SC3K: Self-supervised and Coherent 3D Keypoints Estimation from Rotated, Noisy, and Decimated Point Cloud Data

Mohammad Zohaib, Alessio Del Bue
Pattern Analysis & Computer Vision (PAVIS)
Italian Institute of Technology (IIT), Genoa, Italy
{mohammad.zohaib, alessio.delbue}@iit.it

Abstract

This paper proposes a new method to infer keypoints from arbitrary object categories in practical scenarios where point cloud data (PCD) are noisy, down-sampled and arbitrarily rotated. Our proposed model adheres to the following principles: i) keypoints inference is fully unsupervised (no annotation given), ii) keypoints position error should be low and resilient to PCD perturbations (robustness), iii) keypoints should not change their indexes for the intra-class objects (semantic coherence), iv) keypoints should be close to or proximal to PCD surface (compactness). We achieve these desiderata by proposing a new self-supervised training strategy for keypoints estimation that does not assume any a priori knowledge of the object class, and a model architecture with coupled auxiliary losses that promotes the desired keypoints properties. We compare the keypoints estimated by the proposed approach with those of the state-of-the-art unsupervised approaches. The experiments show that our approach outperforms by estimating keypoints with improved coverage (+9.41%) while being semantically consistent (+4.66%) that best characterize the object's 3D shape for downstream tasks. Code and data are available at: https://github.com/IIT-PAVIS/SC3K

1. Introduction

Representing 3D objects using a set of keypoints [2, 10, 28] is a common and fundamental step for several geometrical reasoning tasks, including pose estimation, action recognition, object tracking, shape registration, deformation, retrieval and reconstruction [23, 35, 30, 42, 9]. As being a first processing step, it is crucial that the keypoints (see Fig. 1) are extracted reliably from point cloud data (PCD) of object shapes, as any error may negatively impact further higher-level tasks.

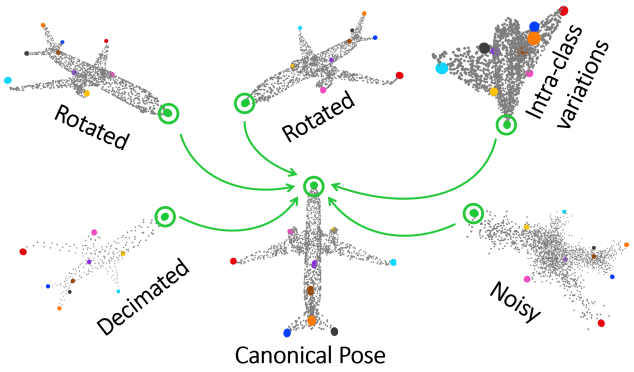


Figure 1: Self/un-supervised keypoints estimation from PCD has to be robust to perturbations such as rotations, intra-class shape variations, noisy data and an arbitrary number of input 3D points. The keypoint localization needs to be not only accurate and pertain to the object surface, but also preserve semantic coherence, as the green keypoint is always associated with a specific object region despite arbitrary variations in the PCD.

The solution to this problem was initially cast as a supervised learning task: given a dataset of manually annotated PCDs with keypoints, a computational model infers the keypoints position given a PCD as input [32, 45, 44, 13, 8, 36]. While these methods provided impressive results on the dataset they were trained on, they also highlighted the limitations of supervised approaches. The basic issue is the requirement of having large enough datasets containing well-defined ground truth annotations for every object. Annotating such datasets is difficult as finding 3D keypoints manually is a hard and time consuming activity. Similarly, noise or missing data on the PCD can compromise quality, and highly symmetric/smooth objects might confuse the annotator in finding the correct keypoints.

Considering such limitations, recent methods have focused on not-supervised approaches to bypass the need

for human annotations. Self-supervision methods define proxy tasks for which a large number of annotations can be obtained during training [31, 18, 35, 1, 39], e.g., geometrical transformations, canonical mapping, reconstruction to learn the prototype of intra-class object, etc. [22, 41, 21, 25, 27]. Unsupervised approaches differently promotes keypoints that are implicitly given by reasoning on the object geometry, e.g. point-level clustering, object's skeleton, consistency between object's symmetry, part contrasting, etc. [15, 33, 24, 9, 37].

The shift towards these learning paradigms clearly allows generalizing keypoint extraction but not without drawbacks. Without human annotations, it is difficult to identify a specific keypoint in a particular semantic 3D region when intra-class variations are present (airplane example in Fig. 1). Moreover, for several applications such as shape registration, it is paramount to maintain the semantic consistency of keypoints, i.e., their vector ordering. Despite these considerations, keypoints extraction has to be robust against common perturbations of PCDs, and the accuracy in localizing the keypoints should be preserved even if PCDs are rotated, noisy and decimated as shown in Fig. 1.

To this end, we propose an approach that reduces the requirement of ground truth labels by utilizing the input PCDs to learn to produce 3D keypoints on the object's surface. It generates two versions of an object by applying a random rotation (as done on images in the methods presented in [7, 14]) and estimates the corresponding keypoints set.

Initially, the network optimizes the keypoints of the individual objects to promote non-overlapping, proximal to the input PCD, and covering the complete object. To ensure consistency in the semantic coherence (order) and positions of the estimated keypoints, the network compares the keypoints of both versions of the input PCD. First, both sets are transformed to the canonical pose and are compared one-to-one between the corresponding keypoints of the sets. Second, as a proxy task, the relative pose between the two sets of keypoints is estimated and minimized against the known relative pose of the PCDs pair. Such learning strategy and network architecture promote the inference of keypoints that are semantically coherent, robust to perturbations, and with better accuracy.

The main contributions of this work are as follows:

- The proposed approach estimates 3D keypoints (from a single PCD), without the need to pre-align a PCD to a canonical pose;
- The presented mutual learning procedure allows to estimate keypoints that are semantically consistent for intra-class objects regardless of perturbations, such as rotation, noise, or down-sampling;
- On an average, the presented approach outperforms the state-of-the-art (SOTA) approaches (coverage: +9.41%, semantic consistency: +4.66%) and is able to generalize to novel object poses.

The rest of the paper is organized as follows; Section 2 presents recent keypoints estimation approaches along with their positive features and limitations, Section 3 describes the proposed approach which is evaluated in Section 4, Section 5 reports the ablations, finally conclusions are given in Section 6.

