SelfGeo: Self-supervised and Geodesic-consistent Estimation of Keypoints on Deformable Shapes

Mohammad Zohaib[1], Luca Cosmo[2], and Alessio Del Bue[1]

[1] Pattern Analysis & Computer Vision, Italian Institute of Technology, Genoa, Italy
[2] Ca' Foscari University of Venice, Venice, Italy
zohaib.mohammad@hotmail.com, luca.cosmo@unive.it, alessio.delbue@iit.it

Abstract. Unsupervised 3D keypoints estimation from Point Cloud Data (PCD) is a complex task, even more challenging when an object shape is deforming. As keypoints should be semantically and geometrically consistent across all the 3D frames – each keypoint should be anchored to a specific part of the deforming shape irrespective of intrinsic and extrinsic motion. This paper presents, "SelfGeo", a self-supervised method that computes persistent 3D keypoints of non-rigid objects from arbitrary PCDs without the need of human annotations. The gist of SelfGeo is to estimate keypoints between frames that respect invariant properties of deforming bodies. Our main contribution is to enforce that keypoints deform along with the shape while keeping constant geodesic distances among them. This principle is then propagated to the design of a set of losses which minimization let emerge repeatable keypoints in specific semantic locations of the non-rigid shape. We show experimentally that the use of geodesic has a clear advantage in challenging dynamic scenes and with different classes of deforming shapes (humans and animals). Code and data are available at: https://github.com/IIT-PAVIS/SelfGeo

Keywords: Self-supervised · Deformable shapes · 3D keypoints

1 Introduction

Modelling and representing 3D deformable/non-rigid shapes is fundamental for enabling Augmented Reality, Virtual Reality and Human-Robot Interaction applications in realistic dynamic environments. In particular, the skeleton of a shape plays a vital role by providing information about the object's structure, poses and actions [7, 8, 27, 31, 42]. To generate a skeleton of a shape, the common practice is to estimate a set of keypoints [26] such that they are temporally consistent and highlight specific semantic parts of the object. However, estimating such keypoints for deformable shapes is a very challenging task due to their intrinsic deformations and intra-class variations.

Some existing approaches compute keypoints in a partially/fully supervised way, where high-quality ground truth annotations are requested [33, 40]. Crucially, labelling points on deformable shapes is a daunting, time and resource-consuming task that is also prone to human errors. Most current methods are

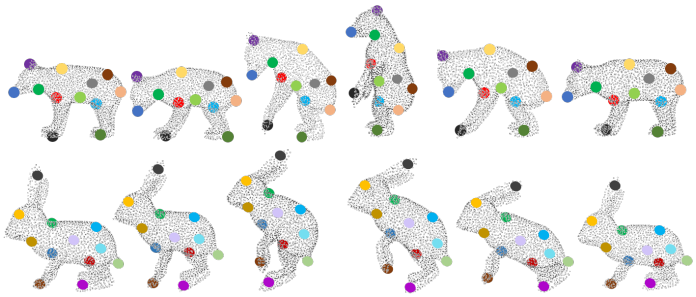


Fig. 1: Overview of the *SelfGeo*. The keypoints estimated for the two shapes of the same category are temporally consistent; they are anchored to their locations (equal geodesic distances) between two frames regardless of deformation, and maintain the semantic information (same colours indicate the corresponding keypoints). Moreover, the keypoints are estimated close to the surface and covering the whole shape.

limited to synthetic datasets where correspondences through frames are given by construction and no annotated keypoints currently exist for real datasets.

Recently, Weng et al. [33] estimated 3D keypoints for a synthetic human shape by using existing ground truth keypoints and later refining them with unsupervised losses to increase generalization to the real data. Unfortunately, due to the nature of the used loss functions, this approach is limited to human shapes. Differently, Fernandez et al. [8] use a linear shape basis for the ordered point correspondences. They learn to estimate consistent 3D keypoints by considering the objects' symmetry. Therefore, they perform very well for symmetric objects such as chairs, airplanes, etc. However, deforming motion may violate symmetry (i.e. having only one arm upright), thus drastically reducing the performance of the approach.

Recent literature dealing with 3D deformable shapes [5, 12, 19, 23, 24, 30, 36] has shown that injecting an isometric prior into Neural Networks architectures, either in the form of geometric losses or specifically designed components, is a good strategy to ease the learning task on the complex deformation patterns typical of deformable shapes.

Following this principle, we propose a novel self-supervised approach, named *SelfGeo*, to estimate stable keypoints that implicitly distil geometrical and temporal properties of non-rigid shapes. Unlike [8, 42], during training, we take a sequence of PCDs as input and estimate a set of keypoints for each input PCD. These keypoints should satisfy determined properties that we promote by a set of differentiable losses. The **Shape loss** acts on the distribution of keypoints on a single PCD, and it is mainly adapted from previous works on rigid keypoints estimation [8, 29, 44–46]. It enforces that the keypoints cover the volume of the object, adhere to its surface (i.e. compactness) and are informative enough to reconstruct the full PCD. Our main novelty is the introduction of a **Deformation loss**. It accounts for the topological structure of the deforming body by enforcing that the geodesic distances do not change among the keypoints estimated for all

the PCDs of the input sequence. To further impose temporal continuity during deformation, we also introduce a pairwise regularisation in consecutive frames. Our architecture shows that, at test time, keypoints emerge with the desirable properties we are seeking (Fig. 1): they are localised in semantically meaningful regions while covering the whole surface and they are anchored to their locations regardless non-rigid and rigid motion of the deformable shape.

To summarise, the main contributions of this work are as follows:

- We propose a self-supervised approach to estimate 3D keypoints from PCDs that does not require any a priori information about the deforming shape.
- To this end, we propose a novel differentiable loss function that preserves the geodesic distances between keypoints.
- We evaluate the CAPE and Deforming Things 4D dataset showing that the proposed approach is general and can be used for any deformable shapes irrespective of body structure or skeleton.
- The evaluation on the real ITOP dataset and the ablation studies show that the proposed approach is robust to noisy or decimated PCDs.

The paper is organized as follows. Section 2 reports the existing methods, Section 3 proposes the approach, Section 4 presents the experiments and discussions on the results, and finally, the conclusions are given in Section 5.

