer* is based on a Point Transformer network (Zhao et al., 2021) that takes as input the SO(3)-equivariant feature $\phi(\mathbf{P}R)$ from our hypothesizer network, and the rotated input point cloud $\mathbf{P}R$. As the layers of Point Transformer violate SO(3)-equivariance, the stability of characteristic orientation predictions is neglected. We use its output to predict the characteristic orientation of the input as well.

For each class, we calculate the stability for each instance in the validation set by applying 10 random rotations using Eq. (1) and report their mean. In the case of consistency, we follow Eq. (2) and report the consistency for each class. To enforce the uniqueness of the Singular Value Decomposition process necessary to calculate the chordal L_2-mean of rotations, we use the mean rotation whose determinant is 1.

Results and analyses. For each method, we compare two types of models for this experiment: class-wise models and multi-class models. The class-wise models are trained

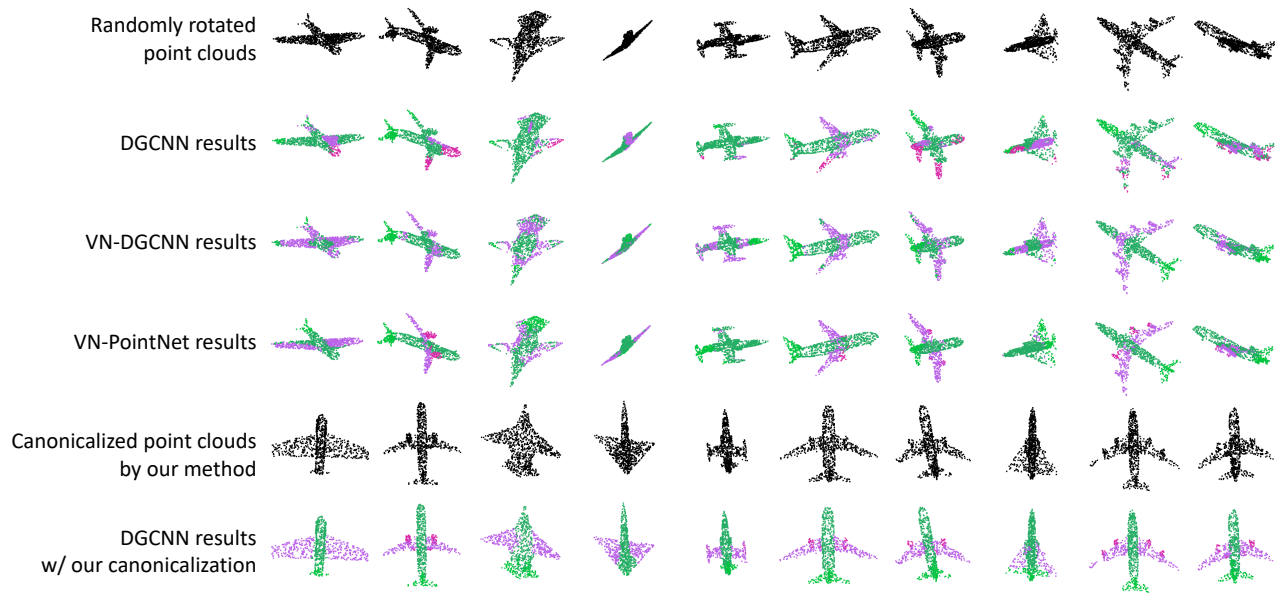

Figure 3. **Qualitative comparison of point cloud part segmentation on the ShapeNet (Yi et al., 2016) dataset.** We use the I/SO(3) setting, where no rotation augmentation is applied during training, but the point clouds are randomly rotated at test time. The first two rows visualize vanilla DGCNN (Wang et al., 2019), where it can be seen that the part segmentation results fail to differentiate the wings of the plane from the body. The next two rows visualize Vector Neuron-based networks (Deng et al., 2021), which can predict SO(3)-invariant point-wise part semantics. However, due to their limited representation power, these SO(3)-invariant networks still fail to discriminate detailed parts (*e.g.*, wings) of airplanes. Finally, the last two rows visualize DGCNN using CHOIR to canonicalize all input point clouds at train and test time. It can be seen that the input point clouds are well aligned, resulting in significantly better part segmentation results.

Table 3. **Class-wise mIoU (%) and Instance-wise mIoU (%) on ShapeNet for part segmentation.** The higher the better. We guide the readers to the main text for details on the I/I, I/SO(3) and SO(3)/SO(3) evaluation settings. Vanilla DGCNN, which does not use any aligning method, performs the best at I/I setting, hinting that no aligning method is perfectly stable *and* consistent. However, at I/SO(3) and SO(3)/SO(3) settings, CHOIR outperforms other methods. Meanwhile, Vector Neuron-based networks including VN-PointNet are all outperformed by DGCNN with aligning networks due to their limited representation power. Note that Point Transformer* achieved the best multi-class consistency, while showing substandard stability. This evidences that having an optimal balance between stability and consistency is the key to yielding better results on downstream tasks, rather than sacrificing stability for consistency.

Method	I/I		I/SO(3)		SO(3)/SO(3)	
	Cls. mIoU	Ins. mIoU	Cls. mIoU	Ins. mIoU	Cls. mIoU	Ins. mIoU
VN-PointNet (2021)	58.16	66.99	58.16	66.99	58.22	67.37
VN-DGCNN (2021)	52.72	63.18	52.72	63.18	52.52	62.85
DGCNN (2019)	**77.27**	**81.66**	25.81	29.06	58.59	69.09
+ Point Transformer* (2021)	66.25	<u>76.95</u>	43.44	51.41	62.95	73.15
+ Canonical Capsules (2021)	64.02	74.32	<u>63.92</u>	<u>74.23</u>	<u>64.45</u>	<u>74.69</u>
+ ConDor (2022)	59.00	69.71	59.07	69.81	58.87	69.75
+ VN-SPD (2022)	63.29	73.18	48.25	57.11	59.84	69.90
+ CHOIR (Ours)	<u>68.84</u>	76.64	**68.88**	**76.67**	**68.12**	**76.46**

on a single class of shapes, while multi-class models are trained on the union of the four ShapeNet classes used in our experiments, facilitating the comparison of multi-class training suitability between different methods. Table 1 shows the results of single-class models, and Table 2 presents those of multi-class models.

For single-class model comparison, it can be seen that CHOIR exhibits the best consistency overall - significantly outperforming baseline methods while being only slightly underperforming VN-SPD (Katzir et al., 2022) on the stability metric. In the case of multi-class model comparison, our model achieves the best stability and consistency compared to existing methods. This shows that using point reconstruction pipelines makes it unsuitable to handle multiple classes within a single model, in contrast to ours which is more adept at training with multiple classes of shapes. However,

Table 4. **Stability (°) and consistency (°) results of ablation study on the ShapeNet (Yi et al., 2016) airplane class.** Starting from VN-SPD (Katzir et al., 2022) as the baseline, we conduct an ablation study to verify the effect of each component of CHOIR. It can be seen that while removing the reconstruction pipeline has negative effects on both stability and consistency, finally incorporating our cross-instance training and residual predictor yields significantly better consistency results for a negligible decline in stability. Note that we empirically showed that removing the reconstruction pipeline makes our architecture (and the Point Transformer* architecture) more suitable to be trained using multiple classes of point clouds.