2. Related works

Several methods have been proposed to estimate 3D keypoints in a supervised way using human-annotated keypoints [32, 6, 40, 12, 45, 11, 36]. As our approach is unsupervised, here we review in more detail the methods that do not use supervision.

Chen et al. [2] present an approach that learns to identify semantically consistent points in the same category in an unsupervised way from an object's PCD. Their network is based on PointNet++ [20] that assigns a probability (of being a keypoint) to each element of the PCD. The final keypoints are computed using a convex combination of the points weighted by the probabilities. Yuan et al. [37] present an approach that uses two different objects of the same category to estimate semantically ordered 3D keypoints. Another similar approach that finds correspondences between different objects of the same category is presented in [3]. Li et al. [10] present a similar approach that first generates another variant of the PCD by random transformation and then utilizes both PCDs for estimating the keypoints. Their network first generates clusters from the input point clouds and then it estimates a keypoint for every cluster. A similar approach is presented by Sun et al. [25]. Their network takes two randomly rotated versions of a PCD and computes K capsules containing the attention mask for every point in the input PCD and the corresponding features. Based on the attention masks, points are arranged to K parts of the object. Fernandez et al. [4] present an approach that estimates symmetric 3D keypoints from PCD. The network estimates N nodes and applies nonmax-suppression for selecting the final keypoints. However, the approach is very sensitive to object symmetry; thus, its performance may decrease for irregular shapes, i.e., airplanes or guitars, whose geometries vary consistently within the category [23]. The authors in [23] present "Skeleton Merger" (SM) to detect aligned and semantic keypoints from PCDs in an unsupervised fashion. It uses the keypoints to generate a skeleton of the object. Both keypoints and the skeleton are used to reconstruct the PCD. Xue et al. present USEEK [34], a teacher-student network

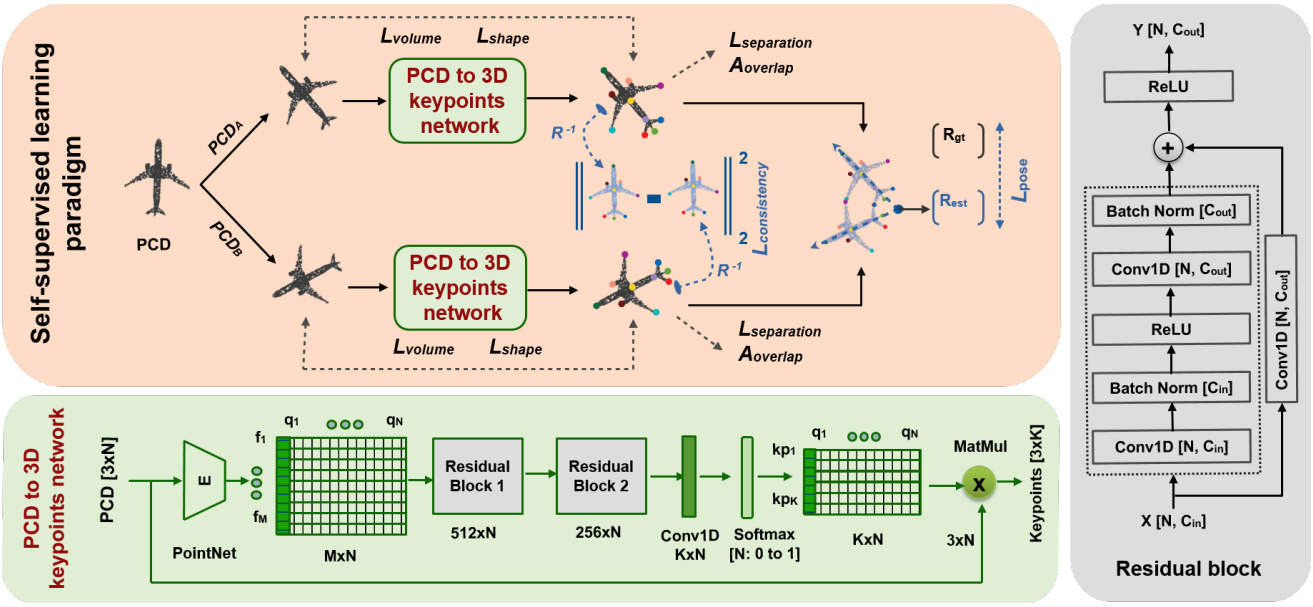


Figure 2: PCD to 3D keypoints network (light lime block) takes a PCD of N points as input and extracts M global features using PointNet encoder. The features are passed by two cascaded residual blocks (gray block) followed by a convolutional and a softmax layer in order to estimate $K \times N$ probabilities that are used to estimate K keypoints. The light orange block shows the proposed self-supervised learning paradigm. The method first estimates keypoints for two randomly rotated versions of the same PCD, and then uses them to minimize the individual (volume, shape, separation and average overlap) and mutual (consistency and pose – as highlighted in navy blue) losses.

that estimates an equivariant set of 3D keypoints from point clouds. Their teacher module is the same as [23] which is based on the PointNet++, whereas the student module uses a SPRIN/Vector Neuron SE(3)-invariant backbone. Their network first generates pseudo labels that are required later to train the student module. Tang et al. propose "LAKe-Net" [26] that uses the keypoints for the shape completion task. It localizes the aligned keypoints, generates surface-skeleton using the keypoints, and uses them to refine the object's shape. Suwajanakorn et al. present an approach [29] to estimate 3D keypoints in the form of 2D positions and depth from a pair of images. Their approach forces 2D keypoints to be estimated within the object silhouette and uses known camera projections. They use their keypoints to estimate the relative pose between objects.

Considering the limitations of the above-reported literature, this work presents an end-to-end architecture that does not require ground truth labels/silhouettes; rather, it utilizes mutual consistency (relative pose/order) between two versions of the same object as a proxy task to improve the 3D position and semantic coherence of the estimated keypoints. Furthermore, our design and selection of the loss functions allow the keypoints to be estimated proximal to the object's surface.

3. Proposed approach - SC3K

Given a PCD of an object, the goal of the proposed approach, named SC3K, is to estimate keypoints that are semantically coherent and accurate despite arbitrarily rotated PCDs and perturbations without requiring ground truth annotations. The architecture of the SC3K is illustrated in Fig. 2. In the following subsections, we describe the network to estimate 3D keypoints from the PCDs, and our self-supervised learning paradigm.

