Unsupervised Learning of Category-Specific Symmetric 3D Keypoints from Point Sets

Clara Fernandez-Labrador[1,2,3], Ajad Chhatkuli[3], Danda Pani Paudel[3],
Jose J. Guerrero[1], Cédric Demonceaux[2], and Luc Van Gool[3,4]

[1] I3A, University of Zaragoza, Spain
[2] VIBOT ERL CNRS 6000, ImViA, Université de Bourgogne Franche-Comté, France
[3] Computer Vision Lab, ETH Zürich, Switzerland
[4] VISICS, ESAT/PSI, KU Leuven, Belgium
{cfernandez,josechu.guerrero}@unizar.es,
cedric.demonceaux@u-bourgogne.fr,
{ajad.chhatkuli,paudel,vangool}@vision.ee.ethz.ch

Abstract. Automatic discovery of category-specific 3D keypoints from a collection of objects of a category is a challenging problem. The difficulty is added when objects are represented by 3D point clouds, with variations in shape and semantic parts and unknown coordinate frames. We define keypoints to be category-specific, if they meaningfully represent objects' shape and their correspondences can be simply established order-wise across all objects. This paper aims at learning such 3D keypoints, in an unsupervised manner, using a collection of misaligned 3D point clouds of objects from an unknown category. In order to do so, we model shapes defined by the keypoints, within a category, using the symmetric linear basis shapes without assuming the plane of symmetry to be known. The usage of symmetry prior leads us to learn stable keypoints suitable for higher misalignments. To the best of our knowledge, this is the first work on learning such keypoints directly from 3D point clouds for a general category. Using objects from four benchmark datasets, we demonstrate the quality of our learned keypoints by quantitative and qualitative evaluations. Our experiments also show that the keypoints discovered by our method are geometrically and semantically consistent.

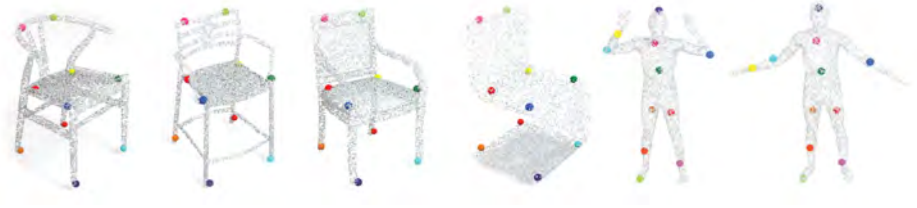


Fig. 1: **Category-specific 3D Keypoints.** The predicted keypoints follow the symmetric linear shape basis prior modeling all instances in a category under a common framework. They not only are consistent across different instances, but also are ordered and correspond to semantically meaningful locations.

1 Introduction

A set of keypoints representing any object is historically of large interest for geometric reasoning, due to their simplicity and ease of handling. Keypoints-based methods [1–3] have been crucial to the success of many vision applications. A few examples include; 3D reconstruction [4–6], registration [7–10], human body pose [11–14], recognition [15,16], and generation [17,18]. That being said, many keypoints are defined manually, while considering their semantic locations such as facial landmarks and human body joints, to address the problem at hand. To further benefit from their widespread utility, several attempts have been made on learning to detect keypoints [19–23], as well as on automatically discovering them [24–27]. In this regard, the task of learning to detect keypoints from several supervision examples, has achieved many successes [20,28]. However, discovering them automatically from unlabeled 3D data –such that they meaningfully represent shapes and semantics– so as to have a similar utility as those of manually defined, has received only limited attention due to its difficulty.

As objects of interest reside in the 3D space, it is not surprising that 3D keypoints are preferred for geometric reasoning. For the given 3D keypoints, their counterparts in 2D images can be associated by merely using camera projection models [29–31]. However, being able to directly predict keypoints on provided 3D data (point clouds) has the advantage that the task can be achieved when multiple camera views or images are not available. In this work, we are interested on learning keypoints using only 3D structures. In fact, 3D structures with keypoints suffice for several applications including, registration [32], shape completion [33], and shape modeling [34]; without requiring their 2D counterparts.

When 3D objects go through shape variations, due to deformation or when two different objects of a category are compared, consistent keypoints are desired for meaningful geometric reasoning. Recall the examples of semantic keypoints such as facial landmarks and body joints. To serve a similar purpose, *can we automatically find kepoints that are consistent over inter-subject shape variations and intra-subject deformations in a category?* This is the primary question that we are interested to answer in this paper. Furthermore, we wish to discover such keypoints directly from 3D point sets, in an unsupervised manner. We call these keypoints "category-specific", which are expected to meaningfully represent objects' shape and offer their correspondence order-wise across all objects. More formally, we define the desired properties of category-specific keypoints as: i) generalizability over different shape instances and alignments in a category, ii) one-to-one ordered correspondences and semantic consistency, iii) representative of the shape as well as the category while preserving shape symmetry. These properties not only make the representation meaningful, but also tend to enhance the usefulness of keypoints. Learning category-specific keypoints on point clouds, however, is a challenging problem because not all the object parts are always present in a category. The challenges are exacerbated when the practical cases of misaligned data and unsupervised learning are considered. Related works do not address all these problems, but instead opt for; dropping category-specificity and using aligned data [26], employing manual supervision on 2D images [20], or using aligned 3D and multiple 2D images with known pose [27]. The latter method achieves category-specificity without explicitly reasoning on the shapes. Yet another work leverages predefined local shape descriptors and a template model [35] specifically on faces.

In this paper, we show that the category-specific keypoints with the listed properties can be learned unsupervised by modeling them with non-rigidity, based on unknown linear basis shapes. We further impose an unknown reflective symmetry on the deformation model, when considering categories with instance-wise symmetry. For categories where instance-wise symmetry is not applicable, we propose the use of symmetric linear basis shapes in order to better model, what we define as symmetric deformation spaces, e.g., human body deformations. This allows us to better constrain the pose and the shape coefficients prediction. Our proposed learning method does not assume aligned shapes [27], pre-computed basis shapes [20] or known planes of symmetry [36] and all quantities are learned in an end-to-end manner. Our symmetry modeling is powerful and more flexible compared to that of previous NRSfM methods [36, 37]. We achieve this by considering the shape basis for a category and the reflective plane of symmetry as the neural network weight variables, optimized during the training process. The training is done on a single input, circumventing the Siamese-like architecture used in [7, 26]. At inference time, the network predicts the basis coefficients and the pose in order to estimate the instance-specific keypoints. Using multiple categories from four benchmark datasets, we evaluate the quality of our learned keypoints both quantitatively and with qualitative visualization. Our experiments show that the keypoints discovered by our method are geometrically and semantically consistent, which are measured respectively by intra-category registration and semantic part-wise assignments. We further show that symmetric basis shapes can be used to model symmetric deformation space of categories such as the human body.