2 Related work

3D keypoints are estimated from images or point clouds for different downstream tasks such as finding distinct locations on objects [3, 9, 25], object deformation [13, 32, 35], generalizable manipulation [34], shape matching [1, 2, 14], clothes perception (folding, laundry, and dressing) [43], etc. Based on the object's geometry, we cluster existing methods into two sections: rigid or non-rigid.

2.1 Keypoints estimation for rigid objects

Suwajanakorn et al. [29] present an approach 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. A similar approach is proposed by Zohaib et al., in [46] that estimates 3D keypoints directly from images. During the training, they utilize the 3D ground truths to localize the kepoints. The presented results validate that their approach overperforms Suwajanakorn's method [29] when estimating the object pose. The SC3K approach [44] estimates semantically coherent keypoints for rigid objects from point clouds in a self-supervised way. This approach is robust to the pose, down-sampling and noise in the input PCDs. A similar approach is presented by Li et al. [15] that first generates clusters from the input point clouds and then estimates a keypoint for every cluster. Their method requires two rotated versions of a PCD for learning the stable keypoints under arbitrary transformations. Xue et al. [34] presents a teacher-student structure to

discover SE(3)-equivariant keypoints estimation from point clouds. The teacher module is similar to Skeleton Merger [26], allowing keypoints estimation in the canonical pose. Whereas, the student module uses a SPRIN backbone [38] to estimate the same keypoints from an object in a random pose. Keypoints can also be used for shape completion tasks as presented by Tang et al. in [31]. Their approach, "LAKe-Net", first estimates aligned keypoints, and then uses them to generate a surface-skeleton that represents the object's topological information. The skeleton is later used in the shape reconstruction and completion. Another method is proposed by Yuan et al. in [39] that estimates keypoints from one object that are good for reconstructing any instance of the same category. During training, their method estimates two sets of 3D semantically consistent keypoints from two different objects of the same category and uses them for self/mutual reconstruction. Shi et al. [25] uses the keypoints in shape and pose estimation in robotics applications, e.g. autonomous driving. They estimate 2D/3D keypoints from the RGB/RGBD images and use the proposed graph-theoretic framework (ROBIN) to remove outliers. Finally, the resulting keypoints are used to obtain the pose and shape of an object.

The techniques adopted in [15, 26, 29, 44, 46] to localize keypoints w.r.t. to an object's shape are significant in the self-supervised settings. Albeit they have used them for rigid objects, we use similar ideas in designing one of our losses that helps our network to estimate keypoints covering the deformable object [29, 46] and adhere to the surface [26, 44].

2.2 Keypoints estimation for deformable objects

Estimating the keypoints for deformable objects, such as humans and animals, is challenging due to the irregular deformation/motion of the object's parts. Chen et al. in [4] estimate 3D keypoints to represent robot joints and to capture an object's motion. Their approach uses multi-view images in an unsupervised way to generate 3D keypoints in the form of 2D heat maps and depths $[u,v,d]$ for each image. These keypoints are aggregated using the camera parameters to obtain the global coordinates of each keypoint. Zhong et al. [41] presents a self-supervised approach (SNAKE) that jointly reconstructs the object's surface and estimates the 3D keypoints. The method estimates keypoint saliency for each continuous query point instead of a discrete input point cloud. Fernandez et al. [8], present an unsupervised 3D keypoints estimation method that assumes object's being symmetric. The network estimates N keypoints and applies nonmax-suppression for selecting the final keypoints. Their main focus is on rigid objects but the presented results show that their approach is also valid for the deformable human body. An approach UKPGAN is presented in [37] that estimates the keypoints and semantic embeddings by optimizing the reconstruction task. Similarly to [8], this method is also used for human bodies in different non-rigid deformation. Weng et al. [33] propose an approach (GC-KPL) that estimates 3D keypoints from the point clouds of the human body. It is first trained on a synthetic dataset [17] using the available ground truths to learn keypoints'

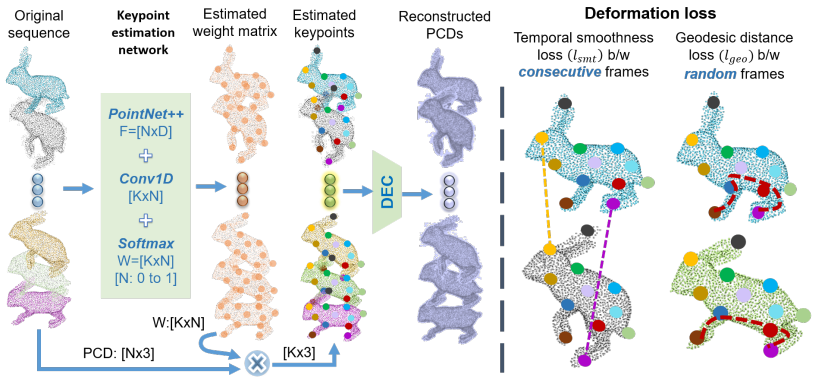


Fig. 2: Left: proposed *SelfGeo*. A sequence of PCD is input one by one to the Keypoints estimation network, which contains a PointNet++ encoder, a Conv1D and a Softmax layer. The network generates K values for each point, indicating its probability to be one of the K keypoints. The expected keypoint positions for each PCD are computed and passed to a decoder (DEC), which consists of 4 Con1D layers, to reconstruct the 3D shape. To improve the keypoints' inference, *SelfGeo* computes the shape loss (reconstruction, coverage and surface loss) using a single PCD, and deformation loss (right: geodesic distance and temporal smoothing loss) between two frames.

positions and semantic segmentation to localize the object's part. Then, the approach is refined on the Waymo Open Dataset [28] using an un-supervised loss functions. In contrast, Zhong et al. [42] estimate self-supervised 3D keypoints mainly for robotic manipulation tasks. Their approach estimates 3D keypoints for two versions of the same PCD of an articulated object in different poses. The keypoints are then used to compute the rotation axis between the two PCDs. The available rigid transformations are given as a prior for network optimization. The method also reconstructs the target shape from the source by transporting the features of the target keypoints to the features of the source shape.

The above methods are either supervised or are object-specific (i.e. skeleton-based and applicable only for humans) and they fail to produce consistent keypoints in case of skeleton-free deformation (i.e. for both humans and animals). In comparison, we propose *SelfGeo*, a self-supervised approach that estimates 3D keypoints for any deformable objects. This is achieved by designing a deformable loss that preserves both temporal and geometric consistency among the keypoints estimated for a sequence of frames.