3.1. PCD to 3D keypoints network ( )

The PCD to 3D keypoints network (light lime block in Fig. 2) uses a PointNet [19] backbone to extract M features for every point in the input PCD. The extracted features pass through two consecutive residual blocks that reduce the features from M to 256. Each residual block (gray block ( ) in Fig. 2) contains a pair of Conv1D layers with batch normalization connected via ReLU, and a skip connection with a single Conv1D layer. The refined features are later projected to a conv1D and a softmax layer to estimate $K \times N$ features, where K represents the total number of keypoints and N represents the weight/probability for every point in the input point cloud to be selected as the keypoint. The weights of every keypoint $(N \times 1)$ are multiplied

to the input PCD ($3 \times N$) in order to estimate the final keypoint (3×1). The final keypoint represents a weighted average point of the PCD. We repeat this process K times to estimate all the keypoints ($3 \times K$).

3.2. Self-supervised learning paradigm ( )

The proposed learning paradigm (as shown in the light orange block in Fig. 2) accepts as input an object PCD that is then randomly rotated twice to obtain two PCDs. These PCDs are then processed by the PCD to 3D keypoints network that outputs two sets of keypoints. This pairwise set will be used as a self-supervised signal to enforce keypoints semantic consistency. For each set of keypoints a loss with four components is computed, based on how well the keypoints fit the shape of the input PCD. We call this loss "individual loss". Then, the two sets of keypoints from the two randomly rotated PCDs are used to compute "mutual dependency loss" that contains two components. In the first component, both the keypoints sets are transformed to the (known) canonical pose to compute the one-to-one consistency between the corresponding keypoints. In the second component, the relative pose of the keypoints are compared with those of the input PCDs to refine the keypoints position and the semantic coherence. The network is trained to minimize both loss functions. In the following, we present these two losses in detail.

3.3. Individual loss

The individual loss is computed for a single shape $\mathcal{PCD} = [p_1, p_2, ..., p_N] \in \mathbb{R}^{3\times N}$ and it outputs a set of K keypoints $\mathcal{KP} = \{k_1, k_2, ..., k_K\}, \in \mathbb{R}^{3\times K}$ with $K \ll N$. The desired properties of the keypoints are that they should be relatively separated, covering as much as possible the whole object's volume while still being close to the PCD, and not overlapping with each other. These properties are described next.

Separation loss: This loss ($\mathcal{L}_{sep}$) maximizes the distance of every keypoint (k_i) from its neighbouring keypoint ($kNN(k_i, \mathcal{KP})$) in $\mathcal{KP}$ thus promoting more spread out configurations of points. It is defined as:

$$\mathcal{L}_{sep} = \frac{1}{max\left(\frac{1}{K}\sum_{i=1}^{K}\left\|k_i - kNN(k_i, \mathcal{KP})\right\|_2, 0.01\right)}, \tag{1}$$

where the term 0.01 is used to avoid the infinite loss value, which can occur if all the keypoints are estimated at the same position.

Shape loss: Since $\mathcal{L}_{sep}$ moves away keypoints from their neighbours without any maximum distance limit, keypoints might move easily far from the object and even further. Therefore, we use the shape loss ($\mathcal{L}_{shape}$) that enforces keypoints being closer to the object's shape. The loss minimizes the distance of every keypoint k_i in $\mathcal{KP}$ from its nearest neighbour point in the input $\mathcal{PCD}$. The loss can be defined as:

$$\mathcal{L}_{shape} = \frac{1}{K}\sum_{i=1}^{K}\left\|k_i - kNN(k_i, \mathcal{PCD})\right\|_2. \tag{2}$$

Volume loss: The $\mathcal{L}_{sep}$ and $\mathcal{L}_{shape}$ losses do not consider how the keypoints are distributed over the whole shape of the object. Therefore to estimate keypoints that cover the entire object, we propose the volume loss as $\mathcal{L}_{volume}$. The loss computes the difference between the 3D volume of the estimated keypoints with that of the input PCD as:

$$\mathcal{L}_{volume} = \left\|vol(\mathcal{KP}) - vol(\mathcal{PCD})\right\|, \tag{3}$$

where $vol()$ is a function that computes a volume from a set of points in terms of a 3D bounding box enclosing the points [4]. To find the difference in volume, we use smooth L1 loss as this loss is less sensitive to outliers compared to the MSE loss [5].

Average overlap: To avoid multiple keypoints being estimated at the same 3D position, we compute the average overlap $\mathcal{A}_{overlap}$ among the keypoints as:

$$\mathcal{A}_{overlap} = \frac{1}{K^2}\sum_{i=1}^{K}\sum_{j=1}^{K}\left[\left\|k_i - k_j\right\|_2 < \tau_1\right], \quad i \neq j$$
$$\left[\left\|k_i - k_j\right\|_2 < \tau_1\right] = \begin{cases} 1 & \text{if true} \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where [.] is the Iverson bracket. Two keypoints are considered as overlapping if the Euclidean distance between them is less than the threshold τ_1, which is 0.05. We add this number to the overall individual loss.

The total individual loss can be summarized as a weighted sum of the above loss components;

$$\mathcal{L}_{individual} = w_{sep}\cdot\mathcal{L}_{sep} + w_{sh}\cdot\mathcal{L}_{shape} + w_{vol}\cdot\mathcal{L}_{volume} + w_{ovr}\cdot\mathcal{A}_{overlap}, \tag{5}$$

where, $\{w_{sep}, w_{sh}, w_{vol}, w_{ovr}\}$ are not optimised hyperparameters fixed to {0.5,6,1,0.07} respectively.

3.4. Mutual dependency loss

In order to refine the positions of the keypoints and to make them semantically coherent across different rotations, we use the mutual dependency loss. Differently from the individual loss, here we consider the pair of keypoints obtained from the randomly rotated shapes. The loss is given by two components as described below.

Suppose that the two randomly rotated versions of the input PCD are $\mathcal{PCD}_{\mathcal{A}} = [a_1, a_2, ..., a_N] \in \mathbb{R}^{3\times N}$ and $\mathcal{PCD}_{\mathcal{B}} = [b_1, b_2, ..., b_N] \in \mathbb{R}^{3\times N}$ while the K keypoints estimated by the proposed approach for each PCD version can be represented as $\mathcal{KP}_{\mathcal{A}} = [k_1^a, k_2^a, ..., k_K^a] \in \mathbb{R}^{3\times K}$ and $\mathcal{KP}_{\mathcal{B}} = [k_1^b, k_2^b, ..., k_K^b] \in \mathbb{R}^{3\times K}$, respectively. Then the loss functions can be described as given below.