2 Related Work

Category-specific keypoints on objects have been extensively used in NRSfM methods, however, only few methods have tackled the problem of estimating them. In terms of the outcome, our work is closest to [27], which learns category-specific 3D keypoints by solving an auxiliary task of rigid registration between multiple renders of the same shape and by considering the category instances to be pre-aligned. Although the method shows promising results on 2D and 3D, it does so without explicitly modeling the shapes. Consequently, it requires renders of different instances to be pre-aligned to reason on keypoint correspondences between instances. A similar task is also solved in [20] for 6-degrees of freedom (DoF) estimation which uses low-rank shape prior to condition keypoints in 3D. Although, the low-rank shape modeling is a powerful tool, [20] requires supervision for heatmap prediction and relies on aligned shapes and pre-computed shape basis. [28] also predicts keypoints for categories with low-rank shape prior but the method is again trained on fully supervised manner. Moreover, all of the mentioned methods learn keypoints on images as heatmaps and thereafter lift them to 3D. Different from the other works, [35] exploits deformation model and symmetry to directly predict keypoints on 3D but requires a face template, aligned shapes and known basis. Shape modeling of category shape instances has been widely explored in NRSfM works. Linear low-rank shape basis [5, 38, 39], low-rank trajectory basis [40], isometry or piece-wise rigidity [41, 42] are some of the different methods used for NRSfM. Recently, a few number of works have used low-rank shape basis in order to devise learned methods [4, 28, 36, 43]. Another useful tool in modeling shape category is the reflective symmetry, which is also directly related to the object pose. Although [37] showed that the low-rank shape basis can be formulated with unknown reflective symmetry, its adaptation to learned NRSfM methods is not trivial. Recent methods, in fact, assume that the plane of symmetry is one among a few known planes [44]. Moreover, none of the methods formulate symmetry applicable for non-rigidly deforming objects such as the human body. A parallel work [45] on this regard models symmetry probabilistically in a warped canonical space to reconstruct 3D of different objects.

While shape modeling is a key aspect of our work, another challenge is to infer ordered keypoints by learning on unordered point sets. Despite several advances on deep neural networks for point sets [46–48], current achievements of learning on images dwarf those of learning on point sets. A related work learns to predict 3D keypoints unsupervised by again solving the auxiliary task of correctly estimating rotations in a Siamese architecture [49]. The keypoint prediction is done without order by pooling features of certain point neighborhoods. Another previous work [7] proposes learning point features for matching, again using alignment as the auxiliary task. Matching such keypoints across shapes is not an easy task as the keypoints are not predicted in any order. In the following sections we show how one can model shape instances using the low-rank symmetric shape basis and use the shape modeling to predict ordered category-specific keypoints.

3 Background and Theory

Notations. We represent sets and matrices with special Latin characters (e.g., $\mathcal{V}$) or bold Latin characters (e.g., V). Lower or uppercase normal fonts, e.g., K denote scalars. Lowercase bold Latin letters represent vectors as in v. We use lowercase Latin letters to represent indices (e.g., i). Uppercase Greek letters represent mappings or functions (e.g., Π). We use $\mathcal{L}$ to denote loss functions. Finally the operator mat(.) converts a vector $\mathsf{v} \in \mathbb{R}^{3N \times 1}$ to a matrix $\mathsf{M} \in \mathbb{R}^{3 \times N}$.

3.1 Category-specific Shape and Keypoints

We represent shapes as point clouds, defined as an unordered set of points $\mathsf{S} = \{\mathsf{s}_1, \mathsf{s}_2, \ldots, \mathsf{s}_M\},\ \mathsf{s}_j \in \mathbb{R}^3,\ j \in \{1, 2, \ldots, M\}$. The set of all such shapes in a category defines the category shape space $\mathcal{C}$. We write a particular i-th category-specific shape instance in $\mathcal{C}$ as S_i. For convenience, we will use the terms category-specific shape and shape interchangeably. The category shape space $\mathcal{C}$ can be anything from a set of discrete shapes to a smooth manifold of category-specific shapes spanned by a deformation function $\mathit{\Psi}_{\mathcal{C}}$. The focus of the work is on learning meaningful 3D keypoints from the point set representation of S_i. To that end, this section defines category-specific keypoints and develops their modeling.

Category-specific keypoints. We represent category-specific keypoints of a shape S_i as a sparse tuple of points, $\mathsf{P}_i = (\mathsf{p}_{i1}, \mathsf{p}_{i2}, \dots, \mathsf{p}_{iN}),\ \mathsf{p}_{ij} \in \mathbb{R}^3, j \in \{1, 2, \dots, N\}$. Unlike the shape, its keypoints are represented as ordered points. Our objective is to learn a mapping $\mathit{\Pi}_{\mathcal{C}} : \mathsf{S}_i \rightarrow \mathsf{P}_i$ in order to obtain the category-specific keypoints from an input shape S_i in $\mathcal{C}$. Although not completely unambiguous, we can define the category-specific keypoints using the properties listed in Sec. 1. In mathematical notations they are:

(i) Generalization: $\mathit{\Pi}_{\mathcal{C}}(\mathsf{S}_i) = \mathsf{P}_i,\ \forall \mathsf{S}_i \in \mathcal{C}$.
(ii) Corresponding points and semantic consistency: Given $\mathsf{S}_a, \mathsf{S}_b \in \mathcal{C}$, we want $\mathsf{p}_{aj} \Leftrightarrow \mathsf{p}_{bj}$. Similarly, p_{aj} and p_{bj} should have the same semantics.
(iii) Representative-ness: $\text{vol}(\mathsf{S}_i) = \text{vol}(\mathsf{P}_i)$ and $\mathsf{p}_{ij} \in \mathsf{S}_i$, where vol(.) is the Volume operator for a shape. If $\mathsf{S}_i \in \mathcal{C}$ has a reflective symmetry, P_i should have the same symmetry.

3.2 Category-specific Shapes as Instances of Non-Rigidity

Several recent works have modeled shapes in a category as instances of non-rigid deformations [4, 28, 36, 43]. The motivation lies in the fact that such shapes often share geometric similarities. Consequently, there likely exists a deformation function $\mathit{\Psi}_{\mathcal{C}} : \mathsf{S}_T \rightarrow \mathsf{S}_i$, which can map a global shape property S_T (shape template or basis shapes) to a category shape instance S_i. However, we argue that modeling $\mathit{\Psi}_{\mathcal{C}}$ is not trivial and in fact a convenient representation of $\mathit{\Psi}_{\mathcal{C}}$ may not exist in many cases. This observation, in fact, is what makes the dense Non-Rigid Structure-from-Motion (NRSfM) so challenging. On the other hand, one can imagine a deformation function $\mathit{\Phi}_{\mathcal{C}} : \mathsf{P}_T \rightarrow \mathsf{P}_i$, going from a global keypoints property P_T to the category-specific keypoints P_i. The deformation function $\mathit{\Phi}_{\mathcal{C}}$ thus satisfies: $\mathsf{p}_{ij} \in \mathit{\Phi}_{\mathcal{C}}$ implies $\mathsf{p}_{ij} \in \mathit{\Psi}_{\mathcal{C}}$ and effectively, $\mathit{\Phi}_{\mathcal{C}} \subset \mathit{\Psi}_{\mathcal{C}}$, if the set order in P_i is ignored. Unlike $\mathit{\Psi}_{\mathcal{C}}$, the deformation function $\mathit{\Phi}_{\mathcal{C}}$ may be simple enough to model and use for estimating the category-specific keypoints P_i. We therefore, choose to seek the non-rigidity modeling in the space of keypoints $\mathcal{P} = \{\mathsf{P}_1, \mathsf{P}_2, \dots, \mathsf{P}_L\}$, which functions as an abstraction of the space $\mathcal{C}$. Non-rigidity can be used to define the prediction function $\mathit{\Pi}_{\mathcal{C}}$ as below:

$$\mathit{\Pi}_{\mathcal{C}}(\mathsf{S}_i; \theta) = \mathit{\Phi}_{\mathcal{C}}(\mathsf{r}_i; \theta) = \mathsf{P}_i \tag{1}$$

where θ denotes the constant function parameters of $\mathit{\Pi}_{\mathcal{C}}$ and r_i is the predicted instance specific vector parameter. In our problem, we want to learn θ from the example shapes in $\mathcal{C}$ without using the ground-truth labels, supervised by $\mathit{\Phi}_{\mathcal{C}}$. In the NRSfM literature, two common approaches of modeling shape deformations are the low-rank shape prior [5, 38–40] and the isometric prior [41, 42]. In this paper, we investigate the modeling using the low-rank shape prior, with instance-wise symmetry as well as symmetry of the deformation space.

3.3 Low-Rank Non-rigid Representation of Keypoints

The NRSfM approach of low-rank shape basis comes as a natural extension of the rigid orthographic factorization prior [50] and was introduced by Bregler et al. [38]. The key idea is that a large number of object deformations can be explained by linearly combining a smaller K number of basis shapes at some pose. In the rigid case, this number is one, hence the rank is 3. In the non-rigid case, it can be higher, while the exact value depends on the complexity of the deformations. Consider F shape instances in $\mathcal{C}$ and N points in each keypoints instance P_i. The following equation describes the projection with shape basis.

$$\mathsf{P}_i = \mathit{\Phi}_{\mathcal{C}}(\mathsf{r}_i; \theta) = \mathsf{R}_i \operatorname{mat}(\mathcal{B}_{\mathcal{C}}\, \mathsf{c}_i) \tag{2}$$

where $\mathcal{B}_{\mathcal{C}} = (\mathsf{B}_1, \dots, \mathsf{B}_K), \mathcal{B}_{\mathcal{C}} \in \mathbb{R}^{3N \times K}$ forms the low-rank shape basis. The rank is lower than the maximum possible rank of $3F$ or N for $3K < 3F$ or $3K < N$. The vector $\mathsf{c}_i \in \mathbb{R}^K$ denotes the coefficients that linearly combines different basis for the keypoints instance i. Each keypoints instance is then completely parametrized by the basis $\mathcal{B}_{\mathcal{C}}$ and the coefficients c_i. Next, the projection matrix $\mathsf{R}_i \in SO_3$ is simply the rotation matrix for the shape instance i.

Unlike in NRSfM, the problem of computing the category-specific keypoints, has P_i as unknown. Similar to NRSfM, the rest of the quantities in Eq. (2) – c_i, $\mathcal{B}_{\mathcal{C}}$ and R_i are also unknown. This fact makes our problem doubly hard. First the problem becomes more than just lifting the 2D keypoints to 3D and second, the order of keypoints present in the NRSfM measurements matrix is not available. We intend to solve the aforementioned problems by learning based on Eq. (2), which is related to the deformation representation of $\mathit{\Phi}_{\mathcal{C}}$ in Eq. (1). Here, θ includes the global parameters or basis $\mathcal{B}_{\mathcal{C}}$ and r_i includes the instance-wise pose R_i and coefficients c_i. To further reduce ambiguities on pose, we propose to also compute the reflective plane of symmetry for a category.

3.4 Modeling Symmetry with Non-Rigidity

Many object categories have shapes which exhibit a fixed reflective symmetry over the whole category. To discover and use symmetry, we consider two different priors: instance-wise symmetry and symmetric deformation space.

Instance-wise symmetry. Instance-wise reflective symmetry about a fixed plane is observed in a large number of rigid object categories (e.g. ShapeNet [51] and ModelNet [52]). Such a symmetry has been previously combined with the shape basis prior in NRSfM [37], however, a convenient representation for learning both the symmetry and the shapes have not been explored yet. A recent learning-based method [36,44] uses the symmetry prior by performing an exhaustive search over a few planes in order to predict symmetric dense non-rigid shapes. However, such a strategy may not work when the shapes are not perfectly aligned. Instance-wise symmetry can be included by re-writing Eq. (2) as follows:

$$\mathsf{P}_{i\frac{1}{2}} = \mathsf{R}_i \operatorname{mat}(\mathcal{B}_{\mathcal{C}\frac{1}{2}}\, \mathsf{c}_i), \quad \mathsf{P}_i = \left[\mathsf{P}_{i\frac{1}{2}}\ A_{\mathcal{C}} \mathsf{P}_{i\frac{1}{2}}\right] \tag{3}$$

where $\mathsf{P}_{i\frac{1}{2}} \in \mathbb{R}^{3\times N/2}$ represents one half of the category-specific keypoints. $\mathsf{P}_{i\frac{1}{2}}$ is reflected using $\mathsf{A}_{\mathcal{C}} \in \mathbb{R}^{3\times 3}$ and concatenated to obtain the final keypoints. Due to the exact instance-wise symmetry, we similarly can parametrize the basis as $\mathcal{B}_{\mathcal{C}\frac{1}{2}} \in \mathbb{R}^{3N/2\times K}$ to denote the shape basis for the first half of the keypoints. The reflection operator $\mathsf{A}_{\mathcal{C}}$ is parametrized by a unit normal vector $\mathsf{n}_{\mathcal{C}} \in \mathbb{R}^3$ of the plane of symmetry passing through the origin. The advantage of going from Eq. (2) to Eq. (3) should be apparent from the reduced dimensionality of the unknowns in $\mathcal{B}_{\mathcal{C}}$ as well as the additional second equality constraint of Eq. (3), which reduces the ambiguities in NRSfM [37].

Symmetric deformation space. In many non-rigid objects, shape instances are not symmetric. However, symmetry may still exist in the deformation space, e.g., in a human body. Suppose that a particular shape instance $\mathsf{S}_k \in \mathcal{C}$ has the reflective symmetry about $\mathsf{n}_{\mathcal{C}}$, which allows us to define its two halves: $\mathsf{S}_{k\frac{1}{2}}$ and $\mathsf{S}'_{k\frac{1}{2}}$ and thus correspondingly for all shape instances.