Method	w/o Recon.	Cross-instance	Residual Predictor $g(\cdot)$	Stability	Consistency
VN-SPD				0.02	49.97
CHOIR w/o cross-instance and $g(\cdot)$	✓			0.49	61.06
CHOIR w/o $g(\cdot)$	✓	✓		0.68	36.25
CHOIR (Ours)	✓	✓	✓	0.67	32.07

CHOIR still predicts characteristic orientations with the best balance between stability and consistency.

5.2. Part Segmentation

Dataset and implementation details. To analyze the influence of stability and consistency on downstream analysis tasks, we evaluate our method on the ShapeNet part segmentation dataset (Yi et al., 2016), specifically on the four data classes of airplane, chair, car, and table. We first pre-train the characteristic orientation prediction methods, including CHOIR, on the ShapeNet dataset under the multi-class training setting, then freeze their parameters. For each method, we then train a DGCNN (Wang et al., 2019) network for part segmentation, where its input point clouds are canonicalized using the predicted characteristic orientation by each characteristic orientation prediction method, at both training and testing. Following VNNs (Deng et al., 2021), we train the DGCNN network with Stochastic Gradient Descent optimizer at a learning rate of 0.001 for 200 epochs for all part segmentation experiments. We evaluate CHOIR against the baseline methods under three evaluation settings: I/I, I/SO(3), and SO(3)/SO(3). I/I provides no rotation to the input point cloud at both train and test time, such that all input point clouds are readily aligned at both train and test times. I/SO(3) applies random rotation to the point cloud only at test time, such that the input point cloud would be randomly oriented at test time. SO(3)/SO(3) applies random rotation to the point cloud at both train and test time, such that all input point clouds are randomly rotated. Note that applying random rotation to the point cloud at train time is the same as using rotation augmentations, but applying random rotations to the point cloud at test time differs from test-time augmentation as the former refers to viewing a single randomly rotated point cloud instance, while the latter refers to viewing multiple randomly rotated instances of the same input point cloud.

Results and analyses. We report the class-wise and instance-wise mIoU in Table 3. At the I/I setting, because the input point clouds are already aligned, vanilla DGCNN yields the best results - however, among methods that do use aligning methods, using CHOIR yields the best results. At the I/SO(3) setting, vanilla DGCNN shows extremely poor results, because the DGCNN network has not learned to segment rotated point clouds. CHOIR shows the best results in this setting, evidencing that canonicalizing the point clouds using our predicted characteristic orientation is the most beneficial. At the SO(3)/SO(3) settings, vanilla DGCNN shows an improved performance in comparison to the I/SO(3) setting, but underperforms in comparison to using aligning, proving that rotation augmentations are insufficient to handle randomly rotated point clouds at test time. CHOIR also achieves the best performance in this setting. Noting that Point Transformer* showed a significantly higher consistency compared to CHOIR, but CHOIR exhibits the best performance overall, our takeaway is that finding a harmonious balance between stability and consistency of characteristic orientation predictions is crucial for downstream analysis tasks.

5.3. Ablation Study and Analyses

Ablation study. We perform an ablation study to demonstrate the efficacy of each technique and component in CHOIR, starting from VN-SPD (Katzir et al., 2022) as the baseline, given that they share the same SE(3)-equivariant VNN (Deng et al., 2021)-based backbone network as CHOIR. The results are illustrated in Table 4.

First, it is evident that removing the point cloud reconstruction pipeline (Groueix et al., 2018) from VN-SPD deteriorates both the stability and consistency, which shows that the disentanglement of pose and shape by leveraging point cloud reconstruction is indeed helpful. Next, we verify that using cross-instance training *i.e.*, using point clouds pairs with different point clouds from the same class significantly improves the performance, already outperforming VN-SPD by a large margin of approximately $14°$. Finally, we demonstrate that incorporating our residual predictor, $g(\cdot)$, improves the consistency even further, without harming the stability of characteristic orientation predictions.

Further analysis on stability of CHOIR. While CHOIR

Table 5. **The numerical error of kNN indices.** We count the number of wrong kNN edges of rotated point clouds on ShapeNet (Yi et al., 2016) test dataset under two different precision levels. We set the cardinality of each point cloud as 1024 and k as 20, which means 20480 edges in total.

Dtype	Airplane	Car	Chair	Table
Float32	17.4	9.6	27.9	62.3
Float64	15.7	7.8	25.8	58.2

Table 6. **On the training pair and kNN fixes (single class).** We conduct the experiment on ShapeNet (Yi et al., 2016) airplane class.

Method	Stability	Consistency
VN-SPD (2022)	**0.02**	49.97
CHOIR (Ours)	0.67	**32.07**
CHOIR w/ fixed kNN	**0.02**	**32.07**
CHOIR w/ same-instance	0.14	47.08
CHOIR w/ same-instance & fixed kNN	0.03	47.08

Table 7. **On the training pair and kNN fixes (multi classes).** Similar to the result of single class experiment Table 6, ours shows a better stability with the same-instance pair and kNN fixes. We denote our CHOIR with the same-instance pair as CHOIR *, CHOIR with fixed kNN as CHOIR †, and CHOIR with both modifications as CHOIR $*^{\dagger}$, respectively.

Method	Airplane		Car		Chair		Table	
	Stability	Consistency	Stability	Consistency	Stability	Consistency	Stability	Consistency
VN-SPD (2022)	97.88	104.82	96.99	95.63	97.96	94.91	97.19	**97.72**
CHOIR (Ours)	0.67	**48.72**	0.40	**21.77**	0.42	**25.65**	4.80	103.64
CHOIR †	**0.02**	**48.72**	**0.02**	**21.77**	**0.02**	**25.65**	0.03	103.64
CHOIR *	0.04	56.63	**0.02**	26.76	0.03	35.87	0.23	107.10
CHOIR $*^{\dagger}$	**0.02**	56.63	**0.02**	26.76	**0.02**	35.87	**0.02**	107.10

strikes a much favorable balance between stablity and consistency compared to VN-SPD, VN-SPD shows slightly favorable stability by an average value of $0.5°$. Through further analyses, we determined that the floating point precision loss of 3D point coordinates causes the results of kNN on point clouds to differ before and after rotation (illustrated in Table 5), which leads to slight instabilities in the SO(3)-equivariant calculations of our network.