Keypoints consistency loss: Consider that $R_a \in \mathbb{R}^{3\times3}$ and $R_b \in \mathbb{R}^{3\times3}$ are the rotations associated to $\mathcal{PCD}_{\mathcal{A}}$ and $\mathcal{PCD}_{\mathcal{B}}$, respectively. We use these rotation matrices and transform the keypoints ($\mathcal{KP}_{\mathcal{A}}$ and $\mathcal{KP}_{\mathcal{B}}$) back to their canonical pose. The keypoints are said to be coherent if they overlap in this common reference system and if their indexes exactly match. To introduce these desiderata, we compute the consistency loss ($\mathcal{L}_{consist}$) between the corresponding keypoints in both the transformed sets as:

$$\mathcal{L}_{consist} = \frac{1}{K}\sum_{i=1}^{K} \left\| R_a^{-1}k_i^a - R_b^{-1}k_i^b \right\|_2^2. \tag{6}$$

In this way, we penalize keypoints with the wrong ordering and 3D position errors.

Pose loss: Our approach also learns to solve an auxiliary and self-supervised keypoints registration task by estimating the rotation matrix that aligns the two sets of keypoints against the (known) rotations in the input PCDs. Suppose R_{est} is the relative pose between $\mathcal{KP}_{\mathcal{A}}$ and $\mathcal{KP}_{\mathcal{B}}$, computed by using orthogonal Procrustes Analysis. Then the pose loss ($\mathcal{L}_{pose}$) can be computed using the Frobenius norm between the R_{est} and relative pose of the PCDs ($R_{ba} = R_a \cdot R_b^T$) as:

$$\mathcal{L}_{pose} = 2\ arcsin\left(\frac{1}{2\sqrt{2}}\ ||R_{est} - R_{ba}||_F\right). \tag{7}$$

It can be observed that if the keypoints in the canonical pose are not aligned/overlapped, the R_{est} will be erroneous, and hence the loss will be high. In other words, the lower pose loss validates the accuracy of the correspondences in both sets of keypoints.

The mutual dependency loss can be defined as the weighted sum of the above two losses:

$$\mathcal{L}_{mutual\_dependency} = w_{con} \cdot \mathcal{L}_{consist} + w_{pose} \cdot \mathcal{L}_{pos}, \tag{8}$$

where $\{w_{con},\ w_{pose}\}$ are defined as $\{1,\ 0.05\}$. The overall training loss is the sum of the position and the mutual dependency loss;

$$\mathcal{L}_{overall} = \mathcal{L}_{individual} + \mathcal{L}_{mutual\_dependency}. \tag{9}$$

3.5. Implementation details

During inference, the proposed approach takes a single PCD as input and estimates a semantically ordered list of K keypoints. The rotation of the input PCD can be arbitrary and we do not need any pre-processing step. The network is implemented in PyTorch and trained using the Adam optimizer with the learning rate $1e^{-3}$. We do not freeze any part of the network. In all the experiments, the batch size is set to 32 and trained on a 12GB GPU. We train [4], [23] and our network for 200 epochs and evaluate them using the best-trained model (with the minimum validation loss).

4. Experiments and evaluation

This section presents the dataset, the evaluation metrics, a comparison between our method SC3K and the SOTA approaches, and ablation studies.

4.1. Dataset

We use KeypointNet dataset [36] in our experiments, considering that this is the standard and most recent dataset used for keypoints estimation. It contains 8329 objects and 83231 keypoints of 16 object categories. We do not use the ground truth keypoints. Whereas, we rotate every object in 24 random poses since during training we need to feed two rotated versions of the same object to the proposed SC3K. We use the same rotation matrices that are used in ONet [16] with a validation and testing split that differs from the training set. For a fair comparison, we use the original (not-rotated) dataset to evaluate SC3K and the SOTA approaches.

4.2. Metrics for unsupervised keypoints estimation

To compare the performance of the proposed approach, we use three different standard metrics. The first metric, **inclusivity metric** [4] computes the percentage of the keypoints ($\mathcal{KP}$), which are estimated close to the $\mathcal{PCD}$. The keypoint (k_i) whose distance (d_i) to the nearest neighbour point in $\mathcal{PCD}$ is below the predefined threshold (τ_2) is considered as a close keypoint. The metric is defined as:

$$\begin{aligned} d_i &= \left\| k_i - kNN(k_i, \mathcal{PCD}) \right\|_2 \\ Inclusivity &= 100 \times \frac{1}{K}\sum_{i=1}^{K} [d_i < \tau_2], \end{aligned} \tag{10}$$

where [.] is the Iverson bracket (as described in Eq. 4). Although the inclusivity loss computes how close the $\mathcal{KP}$ are estimated from the input $\mathcal{PCD}$, it does not evaluate the accuracy of the keypoints in covering the whole object. Therefore, evaluation is further supported by the second metric, **coverage metric** [4], which compares the intersection over union of the 3D bounding boxes containing the $\mathcal{KP}$ with that of the $\mathcal{PCD}$. The

metric is defined as:

$$Cov = 100 \times \left[1 - \frac{|vol(\mathcal{PCD}) - vol(\mathcal{KP})|}{vol(\mathcal{PCD})}\right]$$
$$Coverage = \begin{cases} Cov & \text{if } vol(\mathcal{KP}) \leq 2 \times vol(\mathcal{PCD}) \\ 0 & \text{otherwise,} \end{cases} \quad (11)$$

where $vol(.)$ is the function that accepts a set of points ($\mathcal{KP}$ or $\mathcal{PCD}$), identifies a maximum and a minimum point from the accepted set, and returns their difference (i.e., the diagonal distance of the object's bounding box). The coverage will be 100% if both bounding boxes fully overlap and it will decrease if the bounding box of the $\mathcal{KP}$ is either smaller or greater than the one of $\mathcal{PCD}$. The third metric, **Dual Alignment Score (DAS)** evaluates the semantic consistency between the keypoints estimated for different objects of the same category. By following the same procedure as given in [23], we define the ratio of a set of reference keypoints for each category that are semantically aligned w.r.t. the corresponding human annotated keypoints.