Definition 1 (Symmetric deformation space). *$\mathcal{C}$ is a symmetric deformation space if for every half shape deformation instance $\mathsf{S}_{i\frac{1}{2}}$, there exists any shape instance $\mathsf{S}_j \in \mathcal{C}$ such that the $\mathsf{S}'_{j\frac{1}{2}}$ is symmetric to $\mathsf{S}_{i\frac{1}{2}}$.*

The above definition also applies for the keypoints shape space $\mathcal{P}$. The instance-wise symmetric space is a particular case of the above. However, Eq. (3) cannot model the keypoints instances in the symmetric deformation space. We model such keypoints by introducing symmetric basis that can be weighted asymmetrically, thereby, obtaining the following:

$$\mathsf{P}_i = \mathsf{R}_i \left[\mathrm{mat}(\mathcal{B}_{\mathcal{C}\frac{1}{2}}\, \mathsf{c}_i)\ \mathrm{mat}(\mathcal{B}'_{\mathcal{C}\frac{1}{2}}\, \mathsf{c}'_i)\right] \tag{4}$$

where $\mathcal{B}'_{\mathcal{C}\frac{1}{2}}$ is obtained by reflecting $\mathcal{B}_{\mathcal{C}\frac{1}{2}}$ with $A_{\mathcal{C}}$ and $\mathsf{c}'_i \in \mathbb{R}^K$ forms the coefficients for the second half of the basis. Although Eq. (4) increases the dimension of the unknowns in the coefficients over Eq. (2), the added modeling of the symmetry of the deformation space and the reduced dimensionality of the basis can improve the final keypoints estimate. This brings us to the following proposition.

Proposition 1. *Provided that $\mathcal{B}_{\mathcal{C}\frac{1}{2}}$ and $\mathcal{B}'_{\mathcal{C}\frac{1}{2}}$ are symmetric about a plane, Eq.* (4) *models a symmetric deformation space if the estimates of c_i and c'_i come from the same probabilistic distribution.*

Proof. The proof is straightforward and provided in the supplementary material.

As a consequence of Proposition 1, we can model keypoints in non-rigid symmetric objects with Eq. (4), while also tightly modeling the symmetry as long as we maintain the distribution of c and c' to be the same.

4 Learning Category-specific Keypoints

In this section, we use the modeling of $\varPhi_{\mathcal{C}}$ to describe the unsupervised learning process of the category-specific keypoints. More precisely, we want to learn the function $\varPi_{\mathcal{C}} : \mathsf{S}_i \rightarrow \mathsf{P}_i$ as a neural network of parameters θ, using the supervisory signal from $\varPhi_{\mathcal{C}}$. In regard to learning keypoints on point sets, recent work [26] trains a Siamese network to predict order-agnostic keypoints stable to rotations for rigid objects [26]. Part of our network architecture is inspired from [26], which is based on PointNet [46]. However, we use a single input avoiding the expensive Siamese training. The network architecture is shown in Fig. 2, whose input consists of a single shape S_i misaligned in SO_2. This is reasonable since point clouds are usually aligned to the vertical direction. We describe the different components of the network architecture below.

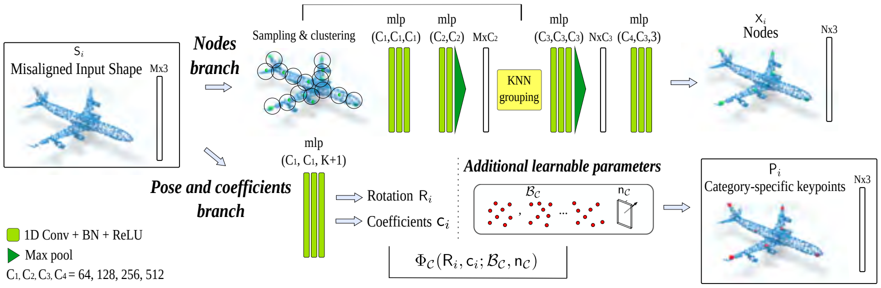

S_i
Misaligned Input Shape
Mx3
Nodes branch
Sampling & clustering
mlp
(C1,C1,C1)
mlp
(C2,C2)
MxC2
KNN grouping
mlp
(C3,C3,C3)
NxC3
mlp
(C4,C3,3)
X_i
Nodes
Nx3
Pose and coefficients branch
mlp
(C1, C1, K+1)
Rotation R_i
Coefficients c_i
Additional learnable parameters
$\mathcal{B}_{\mathcal{C}}$
$\mathsf{n}_{\mathcal{C}}$
$\Phi_{\mathcal{C}}(\mathsf{R}_i, \mathsf{c}_i; \mathcal{B}_{\mathcal{C}}, \mathsf{n}_{\mathcal{C}})$
P_i
Category-specific keypoints
Nx3
🟩 1D Conv + BN + ReLU
▶ Max pool
C1, C2, C3, C4 = 64, 128, 256, 512

Fig. 2: **Network architecture:** The *pose and coefficients branch* and the *additional learnable parameters* generate the output category-specific keypoints. The *nodes branch* estimates the nodes that guide the learning process. "mlp" stands for multi-layer perceptron. Refer to Sec. 3 for the modeling, Sec. 4 for learning.

Node branch. This branch estimates a sparse tuple of nodes that are potentially category-specific keypoints but are not ordered. We denote them as $\mathsf{X}_i = \{\mathsf{x}_{i1}, \mathsf{x}_{i2}, \dots, \mathsf{x}_{iN}\}, \mathsf{x}_{ij} \in \mathbb{R}^3$ and $j \in \{1, 2, \dots, N\}$. Initially, a predefined number of nodes N are sampled from the input shape using the Farthest Point Sampling (FPS) and a local neighborhood of points is built for each node with point-to-node grouping [26, 53], creating N clusters which are mean normalized inside the network. Every point in S_i is associated with one of these nodes. The branch consists of two PointNet-like [46] networks followed by a kNN grouping layer that uses the initial sampled nodes to achieve hierarchical information aggregation. Finally, the local feature vectors are fed into a Multi-Layer Perceptron (MLP) that outputs the nodes.

Pose and coefficients branch. We predict the quantities R_i and c_i with this branch. We use a single rotation angle to parametrize R_i. The branch consists of an MLP that estimates the mentioned parameters. The output size varies depending on whether we are interested in symmetric shape instances as in Eq. (3) or symmetric basis as in Eq. (4), the size being double in the latter.

Additional learnable parameters. Several unknown quantities in Eq. (3) or (4) are constant for a category shape space $\mathcal{C}$. Such quantities need not be predicted instance-wise. We rather choose to optimize them as part of the network parameters θ. They are the shape basis $\mathcal{B}_{\mathcal{C}} \in \mathbb{R}^{3N \times K}$ and the unit normal of the plane of symmetry $\mathsf{n}_{\mathcal{C}} \in \mathbb{R}^3$. We observed that a good choice for the number of shape basis is $5 \leq K \leq 10$. In fact, the generated keypoints are not very sensitive to the choice of K, as a large K tends to generate sparser shape coefficients and similar keypoints. Depending upon the problem, alternate parametrization can be considered for $\mathsf{n}_{\mathcal{C}}$, e.g., Euler angles.

At inference time, we apply Non-Maximal Suppression obtaining the final N' number of keypoints. Our method consistently provides N' keypoints for all instances in the category, as they follow the same geometric model.

4.1 Training Losses

In order to adhere to the definitions of the category-specific keypoints introduced in Sec. 1 as well as our shape modeling, we design our loss functions as below.