3 Proposed approach

This section describes the design of the *SelfGeo*, the loss functions that are used to localize the keypoints, and the complete pipeline for training and testing.

Given a PCD $\mathcal{P} = [x_1, x_2, ..., x_N] \in \mathbb{R}^{3\times N}$ with N points, the goal of the proposed approach is to estimate a set of K keypoints $\mathcal{K} = \{k_1, k_2, ..., k_K\} \in \mathcal{P}$

located at specific regions, robust to the non-rigid deforming motion, and consistent among different subjects. To achieve this goal, we train a neural network to predict a set of K probability distributions $p_{k_j}(x_i)\ \forall\ k_j \in \mathcal{K}$, representing the probability of each $x_i \in \mathcal{P}$ to be the j^{th} keypoint. Then, the position of a keypoint k_j is computed as a convex combination of all $x_i \in \mathcal{P}$ as $\mathbb{E}_{\mathcal{P}}[k_j] = \sum_i x_i\ p_{k_j}(x_i)$.

3.1 Self-supervised keypoints estimation losses

We design *SelfGeo* to extract K keypoints which are informative, semantically consistent, well spread over the PCD and robust to non-rigid motion. To ensure these properties, we devise two types of self-supervised losses. The **shape loss** ($\mathcal{L}_{sha}$) is tasked to promote the extracted keypoints to be as much informative as possible, and ensures that they are well distributed over the PCD. This loss acts on each PCD individually and does not account for the deformation dynamics over time. The **deformation loss** ($\mathcal{L}_{def}$), on the other hand, takes care of the consistency of the keypoints given the non-rigid motion of the PCDs.

The overall training loss is then the sum of the localization and deformation losses:

$$\mathcal{L}_{tot} = \mathcal{L}_{sha} + \mathcal{L}_{def}. \tag{1}$$

Shape loss The shape loss is made up of three components: reconstruction loss, coverage loss, and surface loss. All these losses apply to a single PCD and they enforce a distribution of keypoints on the shape that respects different principles.

Reconstruction loss. Without having the access to any labeled data, it is not trivial to define what "good" keypoints are. Ideally, we would like our keypoints to capture as much information of the original shape as possible. To this end, we pair our keypoints estimation with a proxy PCD reconstruction task. The expected locations of all keypoints $\mathbb{E}_{\mathcal{P}}[k_j]$, are fed to a decoder (DEC) tasked to output a 3D PCD with M points $\tilde{\mathcal{P}} = \text{DEC}(\{\mathbb{E}_{\mathcal{P}}[k_j]\}_{j=1}^{K}) = [\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_M]$ minimizing a reconstruction loss in terms of Chamfer distance from the input point cloud $\mathcal{P}$:

$$\mathcal{L}_{rec} = \sum_{i=1}^{N} \min_j \left\|x_i - \tilde{x}_j\right\|_2^2 + \sum_{j=1}^{M} \min_i \left\|\tilde{x}_j - x_i\right\|_2^2. \tag{2}$$

Notice here that we are flexible in the number of input N and output M points.

Coverage loss. One problem of the reconstruction loss is that it tends to focus keypoints positions on regions that undergo major deformations, while leaving uncovered more "static" regions (e.g. the head in a human is unlikely to attract any keypoints). On the other hand, we would like the estimated keypoints to highlight different parts of an object. We promote this behaviour by penalizing keypoints for being too close to each other:

$$\mathcal{L}_{cov} = \left(\frac{1}{K}\sum_{i=1}^{K} \min_{j\neq i} \left\|\mathbb{E}_{\mathcal{P}}[k_i] - \mathbb{E}_{\mathcal{P}}[k_j]\right\|_2 + \epsilon\right)^{-1}, \tag{3}$$

where epsilon is a small constant (1e-2) to prevent numeric errors.

Surface loss. Being the expected location of a keypoint k_j expressed as a convex combination of all $x_i \in \mathcal{P}$, its position might be located far from the surface of the shape. This behaviour is further accentuated by $\mathcal{L}_{cov}$ whose goal is to push keypoints far apart in the Euclidean space. This would result in probability distributions of keypoints being spread over regions external to the input PCD. To address this issue, we introduce the surface loss ($\mathcal{L}_{surf}$), which restricts the keypoints to be estimated close to the PCD. This loss minimizes the expected distance of a keypoint k_i in $\mathcal{K}$ from the nearest point $x \in \mathcal{P}$ as:

$$\mathcal{L}_{surf} = \frac{1}{K} \sum_{i=1}^{K} \min_{j} \left\| \mathbb{E}_{\mathcal{P}}[k_i] - x_j \right\|_2. \tag{4}$$

The effect of this loss is to promote local support of the keypoints' probabilities.

The total shape loss can be summarized as a weighted sum of the above components as:

$$\mathcal{L}_{sha} = \lambda_{rec} \cdot \mathcal{L}_{rec} + \lambda_{cov} \cdot \mathcal{L}_{cov} + \lambda_{surf} \cdot \mathcal{L}_{surf}, \tag{5}$$

where $\{\lambda_{rec}, \lambda_{cov}, \lambda_{surf}\}$ are experimentally set to $\{1, 2.5, 6\}$, respectively.

Deformation loss The shape loss, albeit allowing the network to identify the most informative points in the PCD, does not impose any spatial consistency of the extracted keypoints between different non-rigid poses, resulting in the keypoints drifting along the surface and swapping in position. To prevent this effect, we devise a deformation loss which takes into account the behavior of the keypoints on a sequence of deforming PCDs.

At training time, we assume to have access to a sequence of T consecutive PCDs $[\mathcal{P}^1, \ldots, \mathcal{P}^T]$ of a non-rigid body under relative motion with points $\mathcal{P}^t = [x_1^t, x_2^t, ..., x_N^t]$. A desirable property of the estimated keypoints is that they should move relatively to the deformation field while keeping their position over the object surface, thus preserving their semantic position and order against the non-rigid motion. Considering this principle, we propose a deformation loss based on two components: geodesic and smoothing losses.

Geodesic loss. Differently from a rigid setting, where only global rigid transformations (i.e. roto-translations) are considered [15,44], in the non-rigid setting the individual parts forming a PCD may change their relative positions due to deformations [6,10]. This implies that, while Geodesic distances on the underlying surface are still (approx.) preserved, this is not true for the Euclidean ones.