4.3. Results and analysis

We compare SC3K with the SOTA approaches ULCS [4] and SM [23] that estimate the 3D keypoints in an unsupervised way. We trained and tested them using KeypointNet [36] dataset, keeping the PCDs in the canonical pose because they do not deal with the random rotation. However, considering the nature of SC3K, we train it for rotated PCDs (i.e., comparatively a more complex problem). We test SC3K under two conditions: *SC3K_rot* (PCDs with the random rotation) and *SC3K_can* (PCDs in the canonical pose). The random rotations are used to evaluate the accuracy of SC3K irrespective of the object's pose. However, to be consistent with our competitors (ULCS and SM), we also test our method for the original PCDs in a canonical pose. Tab. 1 presents a comparison among ULCS, SM and SC3K based on the three performance metrics as discussed in Sec. 4.2. Higher values correspond to better performance for every metric. The first inclusivity metric shows that, on average, the proposed approach (*SC3K_rot*) outperforms the SOTA approaches by estimating the keypoints close to the object's surface. However, *SC3K_can* achieves results better than those of ULCS and comparable to those of SM. The metric depends on the total number of keypoints and the tolerance threshold τ_2. We show in the supplementary material that inclusivity is high for fewer keypoints and it increases with the increase in the τ_2. We select τ_2 as 0.05 and consider 10 keypoints for all the experiments and comparison. The second coverage metric shows that on average the proposed approach is successful in estimating the keypoints whose 3D bounding boxes best overlap those of the input PCDs. For all the categories, *Ours_can* achieves better results. The third DAS metric validates that on average SC3K estimates semantically consistent keypoints. The DAS for the ULCS and SM are the same as reported in [37]. A detailed table presenting the comparison with the MR [37] and ISS [43] is given in the supplementary material.

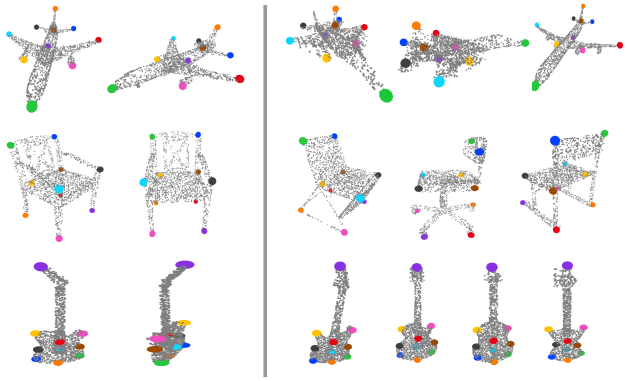


Figure 3: Shape pose variations vs. semantic correspondence: columns 1,2) keypoints estimated for two rotated versions of the same object are pose coherent; columns 3-5) keypoints semantically correspond to intra-class variations, in the same they also correspond to those estimated for different objects of the same category.

Unlike the existing approaches, we also evaluate the coherence property of the keypoints by computing the Matching Error (ME). This error is a localization error of the keypoints given PCD perturbations. We first estimate keypoints for different rotated versions of the same object PCDs and transform them to the canonical pose using the known rotations. Since the estimated keypoints are in the correct order, we compute order-wise position error between the corresponding keypoints on the canonical reference frame. A low error would indicate that 2D projection of a keypoint is rather unaffected by variations of the PCD. We repeat this procedure for all the instances of a category and calculate the ME in terms of mean error (μ) and the standard deviation (σ). The quantitative results are depicted in Tab. 2. The qualitative results of this experiment are illustrated in Fig. 3. Where, columns 1 and 2 (on the left side) show the keypoints estimated for two transformed versions of the same objects. It can be seen that the corresponding keypoints are semantically consistent irrespective of the object's pose, this validates the keypoints are coherent. The keypoints on the right side of the same Fig. 3 (columns 3, 4 and 5) illustrate the keypoints estimated for different rotated objects of the same category. It can be observed that

Category	Inclusivity ↑				Coverage ↑				DAS ↑			
	ULCS	**SM**	**SC3K**_*can*	**SC3K**_*rot*	**ULCS**	**SM**	**SC3K**_*can*	**SC3K**_*rot*	**ULCS**	**SM**	**SC3K**_*can*	**SC3K**_*rot*
Airplane	71.02	72.05	**87.20**	74.30	88.63	92.59	**96.34**	94.37	61.40	77.70	**82.86**	81.32
Bed	67.00	71.89	**80.00**	72.29	94.17	84.28	**98.20**	92.85	–	–	**64.87**	55.97
Bottle	75.44	72.84	77.36	**84.01**	80.93	91.44	**97.95**	94.16	–	–	**62.73**	57.22
Cap	57.50	59.50	56.25	**67.14**	60.83	85.01	**94.64**	91.81	–	53.00	**59.72**	58.10
Car	71.32	71.95	**76.05**	74.45	83.69	**90.69**	89.84	90.19	–	**79.40**	75.19	73.81
Chair	68.54	69.67	56.65	**72.33**	83.92	85.87	**95.31**	90.22	64.30	76.80	**87.04**	86.20
Guitar	50.14	69.29	**96.47**	69.04	79.83	85.65	**97.64**	92.17	–	63.10	**65.67**	64.02
Helmet	64.10	72.41	55.00	**74.68**	79.87	82.09	**90.50**	90.44	–	–	**58.55**	52.32
Knife	52.05	92.03	**98.33**	93.15	76.84	77.39	**98.77**	88.77	–	–	**62.98**	59.69
Motorbike	78.43	**95.28**	85.00	87.74	78.87	86.12	**94.34**	91.33	–	–	**59.41**	54.63
Mug	47.42	65.87	46.25	**82.37**	89.63	83.15	**95.15**	91.22	–	67.20	**75.25**	72.14
Table	60.06	79.13	**79.15**	73.05	82.97	91.31	**97.40**	92.32	–	70.00	**76.03**	71.62
Vessel	76.89	94.24	92.90	**95.24**	78.79	85.28	**97.18**	90.03	–	–	**75.95**	72.19
Average	64.61	75.86	75.89	**78.44**	81.46	86.22	**95.63**	91.53	62.85	69.60	**69.71**	66.09

Table 1: Comparison with the SOTA approaches (ULCS [4] and SM [23]) based on KeypointNet dataset. We test our approach for PCDs in canonical pose (*SC3K_can*) and the PCDs rotated in random poses (*SC3K_rot*). The results are calculated for 10 keypoints and the τ_2 for the inclusivity is selected as 0.1. The DAS of ULCS and SM are the same as reported in [37], thus we consider only the category available in [37]. For all the metrics, higher values are best. Bold and underlined numbers represent the first and second best performance, respectively.