Chamfer loss with symmetry and non-Rigidity. Eq. (1) suggests that the neural network $\Pi_{\mathcal{C}}$ can be trained with an ℓ_2 loss between the node predictions X_i and the deformation function $\mathsf{P}_i = \Phi_{\mathcal{C}}(\mathsf{R}_i, \mathsf{c}_i; \mathcal{B}_{\mathcal{C}}, \mathsf{n}_{\mathcal{C}})$, thus obtaining $\mathsf{P}_i = \mathsf{X}_i$. However, as confirmed by our evaluations as well as in [26], the ℓ_2 loss does not converge as the network is unable to predict the point order. Alternatively, the Chamfer loss [54] does converge, minimizing the distance between each point x_{ik} in the first set X_i and its nearest neighbor p_{ij} in the second set P_i and vice versa.

$$\mathcal{L}_{chf} = \sum_{k=1}^{N} \min_{\mathsf{p}_{ij} \in \mathsf{P}_i} \|\mathsf{x}_{ik} - \mathsf{p}_{ij}\|_2^2 + \sum_{j=1}^{N} \min_{\mathsf{x}_{ik} \in \mathsf{X}_i} \|\mathsf{x}_{ik} - \mathsf{p}_{ij}\|_2^2, \tag{5}$$

The Chamfer loss in Eq. (5) ensures that the learned keypoints follow a generalizable category-specific property – that they are a linear combination of common basis learned specifically for the category. To additionally model symmetry, Eq. (3) or (4) is directly used in Eq. (5). Therefore, two different Chamfer losses are possible modeling two different types of symmetries.

Coverage and inclusivity loss. The Chamfer loss in Eq. (5) does not ensure that the keypoints follow the object shape. However, one can add the following conditions: a) the keypoints cover the whole category shape (coverage loss), b) the keypoints are not far from the point cloud (inclusivity loss). The coverage loss can be defined as a Huber loss between the volume of the nodes X_i and that of the input shape S_i, using the product of the singular values. However, we instead approximate the volume using the 3D bounding box defined by the points. This improves the training speed and, based on our initial evaluations, also does not harm performance. The coverage loss is thus given by:

$$\mathcal{L}_{cov} = \|\mathrm{vol}(\mathsf{X}_i) - \mathrm{vol}(\mathsf{S}_i)\| \tag{6}$$

The inclusivity loss is formulated as a single side Chamfer loss [55] which penalizes nodes in X_i that are far from the original shape S_i, similarly to Eq. (5):

$$\mathcal{L}_{inc} = \sum_{k=1}^{N} \min_{\mathsf{s}_{ij} \in \mathsf{S}_i} \|\mathsf{x}_{ik} - \mathsf{s}_{ij}\|_2^2. \tag{7}$$

5 Experimental Results

We conduct experiments to evaluate the desired properties of the proposed category-specific keypoints and show their generalization over indoor/outdoor objects and rigid/non-rigid objects with four different datasets in total (Sec. 5.1, 5.2). All these properties are also compared with a proposed baseline. We then evaluate the practical use of our keypoints for intra-category shapes registration (Sec. 5.3), analyzing the influence of symmetry. Additional qualitative results are shown in Fig. 1 and the supplementary material.

Datasets. We use four main datasets. They are ModelNet10 [52], ShapeNet parts [51], Dynamic FAUST [56] and Basel Face Model 2017 [57]. Since our method is category-specific, we require separate training data for each class in the datasets. For indoor rigid objects, we choose three categories from ModelNet10 [52]; chair, table and bed. Three outdoor rigid object categories: airplane, car and motorbike, are evaluated from ShapeNet parts [51]. For non-rigid objects, we randomly choose a sequence of the Dynamic Faust [56], that provides high-resolution 4D scans of human subjects in motion. Finally, we generate shape models of faces using the Basel Face Model 2017 [57] combining 50 different shapes and 20 different expressions. All models are normalized in the range -1 to 1 and are randomly misaligned within ± 45 degrees.

Baseline. Since this is the first work computing category-specific keypoints from point sets, we construct our own baseline based on the recent work USIP [26]. The method detects stable interest points in 3D point clouds under arbitrary transformations and is also unsupervised, which makes it the closest method for comparison. The USIP detector is not category-based, so we train the network per category to create the baseline. Additionally, we adapt the number of predicted keypoints so that the results are directly comparable to ours. While training with some of the categories, specifically car and bed, we observe that predicting lower number of keypoints can lead to some degeneracies [26].

Implementation details. Input point clouds of dimension 3×2000 are used. We implement the network in Pytorch [58] and train it end-to-end from scratch using the Adam optimizer [59]. The initial learning rate is 10^{-3}, which is exponentially decayed by a rate of 0.5 every 40 epochs. We use a batch size of 32 and train each model until convergence, for 200 epochs. The final loss function combines the three training losses, Eqs. (5), (6) and (7), and are weighted as follows: $w_{chf} = w_{cov} = 1$ and $w_{inc} = 2$. For ModelNet10 and ShapeNet parts, we use the training and testing split provided by the authors. For the Basel Face Model 2017, we follow the common practice and split the 1000 generated faces in 85% training and 15% test. We use the same split strategy for the sequence '50009_jiggle_on_toes' of Dynamic Fuaust, which contains 244 examples.

Category	*Coverage*	*Model Err*	*Correspondence*	*Inclusivity*	*Sym Err*	*Definition*
	%	%	%	%	°	
chair	**88.83**	0.72	**100**	90.46	0.40	10
table	**93.33**	0.99	**100**	93.38	2.86	6
bed	**80.31**	0.94	**100**	**95.33**	0.13	6
airplane	**89.15**	0.64	**100**	**96.35**	0.20	8
car	**92.39**	0.72	**100**	**97.77**	2.21	8
motorbike	**96.13**	0.79	**100**	**90.53**	1.42	8
human body	**85.59**	0.72	**100**	97.73	33.30	11
faces	**97.93**	0.41	**100**	**100**	0.15	9
chair	79.73	–	55.6	**98.50**	–	10
table	79.72	–	34.5	**99.83**	–	6
bed	42.18	–	49.33	70.00	–	6
airplane	69.24	–	47.5	87.13	–	8
car	26.87	–	32.18	74.0	–	8
motorbike	75.29	–	48.14	84.57	–	8
human body	72.66	–	50.45	**100**	–	11
faces	42.98	–	30.11	**100**	–	9

Table 1: *Properties Analysis:* Top (ours) and bottom (baseline [26]). For coverage, correspondence and inclusivity *higher is better*, and for model and symmetry error *lower is better*. We empirically show the desired properties of our keypoints, as well as the generalization of our method over indoor/outdoor and rigid/non-rigid objects. Best results are in bold.

5.1 Desired Properties Analysis

As described in Sec. 1 and 3, the category-specific keypoints satisfy certain desired properties. We propose six different metrics to evaluate the properties which are also used for comparison against the baseline. All the results are presented in Table 1, and are averaged across the test samples.

Coverage : According to property *iii)*, we seek keypoints that are representative of each instance shape as well as of the category itself. To measure it, we calculate the percentage of the input shape covered by the keypoints' 3D bounding box. On average, we achieve a 29.4% more coverage than the baseline.