Considering this, we present a novel differentiable loss function to preserve the expected geodesic distances between the estimated keypoints along all frames. Given the PCD $\mathcal{P}^t$ at frame t, the expected geodesic distance between two keypoints is defined as:

$$\mathbb{E}_{\mathcal{P}^t}[d_{\mathcal{P}^t}(k_i, k_j)] = \sum_{m,n} p_{k_i}(x_m^t) d_{\mathcal{P}_t}(x_m^t, x_n^t) p_{k_j}(x_n^t) \ ,$$

where $d_{\mathcal{P}^t}(\cdot,\cdot)$ is the pre-computed geodesic distance between two points in $\mathcal{P}^t$. If we stack all the keypoint probabilities row-wise in a matrix $W^t \in \mathbb{R}^{K\times N}$ and all the pairwise geodesic distances in a matrix $D^t \in \mathbb{R}^{N\times N}$ with elements $D^t(i,j) = d^t(x_i^t, x_j^t)$, the matrix containing the expected distance between all keypoint pairs can be written as $W^t D^t {W^t}^\top$, leading to the geodesic loss:

$$\mathcal{L}_{geo} = \sum_{\substack{a,b=1 \\ a\neq b}}^{T} \left\| W^a D^a {W^a}^\top - W^b D^b {W^b}^\top \right\|_F^2. \tag{6}$$

Note that the geodesic distance matrices are needed only during training, and can be created offline during the dataset preprocessing step.

Smoothing loss. Albeit being a strong prior on the keypoint location, the geodesic distance is not robust to the swap of keypoints due to symmetries on the shape. For instance, swapping the left and right arms of the shape might not bring a significant increase in the geodesic loss. To alleviate this problem, we propose to pair the geodesic loss with a temporal smoothing loss.

Under the reasonable assumption of smooth rigid and non-rigid motions, we expect a point of the PCD to not change significantly in its 3D position between two consecutive frames. On the other hand, flipping due to symmetries generally causes an abrupt change in such positions, and thus a larger error for the smoothing loss. We promote the temporal smoothness of the keypoint positions through the following loss:

$$\mathcal{L}_{smt} = \frac{1}{(T-1)K} \sum_{t=1}^{T-1} \sum_{j=1}^{K} \left\| \mathbb{E}_{\mathcal{P}^t}[k_j] - \mathbb{E}_{\mathcal{P}^{t+1}}[k_j] \right\|_2. \tag{7}$$

The overall deformation loss is the weighted sum of both its components as described below:

$$\mathcal{L}_{def} = \lambda_{geo} \cdot \mathcal{L}_{geo} + \lambda_{smt} \cdot \mathcal{L}_{smt}, \tag{8}$$

where $\{\lambda_{geo}$ and $\lambda_{smt}\}$ are hyperparameters experimentally set to $\{6, 2\}$ to equalize the contribution of both losses.

3.2 Network architecture

The architecture of the *SelfGeo* is illustrated in Fig. 2. The approach is trained end-to-end. It uses a PointNet++ [22] backbone to extract D-dimensional features for each point in the input PCD ($\mathcal{P}$). The features are then passed through a Conv1D and a softmax layer to produce the probability matrices W used to compute the geodesic loss. The keypoints are further fed to the reconstruction decoder to reconstruct the PCD ($\tilde{\mathcal{P}}$). The reconstruction decoder consists of four layers; three Conv1D layers followed by a batch normalization layer and a ReLU activation function, and one Conv1D that produces the reconstructed PCD $\tilde{\mathcal{P}}$.

3.3 Inference and implementation details

At inference time, the network takes as input just the raw 3D coordinates of a single PCD, without the need to pre-compute any geodesic distances, and it gives as output the expected position of keypoints. We subsampled each PCD to $N = 2048$ points and, if not specified differently w.r.t. a category, extract $K = 12$ keypoints, We implemented *SelfGeo* in PyTorch. The models were trained with Adam optimizer, batch size of 32 and learning rate of $1e$-3 on a 12GB GPU.

4 Experiments and evaluation

This section presents the dataset used in our experiments, performance evaluation metrics, and a comparison of *SelfGeo* with the existing methods.

4.1 Dataset

We use three datasets for evaluation: Clothed Auto Person Encoding (CAPE) [18], Invariant-Top View Dataset (ITOP) [11] and Deforming Things 4D [16]. The CAPE dataset contains synthetic human models. We considered 61 different temporal sequences divided into train, test, and validation sets as 40 (12432 frames), 11 (3311 frames), and 10 (2228 frames), respectively. The ITOP dataset contains depth videos captured from the real scene. The videos present human actions. We segment offline the human from the four *side-view* scenes (4626 frames) using the given labels. We observed that the humans are not accurately segmented in several frames, so challenging further the keypoints extraction methods in this real test. To validate the claim of generalization, we evaluate *SelfGeo* on the Deforming Things 4D dataset that contains videos of animals performing different actions. We use 9 animals (Bears, Bucks, Bull, Bunny, Chicken, Deer, Dog, Tiger, Rhino) for training. We split the animal sequences randomly into 70%, 10%, and 20%, for the training, validation, and testing sets, respectively.

We convert the meshes into PCDs of 2048 points normalized within the unit volume. For training, we precompute geodesic distances for each PCD by creating a graph connecting the five neighbours of each point and then approximating the geodesic distance as the shortest path distance between all pairs of points.

4.2 Metrics for unsupervised keypoints estimation

We use five metrics to evaluate results on self-supervised keypoints estimation. The *coverage* metric evaluates how well the keypoints are distributed over the surface and the *inclusivity* metric indicates how close the keypoints are to the object surface. These metrics have been proposed in previous literature [8, 44], and they are a standard measure to evaluate unsupervised keypoints estimation. The third metric is the *temporal consistency* (T_{con}), which shows if keypoints are switching their order in consecutive frames. We compute this metric as:

$$T_{con} = \frac{100}{(T-1)K} \sum_{t=1}^{T-1} \sum_{i=1}^{K} eq\left(\underset{j}{\operatorname{argmin}} \left(||\mathbb{E}_{\mathcal{P}^t}[k_i] - \mathbb{E}_{\mathcal{P}^{t+1}}[k_j]||_2\right), i\right), \quad (9)$$

where $eq(x, y)$ is 1 if $x = y$ and 0 otherwise. T_{con} represents the percentage of the semantically consistent keypoints.