ME	Airplane	Bed	Bottle	Cap	Car	Chair	Guitar	Helmet	Knife	Bike	Mug	Table	Vessel	Mean
μ	0.041	0.072	0.058	0.057	0.061	0.045	0.047	0.071	0.055	0.072	0.039	0.072	0.040	0.056
σ	0.019	0.057	0.056	0.038	0.042	0.021	0.020	0.052	0.034	0.040	0.023	0.051	0.031	0.037

Table 2: Pose coherent test: The keypoints estimated for randomly rotated versions of the same object are first transformed to the canonical pose. Then ME (μ and σ) is computed between the corresponding keypoints.

the keypoints also maintain the correspondences across the different intra-class variations of the object class.

5. Ablation studies

This section presents three ablations on: *i)* Choice of individual loss, *ii)* evaluation and performance of the network with combinations of the different training losses; *iii)* effect of varying noise ratio and decimations of the PCDs. Please refer to the supplementary material for additional related ablations.

5.1. Chamfer Distance (CD) vs. individual losses

Most of the existing approaches (including our competitors [4, 23]) have used a variant of the CD to train their networks; instead, we use individual losses. It is because the individual loss is more controllable, i.e., we can regulate every component by setting specific weight. This can be validated from the results presented in Tab. 1, i.e., SC3K outperforms the [4, 23]. Furthermore, we also present an ablation to highlight the significance of our selection. We train our network by replacing the individual losses with the CD loss. We observed that the keypoints are not estimated on the object's surface, and some are close to each other. Also, the inclusivity and coverage of the model trained with the individual loss are **+5.02** and **+19.83** better than the model trained with CD, respectively.

5.2. Performance for the selected losses

In order to highlight the significance of every loss, we train and evaluate the proposed approach by ignoring each loss one by one. The results are illustrated in Tab. 3. The conditional formatting green-to-red shows high-to-low values. It can be observed that approach performs well overall when all the loss functions are used. The overlap loss contributes comparatively low and is required only at the beginning of the training when the keypoints are estimated randomly. The contribution of the separation loss is comparatively higher than the overlap, shape and volume loss since it maintains the distance between the estimated keypoints, thus enforcing the keypoints to move over the whole object and toward the surface. Shape loss avoids the estimation of the keypoints outside the object. The contribution of the volume loss is comparatively lower than the other loss functions. The consistency and pose losses allow the estimation of the corresponding and pose coherent keypoints. Ignoring both the losses affects the overall performance of the proposed approach. The qualitative

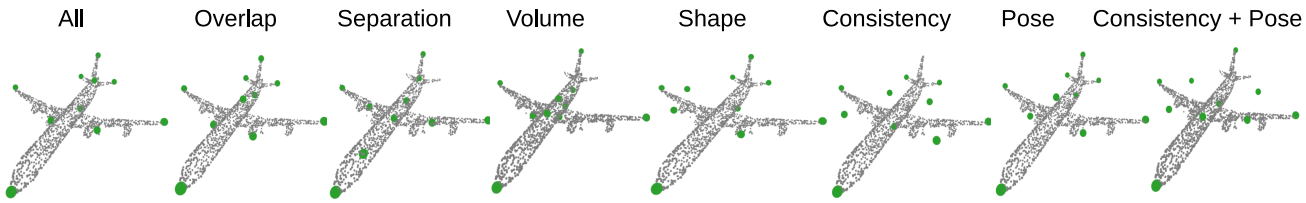


Figure 4: Performance of our approach with different combinations of losses. The leftmost figure shows the keypoints when the network is trained for all the losses. In the remaining figures, the model is trained without a specific loss which is mentioned at the top of every figure.

w.o. loss	Inc. ↑	Cov. ↑	DAS ↑	ME ↓
All loss	78.44	91.53	74.00	0.056
$\mathcal{A}_{overlap}$	77.09	90.72	53.80	0.061
$\mathcal{L}_{sep}$	63.01	85.70	67.38	0.081
$\mathcal{L}_{shape}$	76.05	90.31	58.45	0.064
$\mathcal{L}_{volume}$	77.35	90.90	63.14	0.066
$\mathcal{L}_{consist}$	76.52	91.03	42.44	0.103
$\mathcal{L}_{pos}$	76.76	91.04	53.89	0.095
$\mathcal{L}_{consist} + \mathcal{L}_{pos}$	70.15	88.07	41.95	0.103

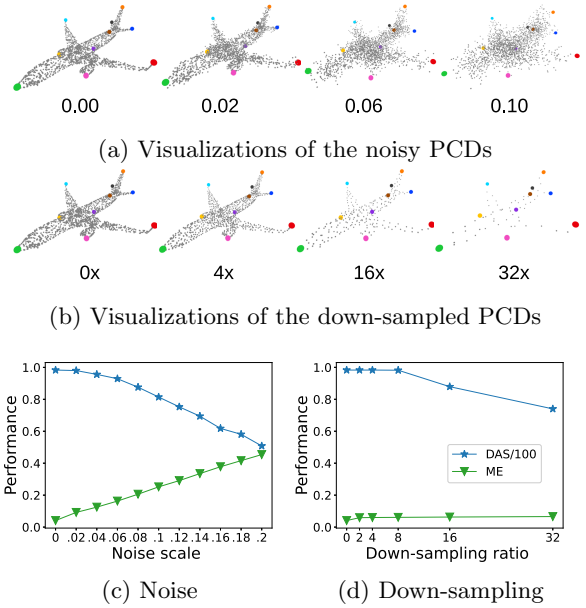
Table 3: Performance of the proposed approach for selected losses where, Inc., Cov., and ME represent inclusivity, coverage and matching error (coherence). The conditional formatting "green-to-red" represents the "good-to-bad" performance. The results are the average values of the test set of the keypointNet dataset.

results of the proposed approach trained without the selected loss function are illustrated in Fig. 4.

5.3. Robustness to perturbations

This ablation highlights the performance of the proposed approach for noisy and down-sampled PCDs of the airplane category. Noisy PCDs are generated by adding Gaussian noise of different variances to the original PCDs. For decimating the PCD, we use the Farthest Point Sampling (FPS) as used in [17, 38]. Fig. 5a and 5b show the keypoints estimated for noisy and down-sampled PCDs, respectively. Our approach successfully estimates the consistent keypoints at accurate positions for the noisy and down-sampled PCDs.