Another factor that affects the stability of CHOIR can be deduced from Table 4, where it can be seen that the usage of our residual predictor and cross-shape dataset pairs incurs slight declines in stability in return for significant gains in consistency. Instead of using cross-shape dataset pairs, we could train CHOIR using same-shape pairs with augmentation transformations to enforce consistency of characteristic orientation predictions while preserving stability.

We present the results of CHOIR in Table 6 for the single-class setting and Table 7 for the multi-class setting, when we fix the kNN indices for each point cloud beforehand at inference time, and when we use same-shape dataset pairs to train CHOIR. It can be seen that using fixed kNN indices results in state-of-the-art stability, demonstrating nearly zero stability even under the multi-class setting. Albeit VN-SPD does not use predetermined kNN indices, their training scheme focuses on same-shape pair training with stronger data augmentations, resulting in satisfactory stability under the single-class setting. Also, using same-shape pairs results in state-of-the-art stability in both single- and multi-class settings, with a slight increase in consistency in comparison to using cross-shape dataset pairs. While it would be favorable to be able to have consistent kNN predictions, this is difficult to facilitate in real-world scenarios; it would therefore be wiser to integrate same-shape dataset pairs together with cross-shape pairs to strike the best balance between stability and consistency of characteristic orientation predictions.

6. Conclusion

We have shown that CHOIR learns to assign a stable yet consistent characteristic orientation for an input point cloud. It leverages SO(3)-equivariant backbone networks to determine the characteristic orientation hypothesis of the input point cloud, and calibrates the hypothesis with the results of our novel residual predictor to successfully canonicalize the input point cloud. We have evidenced the superior stability and consistency of CHOIR in both quantitative and qualitative manners. Specifically, the experiment results on characteristic orientation prediction indicate that by removing the conventional point cloud reconstruction pipelines, CHOIR is more suitable to be trained using multiple classes of point clouds, being more effective for downstream tasks *e.g.*, point cloud part segmentation. Furthermore, by comparing different models with varying balances between stability and consistency on downstream tasks, we draw an insightful conclusion that the key to improving the performance of downstream tasks is an optimal compromise between stability and consistency.

Acknowledgement. This work was supported by the MX division of Samsung Electronics Co., Ltd., the Institute of Information & Communications Technology Planning & Evaluation (IITP) grants (No.2021-0-02068: AI Innovation Hub (50%), No.2022-0-00290: Visual Intelligence based on Multi-layered Visual Common Sense (40%), No.2019-0-01906: AI Graduate School Program at POSTECH (10%)) funded by Korea government (MSIT).

References

Averkiou, M., Kim, V. G., and Mitra, N. J. Autocorrelation descriptor for efficient co-alignment of 3d shape collections. In *Computer Graphics Forum*, 2016.

Chen, H., Liu, S., Chen, W., Li, H., and Hill, R. Equivariant point network for 3d point cloud analysis. In *CVPR*, 2021.

Choe, J., Park, C., Rameau, F., Park, J., and Kweon, I. S. Pointmixer: Mlp-mixer for point cloud understanding. In *ECCV*, 2022.

Deng, C., Litany, O., Duan, Y., Poulenard, A., Tagliasacchi, A., and Guibas, L. J. Vector neurons: A general framework for so(3)-equivariant networks. In *ICCV*, 2021.

Esteves, C., Allen-Blanchette, C., Makadia, A., and Daniilidis, K. Learning so(3) equivariant representations with spherical cnns. In *ECCV*, 2018.

Fang, J., Zhou, D., Song, X., Jin, S., Yang, R., and Zhang, L. Rotpredictor: Unsupervised canonical viewpoint learning for point cloud classification. In *3DV*, 2020.

Groueix, T., Fisher, M., Kim, V. G., Russell, B. C., and Aubry, M. A papier-mâché approach to learning 3d surface generation. In *CVPR*, 2018.

Gu, J., Ma, W.-C., Manivasagam, S., Zeng, W., Wang, Z., Xiong, Y., Su, H., and Urtasun, R. Weakly-supervised 3d shape completion in the wild. In *ECCV*, 2020.

Katzir, O., Lischinski, D., and Cohen-Or, D. Shape-pose disentanglement using se(3)-equivariant vector neurons. In *ECCV*, 2022.

Kim, S., Park, J., and Han, B. Rotation-invariant local-to-global representation learning for 3d point cloud. In *NeurIPS*, 2020.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Li, F., Fujiwara, K., Okura, F., and Matsushita, Y. A closer look at rotation-invariant deep point cloud analysis. In *ICCV*, 2021a.

Li, X., Li, R., Chen, G., Fu, C.-W., Cohen-Or, D., and Heng, P.-A. A rotation-invariant framework for deep point cloud analysis. *IEEE Transactions on Visualization and Computer Graphics*, 2021b.

Li, X., Weng, Y., Yi, L., Guibas, L. J., Abbott, A., Song, S., and Wang, H. Leveraging se (3) equivariance for self-supervised category-level object pose estimation from point clouds. In *NeurIPS*, 2021c.

Luo, S., Li, J., Guan, J., Su, Y., Cheng, C., Peng, J., and Ma, J. Equivariant point cloud analysis via learning orientations for message passing. In *CVPR*, 2022.

Mehr, E., Lieutier, A., Bermudez, F. S., Guitteny, V., Thome, N., and Cord, M. Manifold learning in quotient spaces. In *CVPR*, 2018.

Novotny, D., Ravi, N., Graham, B., Neverova, N., and Vedaldi, A. C3dpo: Canonical 3d pose networks for non-rigid structure from motion. In *ICCV*, 2019.

Qi, C. R., Su, H., Mo, K., and Guibas, L. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017a.

Qi, C. R., Yi, L., Su, H., and Guibas, L. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NIPS*, 2017b.

Rao, Y., Lu, J., and Zhou, J. Spherical fractal convolutional neural networks for point cloud recognition. In *CVPR*, 2019.

Rempe, D., Birdal, T., Zhao, Y., Gojcic, Z., Sridhar, S., and Guibas, L. J. Caspr: Learning canonical spatiotemporal point cloud representations. In *NeurIPS*, 2020.

Sajnani, R., Poulenard, A., Jain, J., Dua, R., Guibas, L. J., and Sridhar, S. Condor: Self-supervised canonicalization of 3d pose for partial shapes. In *CVPR*, 2022.

Shen, W., Zhang, B., Huang, S., Wei, Z., and Zhang, Q. 3d-rotation-equivariant quaternion neural networks. In *ECCV*, 2020.

Spezialetti, R., Stella, F., Marcon, M., Silva, L., Salti, S., and Di Stefano, L. Learning to orient surfaces by self-supervised spherical cnns. In *NeurIPS*, 2020.

Sun, W., Tagliasacchi, A., Deng, B., Sabour, S., Yazdani, S., Hinton, G. E., and Yi, K. M. Canonical capsules: Self-supervised capsules in canonical pose. In *NeurIPS*, 2021.