The fourth metric is the Probability of Correct Keypoints (PCK), which defines keypoints repeatability across frames. Considering a keypoint k_i, we consider its expected position in the first frame $\mathbb{E}_{\mathcal{P}^1}[k_i]$ as a reference and compute its repeatability error in a subsequent frame t as:

$$E_{rep}(k_i, t) = \left\| GT_t(\mathbb{E}_{\mathcal{P}^1}[k_i]) - \mathbb{E}_{\mathcal{P}^t}[k_i] \right\|_2, \tag{10}$$

where $GT_t(\mathbb{E}_{\mathcal{P}^1}[k_i])$ is a function that computes the expected value of keypoint k_i on frame t by transferring its probability distribution over $\mathcal{P}^1$ to $\mathcal{P}^t$ using the ground-truth point-wise map. The PCK_τ measure is computed as the percentage of keypoints on all subsequent frames $t = 2 \dots T$ with an error smaller than τ.

Moreover, we consider a 3D reconstruction metric as a downstream task to assess the representation power of the keypoints in encoding the PCD of the deforming shape. For the evaluation of the reconstructed PCD w.r.t. the input PCD, we use the *Chamfer distance* defined in Eq. 2.

4.3 Results and analysis

We evaluate *SelfGeo* against the SOTA un-/self-supervised keypoints estimation approaches, ULCS [8] which estimates 3D keypoints from non-rigid objects under an arbitrary deformation (irrespective of body structure), as we do, and SC3K [44] which estimates keypoints from an arbitrary posed objects. We train all the methods on the CAPE dataset. When training of the ULCS and SC3K, we considered the hyperparameter as provided in the respective papers. For performance evaluation, we test them on synthetic PCDs of the CAPE dataset, and on real PCDs extracted from the depth acquisitions of the ITOP dataset. As mentioned earlier, ITOP is a particularly challenging dataset, since segmented body PCDs are often distorted and with missing parts.

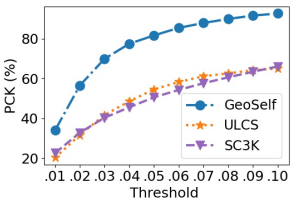
To show the ability of *SelfGeo* to handle different deformable objects, we compare it on the Deforming Thigs 4D dataset, which is composed of different animal shapes. The results reported in Table 1 highlight that *SelfGeo* outperforms the baselines for all the metrics. The keypoints estimated by *SelfGeo* are close to the body surface (high inclusivity), cover the whole body (high coverage), maintain the semantic order across the frames (high temporal consistency), and provide a reasonable shape reconstruction (lower reconstruction error).

The repeatability test on the Deforming Things 4D dataset is depicted in Fig. 3a, which shows the PCK curves computed by evaluating the PCK_τ measure at increasing threshold distances τ, ranging from 0.01 to 0.10. It can be seen that the *SelfGeo* outperforms ULCS and SC3K by a significant margin.

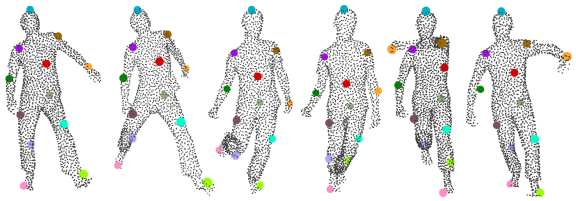
The qualitative results of the CAPE dataset are illustrated in Fig. 3b. The figure shows the keypoints estimated by the *SelfGeo* for a person playing soccer. The colours and positions of the keypoints represent their semantic and geometric information. It can be seen that the keypoints are stable throughout the shape motion. However, in some cases, we observed that due to a significant

Table 1: Comparison of *SelfGeo* with the baseline methods on CAPE, ITOP and Deforming Things 4D datasets. *SelfGeo* outperforms the baselines on all the datasets.

Dataset	Approach	Inclusivity ↑	Coverage ↑	T_{con} ↑	Recon. Err. ↓
CAPE [18]	ULCS	47.82	67.59	88.12	0.156
	SC3K	79.32	86.46	70.72	0.025
	SelfGeo	**85.87**	**91.87**	**90.43**	**0.012**
ITOP [11]	ULCS	42.08	64.28	76.47	0.221
	SC3K	76.35	83.11	42.86	0.213
	SelfGeo	**84.23**	**91.13**	**80.64**	**0.012**
Deforming Things 4D [16]	ULCS	48.06	82.91	41.75	0.279
	SC3K	81.7	84.874	77.44	0.153
	SelfGeo	**85.22**	**88.11**	**80.53**	**0.038**



(a) Repeatability test



(b) Keypoints estimated by *SelfGeo* on the CAPE dataset

Fig. 3: The higher PCK in (a) shows that the keypoints estimated by *SelfGeo* have better correspondences across frames, and (b) demonstrates their temporal consistency.

variation in the human pose, the keypoints do not preserve the semantic order. Still, even if the switch happens, a keypoint remains geometrically consistent with a position in the same region of the shape. This is likely due to the noise in computing the geodesic distances, which is discussed in supplementary material.

The qualitative results for the ITOP datasets are illustrated in Fig. 4, where the top row shows the original depth images containing a human in a real scene. We feed the segmented humans to *SelfGeo* that estimates the keypoints. The estimated keypoints on the input PCD are shown in the bottom row. The *SelfGeo* remains successful in estimating accurate keypoints from real sequences.

The qualitative results for the Deforming Things 4D dataset are illustrated in Fig. 5. The first, third and fifth rows show animals performing different actions (jump, run, attack, etc.). The second, fourth and sixth rows illustrate the corresponding estimated keypoints, respectively. The keypoints are temporally consistent even if the range of the motion is quite relevant, denoting that the geodesic loss helps to obtain consistent keypoints that can account for relevant deformations. Thus the results highlight that, on average, *SelfGeo* is robust enough to estimate stable keypoints irrespective of body deformation.