Quantitative results for the noisy and down-sampled PCDs are illustrated in Fig. 5c and Fig. 5d, respectively where to fix the DAS in the plots [0 to 1], we show DAS/100. The results show that the ME increases and the DAS decrease with the increase in the noise level. Similarly, DAS decreases if down-sampling ratio is reduced to 6 times the original PCD. The ME remains approximately the same for down-sampled PCDs, validating that down-sampling does not affect the keypoints.



(a) Visualizations of the noisy PCDs

(b) Visualizations of the down-sampled PCDs

(c) Noise

(d) Down-sampling

Figure 5: Performance of the proposed approach for noisy and decimated PCDs. (a) and (b) represent qualitative results, whereas, (c) and (d) illustrates the effect of the noise scale and down-sampling ratio, respectively.

6. Conclusions

This paper presented a method, SC3K, to estimate 3D keypoints from a single PCD such that they express the following properties: *robust* – minimum position error across different rotated versions of the same PCD; *compact* – proximal to the PCD surface, and *coherent* – in semantic order for all the intra-class instances. Similarly, the proposed method is *repeatable* – can estimate the accurate keypoints irrespective of the noise, down-sampling or rotation of the PCD; and *self-supervised* – can estimate the same keypoints from single PCD without requiring any labels (pseudo or human annotation) during the inference. We achieved these desiderata by

training the network with a new self-supervised strategy that does not require human annotations, instead, it computes the relative pose between the two sets of keypoints as a proxy task and then minimizes the error against the known relative pose of the input PCDs pair. The proposed approach is compared with the SOTA keypoints estimation approaches using the KeypointNet dataset. The results validated that the proposed SC3K outperforms the SOTA approaches by estimating the coherent keypoints close to the object's surface, characterizing the object's shape.

Limitations. SC3K may fail to estimate keypoints close to the object's surface for a number of keypoints higher than 35 and its performance decreases for symmetrical shapes. For some categories, such as bikes or cars, it is challenging to differentiate between the front and back wheels. In the same way, as it happens in previous approaches, strong intra-class geometrical variations negatively affect the performance, i.e., it is hard to compute semantically coherent keypoints between a single and a bunk bed. SC3K uses the publicly available dataset and estimates the keypoints to represent an object's shape. So, it does have very limited negative societal impacts.

Acknowledgements: We would like to acknowledge Milind Gajanan Padalkar, Matteo Taiana and Pietro Morerio for fruitful discussions, and Seyed Saber Mohammadi and Maryam Saleem for their support during the experimental phase. This work has been supported by the projects "RAISE-Robotics and AI for Socio-economic Empowerment" and "European Union-NextGenerationEU".

References

[1] Hui Chen, Dongge Sun, Wanquan Liu, Hongyan Wu, Man Liang, and Peter Xiaoping Liu. A novel approach to the extraction of key points from 3d rigid point cloud using 2d images transformation. *IEEE Transactions on Geoscience and Remote Sensing*, 2022. 2

[2] Nenglun Chen, Lingjie Liu, Zhiming Cui, Runnan Chen, Duygu Ceylan, Changhe Tu, and Wenping Wang. Unsupervised learning of intrinsic structural representation points. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9121–9130, 2020. 1, 2

[3] An-Chieh Cheng, Xueting Li, Min Sun, Ming-Hsuan Yang, and Sifei Liu. Learning 3d dense correspondence via canonical point autoencoder. *Advances in Neural Information Processing Systems*, 34:6608–6620, 2021. 2

[4] Clara Fernandez-Labrador, Ajad Chhatkuli, Danda Pani Paudel, Jose J Guerrero, Cédric Demonceaux, and Luc Van Gool. Unsupervised learning of category-specific symmetric 3d keypoints from point sets. In *European Conference on Computer Vision*, pages 546–563. Springer, 2020. 2, 4, 5, 6, 7

[5] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 4

[6] Yisheng He, Wei Sun, Haibin Huang, Jianran Liu, Haoqiang Fan, and Jian Sun. Pvn3d: A deep point-wise 3d keypoints voting network for 6dof pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11632–11641, 2020. 2

[7] Wei-Chih Hung, Varun Jampani, Sifei Liu, Pavlo Molchanov, Ming-Hsuan Yang, and Jan Kautz. Scops: Self-supervised co-part segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 869–878, 2019. 2

[8] Umar Iqbal, Kevin Xie, Yunrong Guo, Jan Kautz, and Pavlo Molchanov. Kama: 3d keypoint aware body mesh articulation. In *2021 International Conference on 3D Vision (3DV)*, pages 689–699. IEEE, 2021. 1

[9] Tomas Jakab, Richard Tucker, Ameesh Makadia, Jiajun Wu, Noah Snavely, and Angjoo Kanazawa. Keypointdeformer: Unsupervised 3d keypoint discovery for shape control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12783–12792, 2021. 1, 2

[10] Jiaxin Li and Gim Hee Lee. Usip: Unsupervised stable interest point detection from 3d point clouds. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 361–370, 2019. 1, 2

[11] Shifeng Lin, Zunran Wang, Yonggen Ling, Yidan Tao, and Chenguang Yang. E2ek: End-to-end regression network based on keypoint for 6d pose estimation. *IEEE Robotics and Automation Letters*, 7(3):6526–6533, 2022. 2

[12] Yiqun Lin, Lichang Chen, Haibin Huang, Chongyang Ma, Xiaoguang Han, and Shuguang Cui. Task-aware sampling layer for point-wise analysis. *IEEE Transactions on Visualization and Computer Graphics*, 2022. 2

[13] Xingyu Liu, Rico Jonschkowski, Anelia Angelova, and Kurt Konolige. Keypose: Multi-view 3d labeling and keypoint estimation for transparent objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11602–11610, 2020. 1

[14] Dominik Lorenz, Leonard Bereska, Timo Milbich, and Bjorn Ommer. Unsupervised part-based disentangling of object shape and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10955–10964, 2019. 2

[15] Guofeng Mei, Litao Yu, Qiang Wu, Jian Zhang, and Mohammed Bennamoun. Unsupervised learning on 3d point clouds by clustering and contrasting. *arXiv preprint arXiv:2202.02543*, 2022. 2

[16] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space.