Model Error : This metric refers to the Chamfer distance between the estimated nodes and the learned category-specific keypoints, normalized by the model's scale. We obtain less than 1% of error in all the categories, meaning

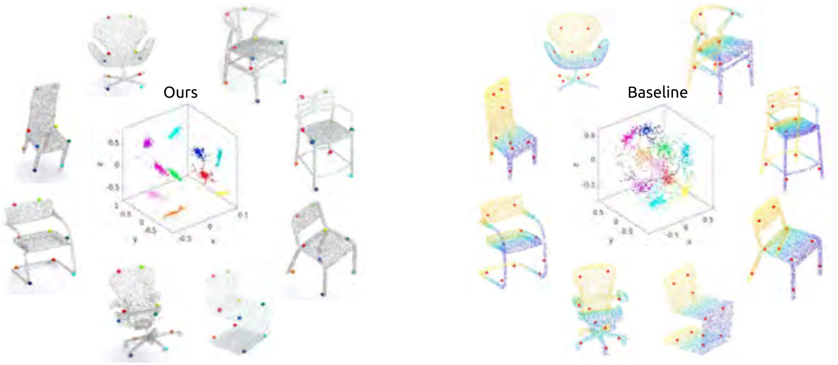


Fig. 3: **Keypoints correspondence/repeatability across instances**. We cluster the predicted keypoints for all the instances in the category to show their geometric consistency. Note how our keypoints are neatly clustered as they are consistently predicted in the corresponding geometric locations, unlike the baseline keypoints. (Note: cluster colors do not correspond to keypoint colors.)

that the network satisfactorily manages to generalize, describing the nodes with the symmteric non-rigidity modeling (Properties *i)* and *iii)*).

Correspondence/ Repeatability : We measure the ability of the model to find the same set of keypoints on different instances of a given category (Property *ii)*). For our method, we cluster the keypoints using their inherent order whereas for the baseline, we use K-means clustering to evaluate and compare this property. We show the evaluation in Fig. S2, the rest of the categories are provided in the supplementary material. One can see at a glance how our keypoints are well clustered, unlike the baseline keypoints. Numerically, we show the % occurrence of each specific keypoint belonging to the same cluster across instances. Our keypoints satisfy 100% the correspondence/repeatability test thanks to our geometric non-rigidity modelling.

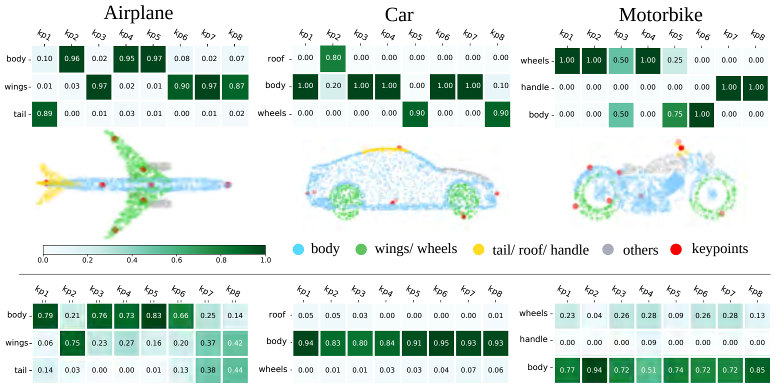
Inclusivity : We measure the percentage of keypoints that lie inside the point cloud (of scale 2) within a chosen threshold of 0.15, which also proves property *iii)*. This is the only metric in which our method doesn't outperform the baseline in all cases. On average, our method achieves $\sim$ 95% inclusivity compared to $\sim$ 89% for the baseline.

Symmetry: The metric shows the angle error of the predicted reflective plane of symmetry. We obtain highly accurate prediction for rigid categories. In the non-rigid human body shape however, the ambiguities are severe. Despite that, the learned keypoints satisfy the other properties, particularly that of semantic correspondence. Both of these facts can be observed in Fig. 1.

Definition: final number of keypoints N' predicted per category after the Non-Maximal Suppression.

5.2 Semantic Consistency

We use the ShapeNet part dataset [51] to show the semantic consistency of the proposed keypoints. Following the low-rank non-rigidity modelling, the keypoints lie on geometrically corresponding locations. The idea of the experiment is to measure keypoint-semantics relationship for every keypoint across instances of the category. The results are presented in Fig. 4 as covariance matrices, along with keypoint visualizations per category for our method. On average, the proposed keypoints have a high semantic consistency of 93% across instances, despite the large intra-category variability. The same experiment is performed for the baseline and presented in bottom of Fig. 4. Here, the degeneracy causes all the keypoints to approach the object centroid for 'Car'. Nonetheless, we observe no semantic consistency even for 'Airplane' without degeneracies. Our model, aiming for a common representation for all the instances of the category, avoids placing keypoints in less representative parts or unique parts, e.g., arm rests in chairs (in Fig. 1), engines in airplanes or gas tank in motorbikes. This highlights significant robustness achieved in modelling and learning the keypoints.



Airplane

	kp1	kp2	kp3	kp4	kp5	kp6	kp7	kp8
body	0.10	0.96	0.02	0.95	0.97	0.08	0.02	0.07
wings	0.01	0.03	0.97	0.02	0.01	0.90	0.97	0.87
tail	0.89	0.00	0.01	0.03	0.01	0.00	0.01	0.02

Car

	kp1	kp2	kp3	kp4	kp5	kp6	kp7	kp8
roof	0.00	0.80	0.00	0.00	0.00	0.00	0.00	0.00
body	1.00	0.20	1.00	1.00	0.00	1.00	1.00	0.10
wheels	0.00	0.00	0.00	0.00	0.90	0.00	0.00	0.90

Motorbike

	kp1	kp2	kp3	kp4	kp5	kp6	kp7	kp8
wheels	1.00	1.00	0.50	1.00	0.25	0.00	0.00	0.00
handle	0.00	0.00	0.00	0.00	0.00	0.00	1.00	1.00
body	0.00	0.00	0.50	0.00	0.75	1.00	0.00	0.00

	kp1	kp2	kp3	kp4	kp5	kp6	kp7	kp8
body	0.79	0.21	0.76	0.73	0.83	0.66	0.25	0.14
wings	0.06	0.75	0.23	0.27	0.16	0.20	0.37	0.42
tail	0.14	0.03	0.00	0.00	0.01	0.13	0.38	0.44

	kp1	kp2	kp3	kp4	kp5	kp6	kp7	kp8
roof	0.05	0.05	0.03	0.00	0.00	0.00	0.00	0.01
body	0.94	0.83	0.80	0.84	0.91	0.95	0.93	0.93
wheels	0.00	0.01	0.01	0.03	0.03	0.04	0.07	0.06

	kp1	kp2	kp3	kp4	kp5	kp6	kp7	kp8
wheels	0.23	0.04	0.26	0.28	0.09	0.26	0.28	0.13
handle	0.00	0.00	0.00	0.09	0.00	0.00	0.00	0.00
body	0.77	0.94	0.72	0.51	0.74	0.72	0.72	0.85

Fig. 4: **Semantic part correspondence.** Top to bottom: the semantic correspondence for the proposed keypoints, qualitative results and the baseline semantic correspondence. Our predicted keypoints show the correct semantic correspondence across the category.