To further validate the robustness of the *SelfGeo*, we also evaluate its resilience to noisy and decimated PCDs of the CAPE dataset. To generate the

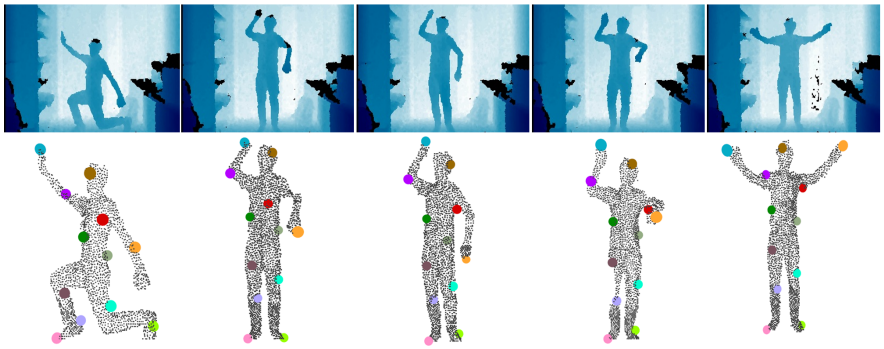


Fig. 4: Performance on real humans. The ITOP dataset contains depth images including the background (top row). We segment humans and pass them to the *SelfGeo*, which remains successful in estimating consistent keypoints as shown in the bottom row.

Table 2: Robustness to the PCD perturbations. The *SelfGeo* pretrained on the CAPE dataset is tested for the noisy and downsampled PCDs for different noise variances and sampling ratios. It has a stronger resilience to decimated PCDs and noisy PCDs.

	Inclusivity ↑	Coverage ↑	T_{con} ↑	Recon. Err. ↓	GD Err. ↓
Variance/Ratio	Nosiy/Downsampled PCDs				
0.01/x2	81.20/82.67	90.79/90.85	90.23/90.02	0.0135/0.0128	0.0449/0.012
0.02/x4	70.46/78.88	89.84/89.33	88.08/89.32	0.0171/0.0144	0.0531/0.025
0.03/x8	64.29/75.61	88.25/88.72	87.74/88.27	0.0213/0.0168	0.0586/0.047
0.04/x16	75.97/73.48	85.97/88.19	85.67/86.11	0.0249/0.0214	0.0623/0.069
0.05/x32	54.07/62.13	82.81/87.85	83.93/74.32	0.0285/0.0332	0.0711/0.089

noisy PCDs, we add Gaussian noise with different variances to the original PCDs. For decimating PCDs, we downsampled the original PCDs using the Farthest Point Sampling (FPS) as used in [20,44]. This experiment simulates the common issues when using different depth/3D sensors with different accuracies and data points densities. The results presented in Tab. 2 shows that the *SelfGeo* has a stronger resilience to decimated PCDs than the noisy case, as the x32 downsampling is extremely aggressive. This is because the downsampled PCDs maintain the object's structure. In contrast, introducing noise of a high variance (>0.02) changes the shape of the objects, thus reducing the performance. We observed that when the noise variance is 0.02, the SelfGeo has estimated keypoints that cover the object and hence are good for the reconstruction task (i.e. the error is 0.0144, which is only 0.0016 greater than that when the variance is 0.01).

5 Ablations

This section provides two main ablation experiments for *SelfGeo*. First, we train it by excluding one loss at a time to understand the impact of each loss on the

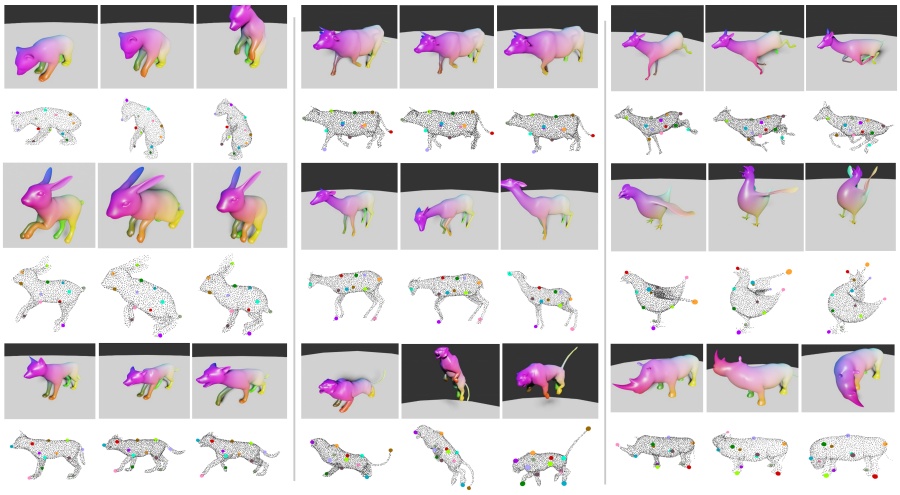


Fig. 5: Keypoints estimated on the Deforming Things 4D dataset. The first, third and fifth rows (from left to right) show animals performing different actions. Corresponding estimated keypoints on the input PCDs are illustrated in the second, fourth and sixth rows. The keypoints/PCDs are shown from the side view for a better visualization.

approach. Second, we train it for the PointNet [21] backbone.

Contribution of each loss. To test the contribution of each loss, we train *SelfGeo* on the mini-CAPE dataset selected from a part of the CAPE dataset. Mini-CAPE contains 405, 195, and 270 frames in each training, validation and testing set, respectively. At each training round, we ignore one loss and report results in Tab. 3. We do not show the test without the coverage ($\mathcal{L}_{cov}$) and shape ($\mathcal{L}_{sha}$) loss as their removal makes *SelfGeo* fail, denoting the cardinal importance of these elements. If we ignore surface loss ($\mathcal{L}_{surf}$), the inclusivity, coverage and T_{con} reduces by 8.66, 6.45 and 5.01, respectively. It is due to the fact that they are estimated at random positions, mostly outside the shape because of the coverage loss. Since they could not preserve the object's geometrical structure, they do not support the reconstruction task – reconstruction error is increased by 0.007. Ignoring the reconstruction loss ($\mathcal{L}_{rec}$) decreases the keypoints accuracy in terms of coverage (5.74) and inclusivity (5.91) – they are estimated in the surroundings of the shape. However, their consistency in the consecutive frames is slightly affected (0.72). This shows that this loss has less influence on the temporal consistency of the keypoints. Considering that the smoothing loss ($\mathcal{L}_{smt}$) allows maintaining the distance between the keypoints in the consecutive frames, its ignorance reduces the semantic correspondences between the keypoints (T_{con} is decreased by 2.11). In the same way, removing the geodesic distance loss ($\mathcal{L}_{geo}$) reduces also the keypoints localization in different frames (inclusivity and coverage are decreased by 3.74 and 4.14, respectively). While removing the deformation loss ($\mathcal{L}_{smt}$ and $\mathcal{L}_{geo}$) greatly reduces the performance

Table 3: Ablation on the loss functions (first five rows) and backbone (last row). We train *SelfGeo* by removing one loss at each training round on the mini-CAPE dataset. The increase or decrease in performance is given in the + or - sign, respectively. The colours also code the extent of the performance drop from green to red. The last row highlights that changing the backbone decreases the performance for all the metrics.