In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019. 5

[17] Seyed Saber Mohammadi, Yiming Wang, and Alessio Del Bue. Pointview-gcn: 3d shape classification with multi-view point clouds. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 3103–3107. IEEE, 2021. 8

[18] Omid Poursaeed, Tianxing Jiang, Han Qiao, Nayun Xu, and Vladimir G Kim. Self-supervised learning of point clouds via orientation estimation. In *2020 International Conference on 3D Vision (3DV)*, pages 1018–1028. IEEE, 2020. 2

[19] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 3

[20] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 2

[21] Caner Sahin. Cmd-net: Self-supervised category-level 3d shape denoising through canonicalization. *Applied Sciences*, 12(20):10474, 2022. 2

[22] Rahul Sajnani, Adrien Poulenard, Jivitesh Jain, Radhika Dua, Leonidas J Guibas, and Srinath Sridhar. Condor: Self-supervised canonicalization of 3d pose for partial shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16969–16979, 2022. 2

[23] Ruoxi Shi, Zhengrong Xue, Yang You, and Cewu Lu. Skeleton merger: an unsupervised aligned keypoint detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 43–52, 2021. 1, 2, 3, 5, 6, 7

[24] Ivan Sipiran and Benjamin Bustos. Harris 3d: a robust extension of the harris operator for interest point detection on 3d meshes. *The Visual Computer*, 27(11):963–976, 2011. 2

[25] Weiwei Sun, Andrea Tagliasacchi, Boyang Deng, Sara Sabour, Soroosh Yazdani, Geoffrey E Hinton, and Kwang Moo Yi. Canonical capsules: Self-supervised capsules in canonical pose. *Advances in Neural Information Processing Systems*, 34:24993–25005, 2021. 2

[26] Junshu Tang, Zhijun Gong, Ran Yi, Yuan Xie, and Lizhuang Ma. Lake-net: Topology-aware point cloud completion by localizing aligned keypoints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1726–1735, 2022. 3

[27] Junshu Tang, Jiachen Xu, Jingyu Gong, Haichuan Song, Yuan Xie, and Lizhuang Ma. Prototype-aware heterogeneous task for point cloud completion. *arXiv preprint arXiv:2209.01733*, 2022. 2

[28] Rongxiao Tang, Luyang Wang, and Zhenhua Guo. A multi-task neural network for action recognition with 3d key-points. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 3899–3906. IEEE, 2021. 1

[29] Suwajanakorn *et al.* Discovery of latent 3d keypoints via end-to-end geometric reasoning. *NeurIPS*, 2018. 3

[30] Hanyu Wang, Jianwei Guo, Dong-Ming Yan, Weize Quan, and Xiaopeng Zhang. Learning 3d keypoint descriptors for non-rigid shape matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018. 1

[31] Yue Wang and Justin M Solomon. Prnet: Self-supervised learning for partial-to-partial registration. *Advances in neural information processing systems*, 32, 2019. 2

[32] Guangshun Wei, Long Ma, Chen Wang, Christian Desrosiers, and Yuanfeng Zhou. Multi-task joint learning of 3d keypoint saliency and correspondence estimation. *Computer-Aided Design*, 141:103105, 2021. 1, 2

[33] Han Xue, Liu Liu, Wenqiang Xu, Haoyuan Fu, and Cewu Lu. Omad: Object model with articulated deformations for pose estimation and retrieval. *arXiv preprint arXiv:2112.07334*, 2021. 2

[34] Zhengrong Xue, Zhecheng Yuan, Jiashun Wang, Xueqian Wang, Yang Gao, and Huazhe Xu. Useek: Unsupervised se(3)-equivariant 3d keypoints for generalizable manipulation. 2023. 2

[35] Yang You, Wenhai Liu, Yanjie Ze, Yong-Lu Li, Weiming Wang, and Cewu Lu. Ukpgan: A general self-supervised keypoint detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17042–17051, 2022. 1, 2

[36] Yang You, Yujing Lou, Chengkun Li, Zhoujun Cheng, Liangwei Li, Lizhuang Ma, Cewu Lu, and Weiming Wang. Keypointnet: A large-scale 3d keypoint dataset aggregated from numerous human annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13647–13656, 2020. 1, 2, 5, 6

[37] Haocheng Yuan, Chen Zhao, Shichao Fan, Jiaxi Jiang, and Jiaqi Yang. Unsupervised learning of 3d semantic keypoints with mutual reconstruction. *arXiv preprint arXiv:2203.10212*, 2022. 2, 6, 7

[38] Wentao Yuan, Tejas Khot, David Held, Christoph Mertz, and Martial Hebert. Pcn: Point completion network. In *International Conference on 3D Vision*, pages 728–737. IEEE, 2018. 8

[39] Yijun Yuan, Dorit Borrmann, Jiawei Hou, Yuexin Ma, Andreas Nüchter, and Sören Schwertfeger. Self-supervised point set local descriptors for point cloud registration. *Sensors*, 21(2):486, 2021. 2

[40] Peng Zhang, Ruoyin Xie, Jinsheng Sun, Weiqing Li, and Zhiyong Su. Au-pd: An arbitrary-size and uniform downsampling framework for point clouds. *arXiv preprint arXiv:2211.01110*, 2022. 2

[41] Yongheng Zhao, Guangchi Fang, Yulan Guo, Leonidas Guibas, Federico Tombari, and Tolga Birdal. 3dpointcaps++: Learning 3d representations with capsule

networks. *International Journal of Computer Vision*, 130(9):2321–2336, 2022. 2

[42] Chengliang Zhong, Peixing You, Xiaoxue Chen, Hao Zhao, Fuchun Sun, Guyue Zhou, Xiaodong Mu, Chuang Gan, and Wenbing Huang. Snake: Shape-aware neural 3d keypoint field. *arXiv preprint arXiv:2206.01724*, 2022. 1

[43] Yu Zhong. Intrinsic shape signatures: A shape descriptor for 3d object recognition. In *2009 IEEE 12th international conference on computer vision workshops, ICCV Workshops*, pages 689–696. IEEE, 2009. 6

[44] Mohammad Zohaib, Milind Gajanan Padalkar, Pietro Morerio, Matteo Taiana, and Alessio Del Bue. Cdhn: Cross-domain hallucination network for 3d keypoints estimation. *Available at SSRN 4349267*. 1

[45] Mohammad Zohaib, Matteo Taiana, Milind Gajanan Padalkar, and Alessio Del Bue. 3d key-points estimation from single-view rgb images. In *International Conference on Image Analysis and Processing*, pages 27–38. Springer, 2022. 1, 2