5.3 Object Pose and Intra-category Registration

Previous methods do not handle misaligned data due to the obvious difficulty it poses to unsupervised learning. This deserves special attention since real data is never aligned. In this section we evaluate the intra-category registration performance of our model and show the impact of the different symmetry models proposed. These results implicitly measure the object poses estimated as well.

Rotation Ambiguities. Recent unsupervised approaches for keypoint detection actually self-supervise rotation during training, e.g., [26, 27], and highlight that it is crucial for achieving a good performance. In our case, we do not directly supervise the rotations. Therefore, the different combination of basis shapes can result in different alignments. This implies that computing P_i with the deformation function $\Phi_\mathcal{C}$ will give the correct set of keypoints along with the correct plane of symmetry, but the predicted rotation alone is not meaningful for registration.

Experimental setup. Despite the above ambiguity, an important characteristic of the proposed keypoints is that they are ordered, which empowers direct inter-instances registration since no extra descriptors are needed for matching. We perform experiments for the chair category, using 10 keypoints (Table 1) and a misalignment of ±45 degrees. Three different models are compared. The first one is trained without symmetry awareness following Eq. (2). A second one uses shape symmetry during training as shown in Eq. (3). The last model is trained with basis symmetry as in Eq. (4). We attempt to register keypoints in each instance to those of randomly chosen three aligned templates by computing a similarity transformation and observe the mean error. Fig. 5 shows that symmetry helps to have more control over the rotations and tackle higher misalignment. More results and analysis are provided in the supplementary.

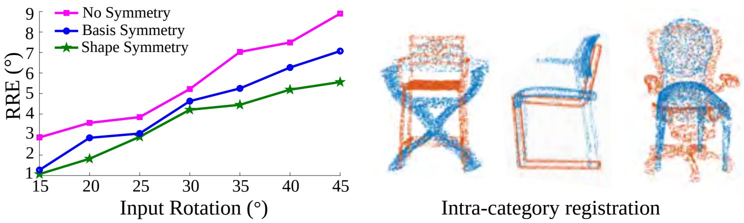


Fig. 5: Left: Relative rotation error for different symmetry modelings. Right: 3 examples of registration between different instances of the same category.

6 Conclusions

This paper investigates automatic discovery of kepoints in 3D misaligned point clouds that are consistent over inter-subject shape variations and intra-subject deformations in a category. We find that this can be solved, with unsupervised learning, by modeling keypoints with non-rigidity, based on symmetric linear basis shapes. Additionally, the proposed category-specific keypoints have one-to-one ordered correspondences and semantic consistency. Applications for the learned keypoints include registration, recognition, generation, shape completion and many more. Our experiments showed that high quality keypoints can be obtained using the proposed methods and that the method can be extended to complex non-rigid deformations. Future work could focus on better modeling complex deformations with non-linear approaches.

Acknowledgements

This research was funded by the EU Horizon 2020 research and innovation program under grant agreement No. 820434. This work was also supported by Project RTI2018-096903-B-I00 (AEI/FEDER, UE) and Regional Council of Bourgogne Franche-Comté (2017-9201AAO048S01342).

References

1. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60**(2) (2004) 91–110
2. Tola, E., Lepetit, V., Fua, P.: Daisy: An efficient dense descriptor applied to wide-baseline stereo. IEEE transactions on pattern analysis and machine intelligence **32**(5) (2009) 815–830
3. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). Computer Vision and Image Understanding **110**(3) (2008) 346 – 359
4. Novotny, D., Ravi, N., Graham, B., Neverova, N., Vedaldi, A.: C3dpo: Canonical 3d pose networks for non-rigid structure from motion. In: Proceedings of the IEEE International Conference on Computer Vision. (2019) 7688–7697
5. Dai, Y., Li, H., He, M.: A simple prior-free method for non-rigid structure-from-motion factorization. In: CVPR. (2012)
6. Snavely, N., Seitz, S.M., Szeliski, R.: Modeling the world from internet photo collections. Int. J. Comput. Vision **80**(2) (2007) 189–210
7. Yew, Z.J., Lee, G.H.: 3dfeat-net: Weakly supervised local 3d features for point cloud registration. In: European Conference on Computer Vision, Springer (2018) 630–646
8. Kneip, L., Li, H., Seo, Y.: Upnp: An optimal o(n) solution to the absolute pose problem with universal applicability. In Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T., eds.: Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I. (2014)
9. Luong, Q.T., Faugeras, O.: The fundamental matrix: theory, algorithms, and stability analysis. International Journal of Computer Vision **17** (1995) 43–75
10. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: SMPL: A skinned multi-person linear model. ACM Trans. Graphics (Proc. SIGGRAPH Asia) **34**(6) (2015) 248:1–248:16
11. Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A.: Real-time human pose recognition in parts from single depth images. In: CVPR 2011, Ieee (2011) 1297–1304
12. Moreno-Noguer, F.: 3d human pose estimation from a single image via distance matrix regression. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 2823–2832
13. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2d pose estimation using part affinity fields. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2017) 7291–7299
14. Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., Black, M.J.: Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In: European Conference on Computer Vision, Springer (2016) 561–578
15. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. (2017) 2961–2969