Removed Loss	Inclusivity ↑	Coverage ↑	T_{con} ↑	Recon. Err. ↓	GD Err. ↓
Surface	-8.66	-6.45	-5.01	+0.007	+2.8E-05
Reconstruction	-5.91	-5.74	-0.72	+0.230	+2.1E-05
Smoothing	-3.36	-2.72	-2.11	+0.001	+1.8E-04
Geodesic	-3.74	-4.14	-3.31	+0.002	+2.6E-03
Deformation	-3.99	-5.94	-3.82	+0.004	+2.1E-01
PointNet backbone	-1.31	-0.53	-1.95	+0.002	+1.75E-03

of the network. Not only the inclusivity and coverage are further decreased, but also the T_{con} is decreased with an increase in the reconstruction error.

Impact of the backbone. We provide a further ablation for two widely used backbones; PointNet and PointNet++. We observed that the performance of the *SelfGeo* is higher when PointNet++ backbone is used. The last row of Tab. 3 depicts the performance drop when PointNet encoder is integrated in *SelfGeo*.

6 Conclusions

In this paper, we introduce *SelfGeo*, a novel method to estimate the 3D keypoints on the deformable shapes in a self-supervised way without requiring the ground truth labels. Such keypoints should remain temporally consistent across the 3D frames, irrespective of the shape deformation. *SelfGeo* achieved this goal by considering the invariant properties of the deforming shapes – the keypoints should deform along with the non-rigid motion of the shape (semantically consistent), however, they should also maintain a constant geodesic distance among them (geometrically consistent). We evaluate our method against the SOTA approaches on three different deformable datasets. The results demonstrate the superiority of *GeoSlef* over the baselines. Morevoer, we also present the test on the noisy and decimated PCDs. It shows that the performance of the *SelfGeo* has slightly decreased; however, it remains successful in estimating the consistent keypoints.

Limitations of the proposed approach. Noise in the geodesics computation can affect our method. We have observed that when two body parts touch each other so that the 3D points are so close, they are considered connected (and they should not be). Thus, the geodesic distance between them approaches zero, which is an error. When this error happens, the keypoints estimated by *SelfGeo* might change their positions. See a practical example in the supplementary material. Moreover, symmetry in the object's parts, which is considered a basic problem in 3D vision, also reduces the network's performance.

Acknowledgements: We would like to acknowledge Pietro Morerio for fruitful discussions. This work was carried out within the frameworks of the project "RAISE - Robotics, and AI for Socio-economic Empowerment" and the PRIN 2022 project n. 2022AL45R2 (EYE-FI.AI, CUP H53D2300350-0001). This work has been supported by European Union - NextGenerationEU.

References

1. Attaiki, S., Li, L., Ovsjanikov, M.: Generalizable local feature pre-training for deformable shape analysis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13650–13661 (2023)
2. Attaiki, S., Ovsjanikov, M.: Ncp: Neural correspondence prior for effective unsupervised shape matching. Advances in Neural Information Processing Systems **35**, 28842–28857 (2022)
3. Bai, Y., Wang, A., Kortylewski, A., Yuille, A.: Coke: Contrastive learning for robust keypoint detection. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 65–74 (2023)
4. Chen, B., Abbeel, P., Pathak, D.: Unsupervised learning of visual 3d keypoints for control. In: International Conference on Machine Learning. pp. 1539–1549 (2021)
5. Cosmo, L., Minello, G., Bronstein, M., Rodolà, E., Rossi, L., Torsello, A.: 3d shape analysis through a quantum lens: the average mixing kernel signature. International Journal of Computer Vision **130**(6), 1474–1493 (2022)
6. Cosmo, L., Norelli, A., Halimi, O., Kimmel, R., Rodola, E.: Limp: Learning latent shape representations with metric preservation priors. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16. pp. 19–35. Springer (2020)
7. Dai, X., Li, S., Zhao, Q., Yang, H.: Animal pose estimation based on 3d priors. Applied Sciences **13**(3), 1466 (2023)
8. Fernandez-Labrador, C., Chhatkuli, A., Paudel, D.P., Guerrero, J.J., Demonceaux, C., Gool, L.V.: Unsupervised learning of category-specific symmetric 3d keypoints from point sets. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16. pp. 546–563. Springer (2020)
9. Gupta, A., Hoffmann, P.F., Prepelită, S., Robinson, P., Ithapu, V.K., Alon, D.L.: Learning to personalize equalization for high-fidelity spatial audio reproduction. In: ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 1–5. IEEE (2023)
10. Halimi, O., Litany, O., Rodola, E., Bronstein, A.M., Kimmel, R.: Unsupervised learning of dense shape correspondence. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4370–4379 (2019)
11. Haque, A., Peng, B., Luo, Z., Alahi, A., Yeung, S., Fei-Fei, L.: Towards viewpoint invariant 3d human pose estimation. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. pp. 160–177. Springer (2016)
12. Huang, K., Zhang, Y., Chen, J., Ma, F., Tan, Z., Xu, Z., Jiao, Z.: Skeleton-based coordinate system construction method for non-cooperative targets. Measurement **226**, 114128 (2024)