Sun, X., Lian, Z., and Xiao, J. Srinet: Learning strictly rotation-invariant representations for point cloud classification and segmentation. In *ACM MM*, 2019.

Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F., and Guibas, L. J. Kpconv: Flexible and deformable convolution for point clouds. In *ICCV*, 2019.

Thomas, N., Smidt, T., Kearnes, S., Yang, L., Li, L., Kohlhoff, K., and Riley, P. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv*, 2018.

Wang, H., Liu, Y., Dong, Z., and Wang, W. You only hypothesize once: Point cloud registration with rotation-equivariant descriptors. In *ACM MM*, 2022.

Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., and Solomon, J. M. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics*, 2019.

Weinmann, M., Jutzi, B., Hinz, S., and Mallet, C. Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2015.

Wu, W., Qi, Z., and Li, F. Pointconv: Deep convolutional networks on 3d point clouds. In *CVPR*, 2019.

Xiao, C. and Wachs, J. Triangle-net: Towards robustness in point cloud learning. In *WACV*, 2021.

Yi, L., Kim, V. G., Ceylan, D., Shen, I.-C., Yan, M., Su, H., Lu, C., Huang, Q., Sheffer, A., and Guibas, L. A scalable active framework for region annotation in 3d shape collections. In *SIGGRAPH Asia*, 2016.

Yu, R., Wei, X., Tombari, F., and Sun, J. Deep positional and relational feature learning for rotation-invariant point cloud analysis. *arXiv*, 2020.

Zhao, H., Jiang, L., Jia, J., Torr, P. H., and Koltun, V. Point transformer. In *ICCV*, 2021.

Zhao, Y., Birdal, T., Lenssen, J. E., Menegatti, E., Guibas, L., and Tombari, F. Quaternion equivariant capsule networks for 3d point clouds. In *ECCV*, 2020.

A. On the Capacity of CHOIR

In principle, using a large enough network with a strong enough supervision from a *large-scale* dataset should be competitive to our choice of leveraging SO(3)-equivariant networks which maintain the inductive bias on rotation while consequently incurring capacity limits. Using a large enough network to rely on strong enough supervision to learn equivariance has two disadvantages: 1) while it has to solely rely on a sufficiently large-scale dataset, it severely lacks sample efficiency, and still would not guarantee the generalization of equivariance, and 2) this poses difficulties in guaranteeing stability while aiming to improve consistency of predictions. On the other hand, our design of predicting the SO(3)-invariant residual while maintaining SO(3)-equivariance has two advantages: 1) our method is theoretically guaranteed to satisfy SO(3)-equivariance, and 2) we can therefore exploit sample efficiency in training. The main reason we maintain the inductive bias on rotation in CHOIR is for its high sampling efficiency, noting that we do not need to train our model with large-scale datasets with extensive rotation augmentations to learn equivariance to rotation.

To provide deeper insights into this matter, we train two different straightforward approaches - 1) the vanilla Point Transformer, and 2) our Point Transformer* architecture, which is Point Transformer followed by our SO(3)-equivariant hypothesizer - with varying Point Transformer sizes in terms of parameter. Note that both of these approaches break SO(3)-equivariance. Specifically, we newly train a total of six networks for the task of characteristic orientation prediction:

- PT-S: vanilla Point Transformer with 4 transformer blocks (the same number of blocks as CHOIR)
- PT-M: vanilla Point Transformer with 8 transformer blocks
- PT-L: vanilla Point Transformer with 16 transformer blocks
- PT*-S: SO(3)-equivariant hypothesizer + PT-S
- PT*-M: SO(3)-equivariant hypothesizer + PT-M
- PT*-L: SO(3)-equivariant hypothesizer + PT-L

Note that PT-variants tend to diverge when trained with the same learning rate as CHOIR, and were trained with 1/100 times the learning rate. For a fair comparison, we additionally train PT*- variants with this new learning rate (1/100 of that of CHOIR), which we name PT*-lr-S/M/L. The results are illustrated in Table 8.

Table 8. **Multi-class stability (°) and consistency (°) using SO(3)-variant backbones with varying capacity on ShapeNet.** The results do not show a clear correlation between stability/consistency and the model capacity. CHOIR shows a stronger characteristic orientation prediction performance in comparison. Note that VN-SPD has a SO(3)-equivariant encoder (341K parameters) and a reconstruction network (17,069K parameters).

Method	#Param(K)	Airplane		Car		Chair		Table	
		Stability	Consistency	Stability	Consistency	Stability	Consistency	Stability	Consistency
VN-SPD (2022)	17410	97.88	104.82	96.99	95.63	97.96	94.91	97.19	97.72
CHOIR (Ours)	314	**0.67**	48.72	**0.40**	21.77	**0.42**	25.65	**4.80**	103.64
PT-S	**92**	40.27	58.83	27.23	42.69	50.96	66.97	76.61	75.42
PT-M	178	28.25	64.79	19.02	43.52	19.16	43.52	65.59	86.31
PT-L	352	33.42	67.06	21.04	39.71	24.43	62.01	70.69	75.91
PT*-S	135	86.12	**16.00**	85.89	21.19	85.44	**13.07**	87.13	**18.36**
PT*-M	222	85.34	22.76	80.67	23.08	86.12	14.85	81.38	22.20
PT*-L	396	88.43	35.61	86.38	**12.05**	86.89	20.62	86.60	26.13
PT*-lr-S	135	85.38	52.63	86.63	28.17	87.70	33.07	86.45	35.60
PT*-lr-M	222	11.25	62.64	7.69	32.73	8.34	40.49	42.19	83.29
PT*-lr-L	396	85.65	36.15	84.71	24.87	84.86	45.07	84.81	52.25

The results show that the stability or consistency of characteristic orientation predictions does not correlate with the model capacity (# parameters), and the point transformer variants overall show a much worse stability-consistency tradeoff in comparison with CHOIR. We suspect that this is mainly because the dataset used is not large enough to learn both the stability and consistency better than CHOIR, which exploits the sampling efficiency of SO(3)-equivariant networks. An interesting observation is seen in PT*-lr-M, which actually demonstrates rather satisfactory stability compared to other PT- or PT*- variants, but they tend to show worse consistency in return. Overall, as of now, the capacity limits enforced by the usage of SO(3)-equivariance yields better results compared to models with larger learning capacities, which we mainly suspect is due to the lack of large-scale datasets (and conversely the sampling efficiency of SO(3)-equivariant networks).

B. Effect of Using Test-time Augmentation

We present additional quantitative results when applying test-time augmentation for the task of characteristic orientation prediction. Specifically, we use point resampling and Gaussian noise data augmentations for data augmentation at test time. The results are shown in Table 9 and Table 10, where it can be seen that applying resampling or noise augmentations at test time leads to worse results overall. Nonetheless, CHOIR still achieves the best stability-consistency tradeoff overall.