13. Jakab, T., Tucker, R., Makadia, A., Wu, J., Snavely, N., Kanazawa, A.: Keypointdeformer: Unsupervised 3d keypoint discovery for shape control. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12783–12792 (2021)
14. Kim, S., Joo, M., Lee, J., Ko, J., Cha, J., Kim, H.J.: Semantic-aware implicit template learning via part deformation consistency. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 593–603 (2023)
15. Li, J., Lee, G.H.: Usip: Unsupervised stable interest point detection from 3d point clouds. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 361–370 (2019)
16. Li, Y., Takehara, H., Taketomi, T., Zheng, B., , Nießner, M.: 4dcomplete: Non-rigid motion estimation beyond the observable surface. IEEE International Conference on Computer Vision (ICCV) (2021)
17. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: A skinned multi-person linear model. ACM transactions on graphics (TOG) **34**(6), 1–16 (2015)
18. Ma, Q., Yang, J., Ranjan, A., Pujades, S., Pons-Moll, G., Tang, S., Black, M.J.: Learning to Dress 3D People in Generative Clothing. In: Computer Vision and Pattern Recognition (CVPR) (June 2020)
19. Maharjan, A., Yuan, X.: Registration of human point set using automatic key point detection and region-aware features. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 741–749 (2022)
20. Mohammadi, S.S., Wang, Y., Del Bue, A.: Pointview-gcn: 3d shape classification with multi-view point clouds. In: 2021 IEEE International Conference on Image Processing (ICIP). pp. 3103–3107. IEEE (2021)
21. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 652–660 (2017)
22. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems **30** (2017)
23. Saleh, M., Wu, S.C., Cosmo, L., Navab, N., Busam, B., Tombari, F.: Bending graphs: Hierarchical shape matching using gated optimal transport. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11757–11767 (2022)
24. Sengupta, A., Bartoli, A.: Totem nrsfm: Object-wise non-rigid structure-from-motion with a topological template. International Journal of Computer Vision pp. 1–42 (2024)
25. Shi, J., Yang, H., Carlone, L.: Optimal and robust category-level perception: Object pose and shape estimation from 2-d and 3-d semantic keypoints. IEEE Transactions on Robotics (2023)
26. Shi, R., Xue, Z., You, Y., Lu, C.: Skeleton merger: an unsupervised aligned keypoint detector. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 43–52 (2021)
27. Su, S.Y., Yu, F., Zollhöfer, M., Rhodin, H.: A-nerf: Articulated neural radiance fields for learning human shape, appearance, and pose. Advances in Neural Information Processing Systems **34**, 12278–12291 (2021)
28. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2446–2454 (2020)

29. Suwajanakorn, S., Snavely, N., Tompson, J.J., Norouzi, M.: Discovery of latent 3d keypoints via end-to-end geometric reasoning. Advances in neural information processing systems **31** (2018)
30. Tan, F., Tang, D., Dou, M., Guo, K., Pandey, R., Keskin, C., Du, R., Sun, D., Bouaziz, S., Fanello, S., et al.: Humangps: Geodesic preserving feature for dense human correspondences. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1820–1830 (2021)
31. Tang, J., Gong, Z., Yi, R., Xie, Y., Ma, L.: Lake-net: Topology-aware point cloud completion by localizing aligned keypoints. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1726–1735 (2022)
32. Wang, Q., Kou, C., Liu, P.: Keypoint extraction of auroral arc using curvature-constrained pointnet++. In: Proceedings of the 2022 5th International Conference on Artificial Intelligence and Pattern Recognition. pp. 462–467 (2022)
33. Weng, Z., Gorban, A.S., Ji, J., Najibi, M., Zhou, Y., Anguelov, D.: 3d human keypoints estimation from point clouds in the wild without human labels. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1158–1167 (2023)
34. Xue, Z., Yuan, Z., Wang, J., Wang, X., Gao, Y., Xu, H.: Useek: Unsupervised se (3)-equivariant 3d keypoints for generalizable manipulation. In: 2023 IEEE International Conference on Robotics and Automation (ICRA). pp. 1715–1722. IEEE (2023)
35. Yang, J., Zuo, X., Wang, S., Yu, Z., Li, X., Ni, B., Gong, M., Cheng, L.: Object wake-up: 3d object rigging from a single image. In: European Conference on Computer Vision. pp. 311–327. Springer (2022)
36. Yang, Z., Litany, O., Birdal, T., Sridhar, S., Guibas, L.: Continuous geodesic convolutions for learning on 3d shapes. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 134–144 (2021)
37. You, Y., Liu, W., Ze, Y., Li, Y.L., Wang, W., Lu, C.: Ukpgan: A general self-supervised keypoint detector. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17042–17051 (2022)
38. You, Y., Lou, Y., Shi, R., Liu, Q., Tai, Y.W., Ma, L., Wang, W., Lu, C.: Prin/sprin: On extracting point-wise rotation invariant features. IEEE Transactions on Pattern Analysis and Machine Intelligence **44**(12), 9489–9502 (2021)
39. Yuan, H., Zhao, C., Fan, S., Jiang, J., Yang, J.: Unsupervised learning of 3d semantic keypoints with mutual reconstruction. In: European Conference on Computer Vision. pp. 534–549. Springer (2022)
40. Zanfir, A., Zanfir, M., Gorban, A., Ji, J., Zhou, Y., Anguelov, D., Sminchisescu, C.: Hum3dil: Semi-supervised multi-modal 3d humanpose estimation for autonomous driving. In: Conference on Robot Learning. pp. 1114–1124. PMLR (2023)
41. Zhong, C., You, P., Chen, X., Zhao, H., Sun, F., Zhou, G., Mu, X., Gan, C., Huang, W.: Snake: Shape-aware neural 3d keypoint field. Advances in Neural Information Processing Systems **35**, 7052–7064 (2022)
42. Zhong, C., Zheng, Y., Zheng, Y., Zhao, H., Yi, L., Mu, X., Wang, L., Li, P., Zhou, G., Yang, C., et al.: 3d implicit transporter for temporally consistent keypoint discovery. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3869–3880 (2023)
43. Zhou, B., Zhou, H., Liang, T., Yu, Q., Zhao, S., Zeng, Y., Lv, J., Luo, S., Wang, Q., Yu, X., et al.: Clothesnet: An information-rich 3d garment model repository with simulated clothes environment. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 20428–20438 (2023)

44. Zohaib, M., Del Bue, A.: Sc3k: Self-supervised and coherent 3d keypoints estimation from rotated, noisy, and decimated point cloud data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 22509–22519 (2023)
45. Zohaib, M., Padalkar, M.G., Morerio, P., Taiana, M., Del Bue, A.: Cdhn: Cross-domain hallucination network for 3d keypoints estimation. Available at SSRN 4349267 (2023)
46. Zohaib, M., Taiana, M., Padalkar, M.G., Del Bue, A.: 3d key-points estimation from single-view rgb images. In: International Conference on Image Analysis and Processing. pp. 27–38. Springer (2022)