Table 9. **Multi-class stability (°) and consistency (°) comparison on the ShapeNet dataset using resampling augmentation at test time.** Lower is better. It can be seen that CHOIR strikes the most favorable balance between stability and consistency overall.

Method	Airplane		Car		Chair		Table	
	Stability	Consistency	Stability	Consistency	Stability	Consistency	Stability	Consistency
Canonical Capsules (2021)	76.26	129.03	57.16	78.29	46.32	109.04	85.40	123.25
ConDor (2022)	98.87	122.86	92.44	113.83	87.01	118.89	98.38	128.89
VN-SPD (2022)	100.21	104.82	100.87	95.63	99.98	94.91	101.25	97.72
CHOIR (Ours)	**13.53**	48.72	**7.69**	21.77	**7.56**	25.65	**66.67**	103.64
Point Transformer* (2021)	86.38	**16.00**	86.08	**21.19**	85.72	**13.07**	87.56	**18.36**

Table 10. **Multi-class stability (°) and consistency (°) comparison on the ShapeNet dataset using Gaussian noise augmentation at test time.** Lower is better. It can be seen that CHOIR strikes the most favorable balance between stability and consistency overall.

Method	Airplane		Car		Chair		Table	
	Stability	Consistency	Stability	Consistency	Stability	Consistency	Stability	Consistency
Canonical Capsules (2021)	22.81	129.03	9.23	78.29	10.56	109.04	21.52	123.25
ConDor (2022)	58.39	122.86	50.89	113.83	46.90	118.89	54.61	128.89
VN-SPD (2022)	98.42	104.82	97.31	95.63	98.41	94.91	98.51	97.72
CHOIR (Ours)	**12.00**	48.72	**5.91**	21.77	**5.34**	25.65	45.30	103.64
Point Transformer* (2021)	86.88	**16.00**	86.09	**21.19**	85.73	**13.07**	87.34	**18.36**

C. Additional Qualitative Results

In this appendix, we provide more qualitative results of our method on both characteristic orientation estimation (Figures 4,5,6,7, and 8) and part segmentation tasks (Figures 9, 10, and 11).

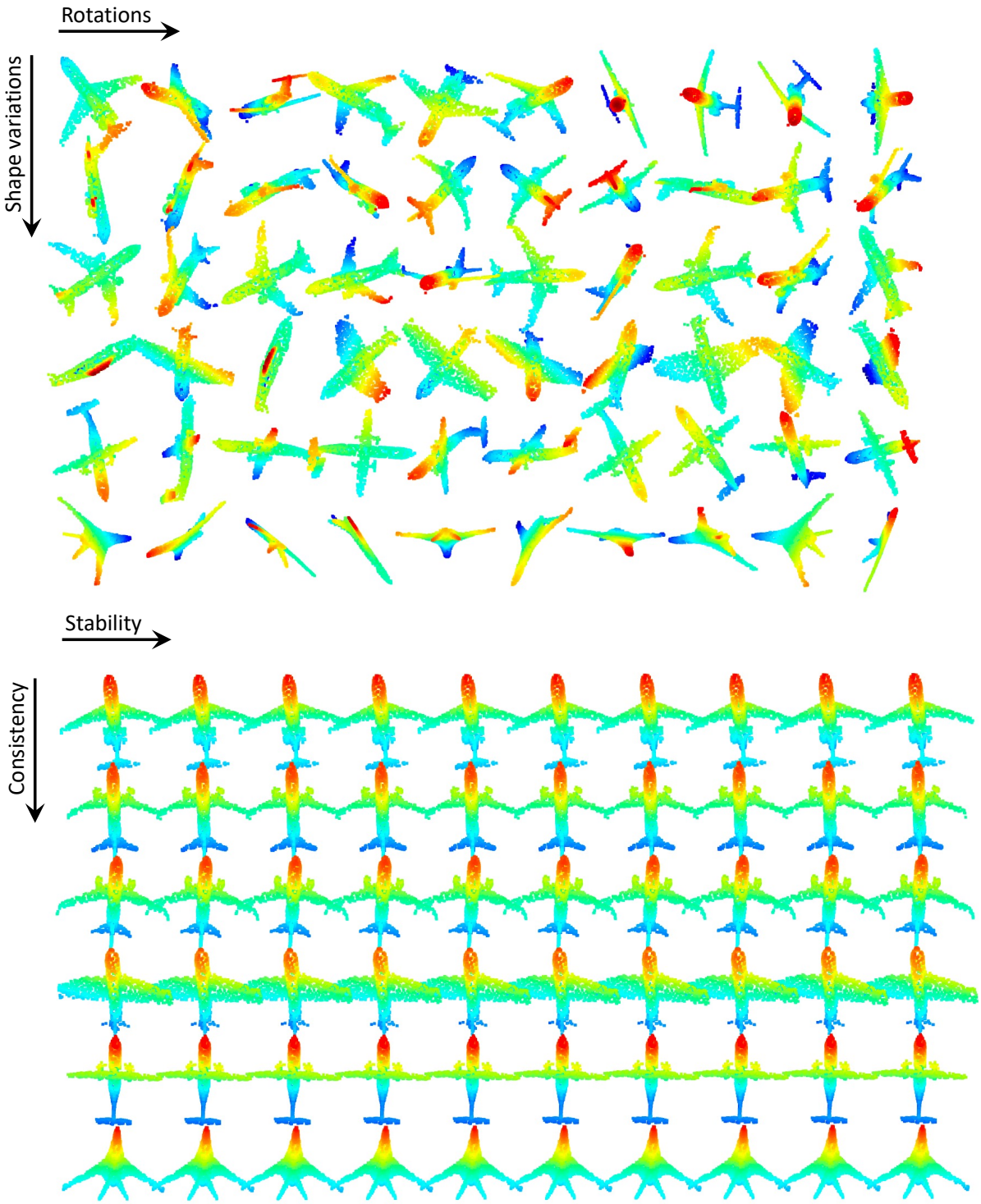

Figure 4. **Qualitative visualization of canonicalized airplanes by our method (single-class) on the ShapeNet dataset.**

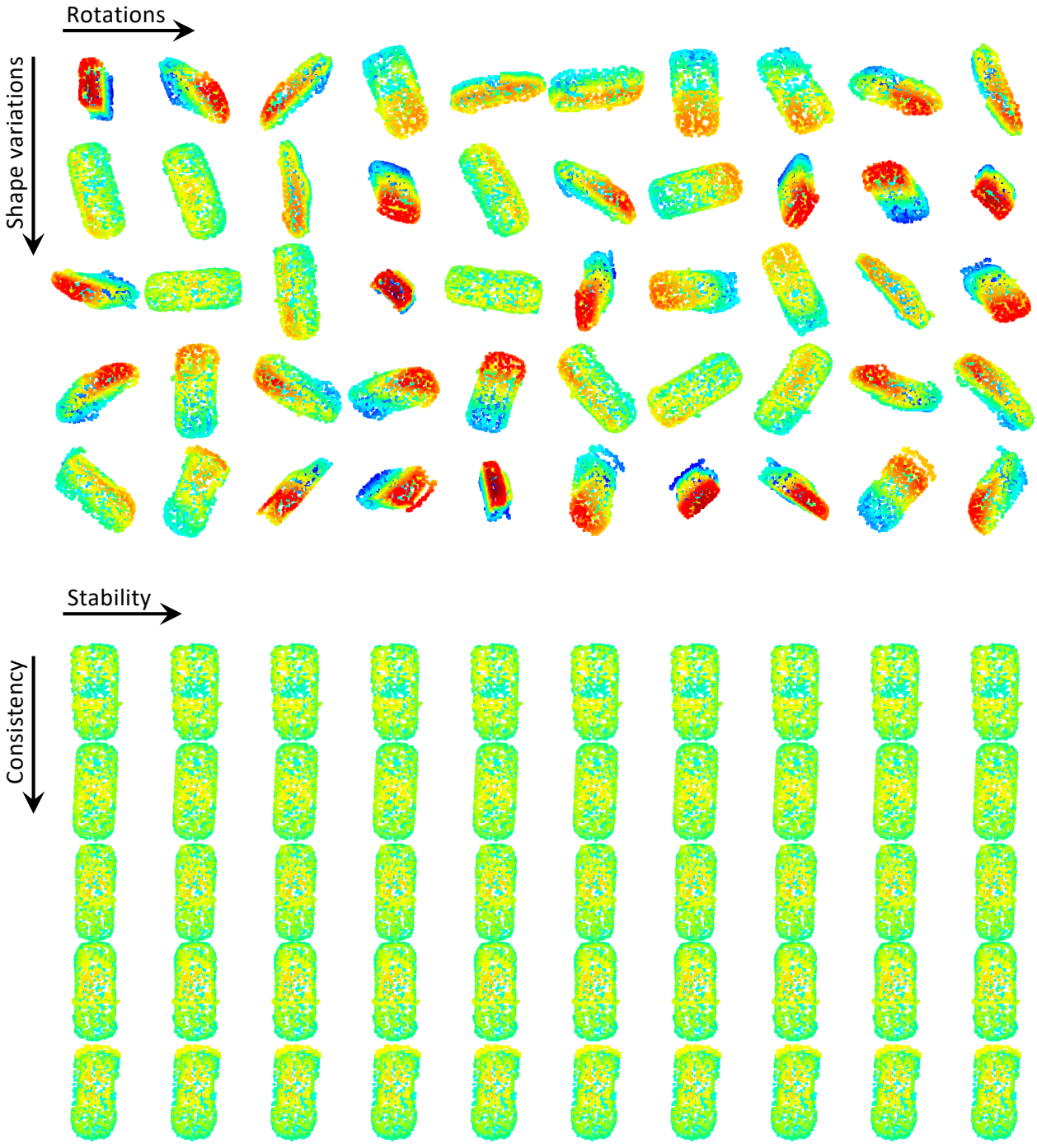

Figure 5. **Qualitative visualization of canonicalized cars by our method (single-class) on the ShapeNet dataset.**

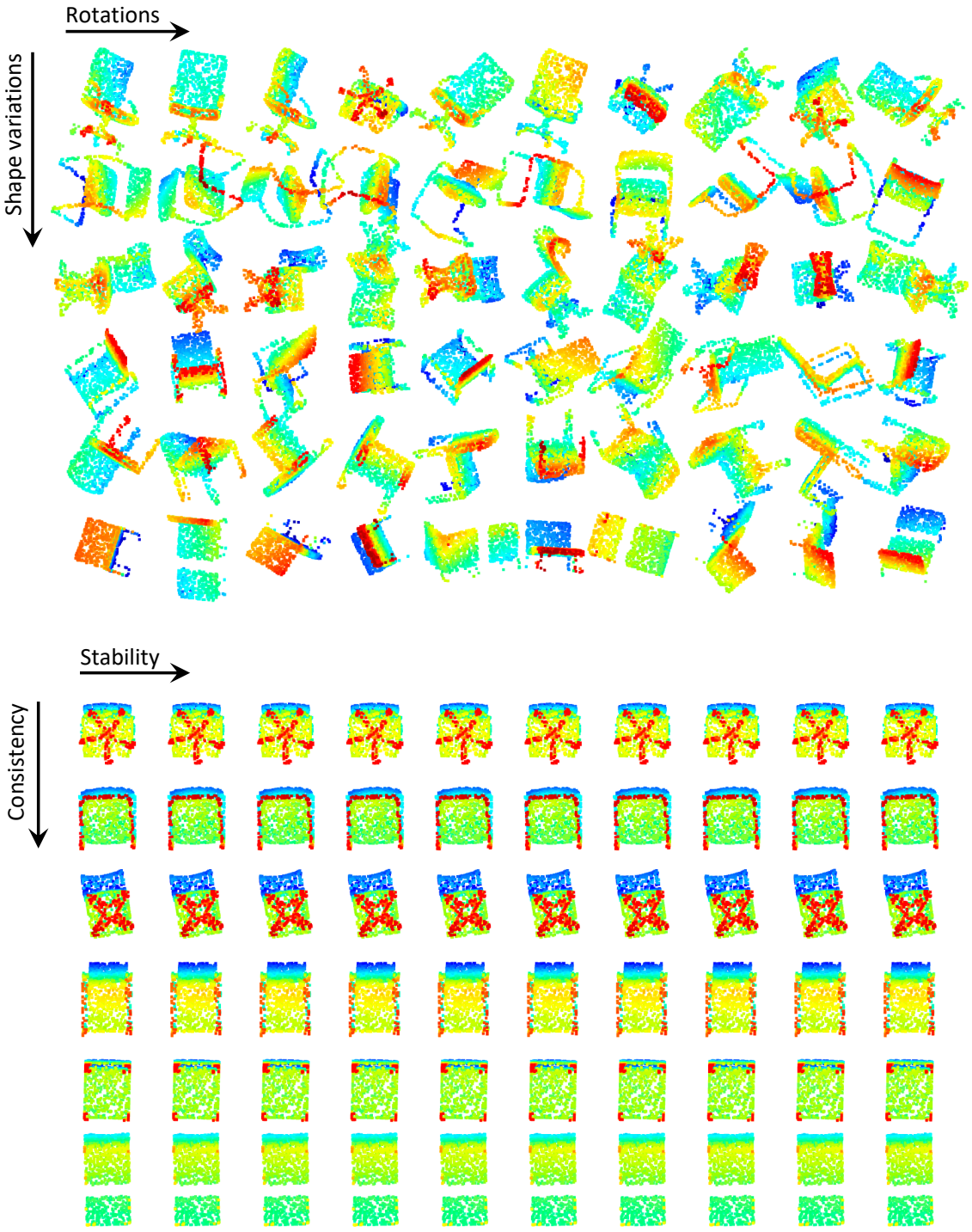

Figure 6. **Qualitative visualization of canonicalized chairs by our method (single-class) on the ShapeNet dataset.**

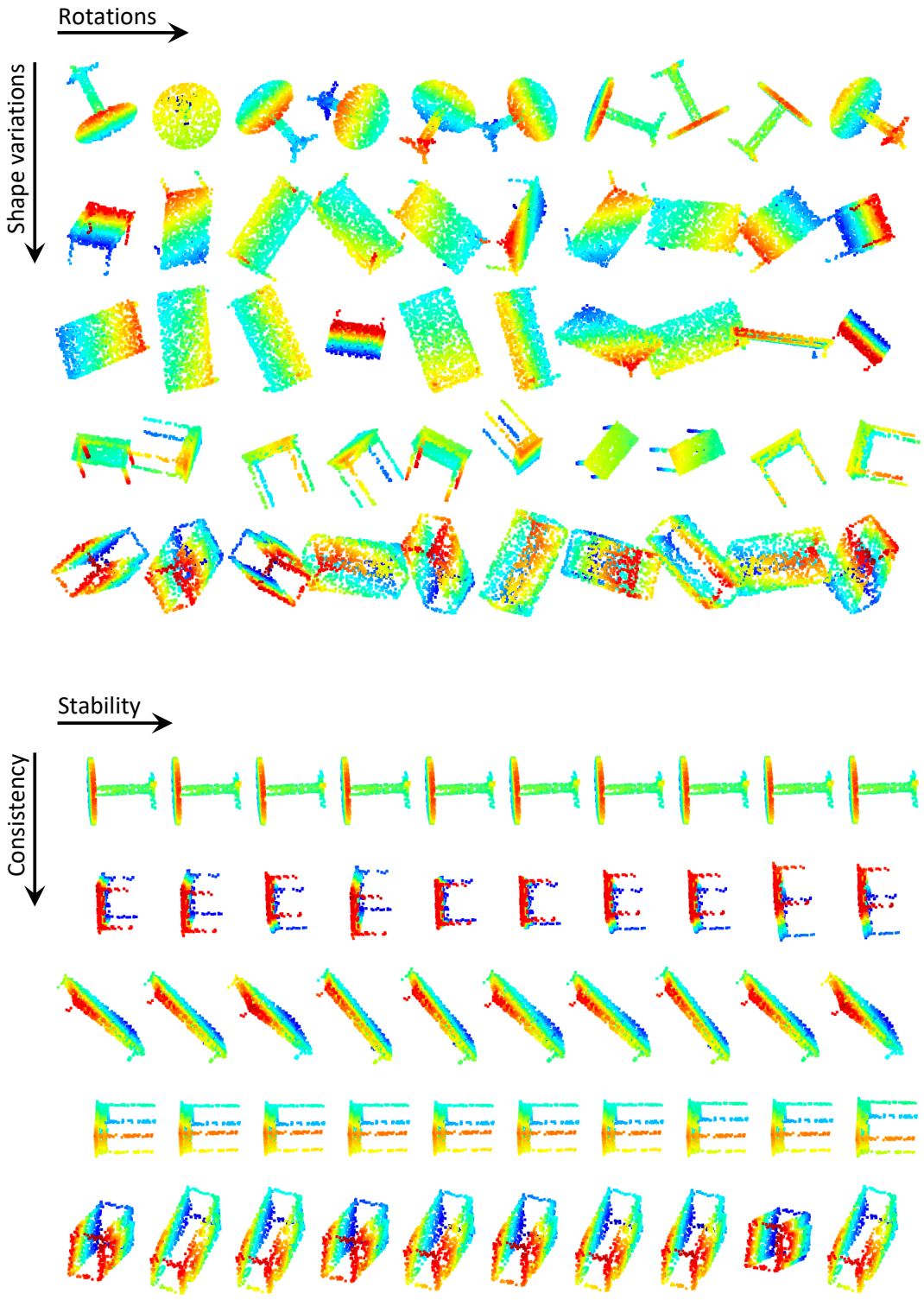

Figure 7. **Qualitative visualization of canonicalized tables by our method (single-class) on the ShapeNet dataset.**

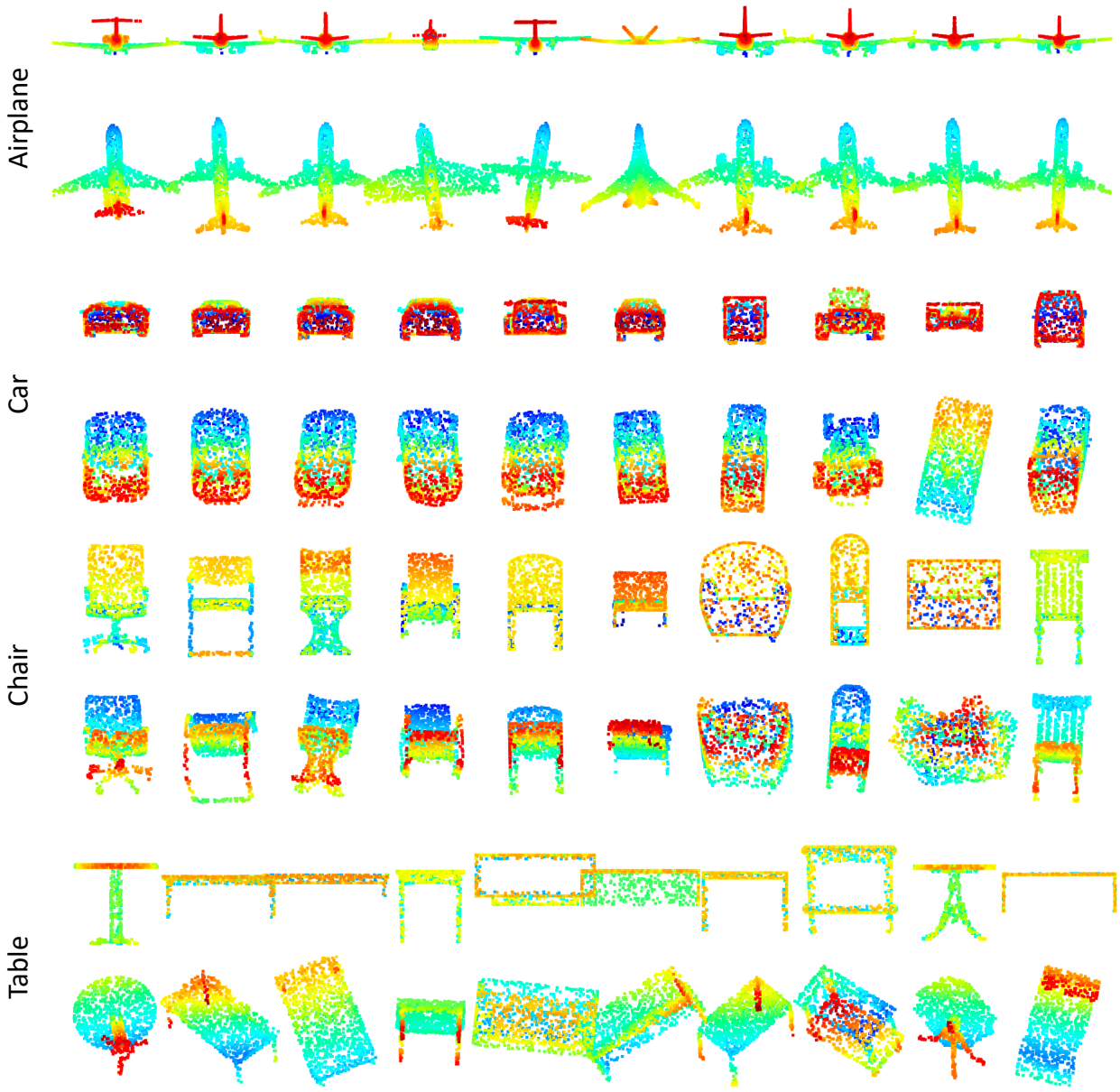

Figure 8. **Qualitative visualization of canonicalized point clouds of four classes.** All point clouds are canonicalized by our method, which is trained with multi-class training.

Figure 9. **Qualitative visualization of part segmentations of our method (car, I/SO(3)) on the ShapeNet dataset.**

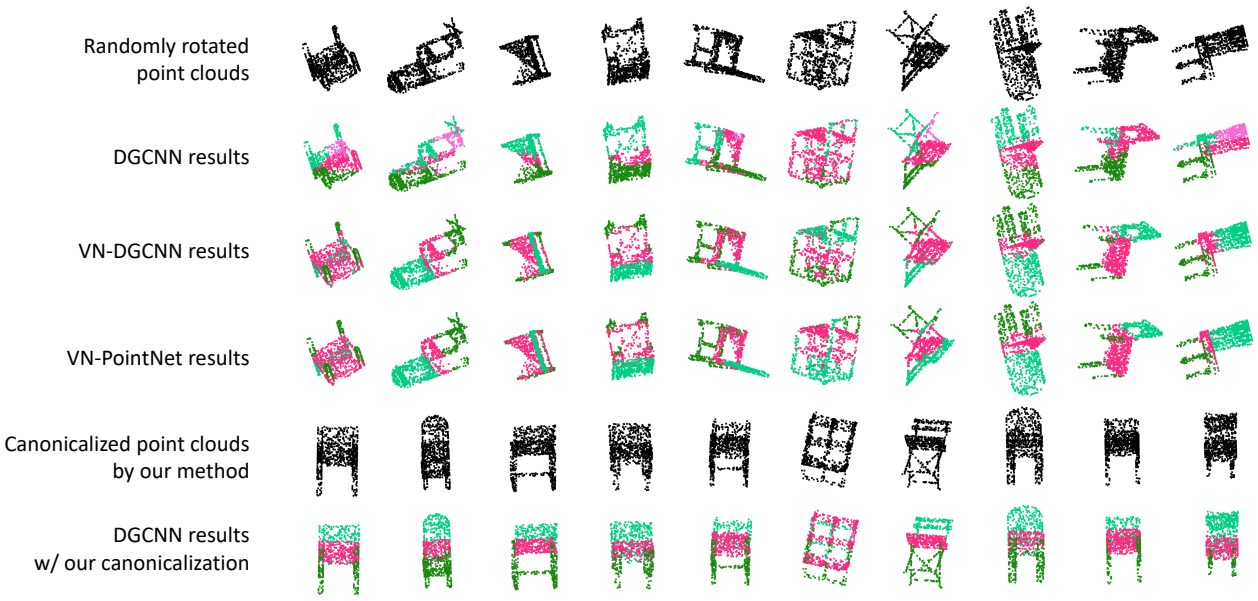

Figure 10. **Qualitative visualization of part segmentations of our method (chair, I/SO(3)) on the ShapeNet dataset.**

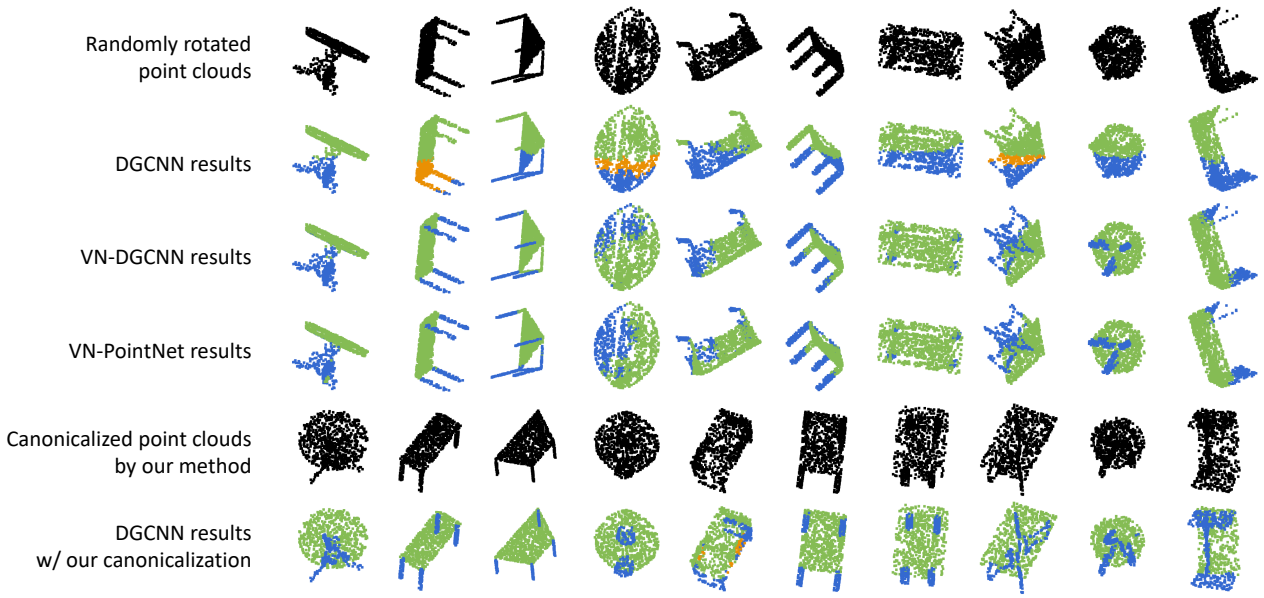

Figure 11. **Qualitative visualization of part segmentations of our method (table, I/SO(3)) on the ShapeNet dataset.**