16. Sattler, T., Leibe, B., Kobbelt, L.: Fast image-based localization using direct 2d-to-3d matching. In: 2011 International Conference on Computer Vision, IEEE (2011) 667–674
17. Tang, H., Xu, D., Liu, G., Wang, W., Sebe, N., Yan, Y.: Cycle in cycle generative adversarial networks for keypoint-guided image generation. In: Proceedings of the 27th ACM International Conference on Multimedia. (2019) 2052–2060
18. Zafeiriou, S., Chrysos, G.G., Roussos, A., Ververas, E., Deng, J., Trigeorgis, G.: The 3d menpo facial landmark tracking challenge. In: Proceedings of the IEEE International Conference on Computer Vision Workshops. (2017) 2503–2511
19. Huang, S., Gong, M., Tao, D.: A coarse-fine network for keypoint localization. In: Proceedings of the IEEE International Conference on Computer Vision. (2017) 3028–3037
20. Pavlakos, G., Zhou, X., Chan, A., Derpanis, K.G., Daniilidis, K.: 6-dof object pose from semantic keypoints. In: ICRA. (2017)
21. Zhang, Z., Luo, P., Loy, C.C., Tang, X.: Facial landmark detection by deep multi-task learning. In: European conference on computer vision, Springer (2014) 94–108
22. Dong, X., Yan, Y., Ouyang, W., Yang, Y.: Style aggregated network for facial landmark detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2018) 379–388
23. Yu, X., Zhou, F., Chandraker, M.: Deep deformation network for object landmark localization. In: European Conference on Computer Vision, Springer (2016) 52–70
24. Alahi, A., Ortiz, R., Vandergheynst, P.: Freak: Fast retina keypoint. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, Ieee (2012) 510–517
25. Li, Y.: A novel fast retina keypoint extraction algorithm for multispectral images using geometric algebra. IEEE Access **7** (2019) 167895–167903
26. Li, J., Lee, G.H.: Usip: Unsupervised stable interest point detection from 3d point clouds. In: Proceedings of the IEEE International Conference on Computer Vision. (2019) 361–370
27. Suwajanakorn, S., Snavely, N., Tompson, J.J., Norouzi, M.: Discovery of latent 3d keypoints via end-to-end geometric reasoning. In: Advances in Neural Information Processing Systems. (2018) 2059–2070
28. Wu, J., Xue, T., Lim, J.J., Tian, Y., Tenenbaum, J.B., Torralba, A., Freeman, W.T.: Single image 3d interpreter network. In: European Conference on Computer Vision, Springer (2016) 365–382
29. Yang, H., Carlone, L.: In perfect shape: Certifiably optimal 3d shape reconstruction from 2d landmarks. arXiv preprint arXiv:1911.11924 (2019)
30. Hejrati, M., Ramanan, D.: Analyzing 3d objects in cluttered images. In: Advances in Neural Information Processing Systems. (2012) 593–601
31. Wang, C., Wang, Y., Lin, Z., Yuille, A.L., Gao, W.: Robust estimation of 3d human poses from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2014) 2361–2368
32. Persad, R.A., Armenakis, C.: Automatic 3d surface co-registration using keypoint matching. Photogrammetric engineering & remote sensing **83**(2) (2017) 137–151
33. Mitra, N.J., Wand, M., Zhang, H., Cohen-Or, D., Kim, V., Huang, Q.X.: Structure-aware shape processing. In: ACM SIGGRAPH 2014 Courses. (2014) 1–21
34. Reed, M.P.: Modeling body shape from surface landmark configurations. In: International Conference on Digital Human Modeling and Applications in Health, Safety, Ergonomics and Risk Management, Springer (2013) 376–383
35. Creusot, C., Pears, N., Austin, J.: 3d landmark model discovery from a registered set of organic shapes. In: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, IEEE (2012) 57–64

36. Sridhar, S., Rempe, D., Valentin, J., Sofien, B., Guibas, L.J.: Multiview aggregation for learning category-specific shape reconstruction. In: Advances in Neural Information Processing Systems. (2019) 2348–2359
37. Gao, Y., Yuille, A.L.: Symmetric non-rigid structure from motion for category-specific object structure estimation. In: European Conference on Computer Vision, Springer (2016) 408–424
38. Bregler, C., Hertzmann, A., Biermann, H.: Recovering non-rigid 3D shape from image streams. In: CVPR. (2000)
39. Torresani, L., Hertzmann, A., Bregler, C.: Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors. IEEE Trans. Pattern Anal. Mach. Intell. **30**(5) (2008) 878–892
40. Akhter, I., Sheikh, Y., Khan, S., Kanade, T.: Nonrigid structure from motion in trajectory space. In: NIPS. (2008)
41. Taylor, J., Jepson, A.D., Kutulakos, K.N.: Non-rigid structure from locally-rigid motion. In: CVPR. (2010)
42. Parashar, S., Pizarro, D., Bartoli, A.: Isometric non-rigid shape-from-motion in linear time. In: CVPR. (2016)
43. Kong, C., Lucey, S.: Deep non-rigid structure from motion. In: Proceedings of the IEEE International Conference on Computer Vision. (2019) 1558–1567
44. Wang, H., Sridhar, S., Huang, J., Valentin, J., Song, S., Guibas, L.J.: Normalized object coordinate space for category-level 6d object pose and size estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2019) 2642–2651
45. Wu, S., Rupprecht, C., Vedaldi, A.: Unsupervised learning of probably symmetric deformable 3d objects from images in the wild. In: CVPR. (2020)
46. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: CVPR. (2017)
47. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: Advances in neural information processing systems. (2017)
48. Verma, N., Boyer, E., Verbeek, J.: Feastnet: Feature-steered graph convolutions for 3d shape analysis. In: CVPR. (2018)
49. Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., Shah, R.: Signature verification using a" siamese" time delay neural network. In: Advances in neural information processing systems. (1994) 737–744
50. Tomasi, C., Kanade, T.: Shape and motion from image streams under orthography: a factorization method. International journal of computer vision **9**(2) (1992) 137–154
51. Yi, L., Kim, V.G., Ceylan, D., Shen, I.C., Yan, M., Su, H., Lu, C., Huang, Q., Sheffer, A., Guibas, L.: A scalable active framework for region annotation in 3d shape collections. ACM Transactions on Graphics (TOG) **35**(6) (2016) 1–12
52. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: CVPR. (2015) 1912–1920
53. Li, J., Chen, B.M., Hee Lee, G.: So-net: Self-organizing network for point cloud analysis. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2018) 9397–9406
54. Fan, H., Su, H., Guibas, L.J.: A point set generation network for 3d object reconstruction from a single image. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017) 605–613

55. Besl, P.J., McKay, N.D.: Method for registration of 3-d shapes. In: Sensor fusion IV: control paradigms and data structures. Volume 1611., International Society for Optics and Photonics (1992) 586–606

56. Bogo, F., Romero, J., Pons-Moll, G., Black, M.J.: Dynamic faust: Registering human bodies in motion. In: CVPR. (2017) 6233–6242

57. Gerig, T., Morel-Forster, A., Blumer, C., Egger, B., Luthi, M., Schönborn, S., Vetter, T.: Morphable face models-an open framework. In: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), IEEE (2018) 75–82

58. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style, high-performance deep learning library. In: NIPS. (2019)

59. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)

60. Song, S., Lichtenberg, S.P., Xiao, J.: Sun rgb-d: A rgb-d scene understanding benchmark suite. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2015) 567–576

Supplementary Material

S1 Symmetry

S1.1 Symmetric deformation space.

Proof. The two linear spaces due to the two basis $\mathcal{B}_{\mathcal{C}\frac{1}{2}}$ and $\mathcal{B}'_{\mathcal{C}\frac{1}{2}}$ are symmetric by Definition 1 as $\mathcal{B}_{\mathcal{C}\frac{1}{2}}$ is symmetric to $\mathcal{B}'_{\mathcal{C}\frac{1}{2}}$ for any $K \in \mathbb{Z}$. Let $\mathsf{c}_i \in \mathcal{L}$ and $\mathsf{c}'_j \in \mathcal{L}'$ represent the respective half coefficients for any two shape instances i and j, where $\mathcal{L}$ and $\mathcal{L}'$ defines the spaces of the predicted half coefficient vectors. Consequently, the actual deformation spaces are symmetric to one another if $\mathcal{L}$ and $\mathcal{L}'$ are equal. We define $p : p(\mathsf{c}_i)$ as the probability distribution of c_i and $q : q(\mathsf{c}'_j)$ as the probability distribution of c'_j. If p and q come from the same distribution, we approach $p = q$. Then we have:

$$\begin{aligned}
&\text{if } \mathsf{c}_i = \mathsf{c}'_j,\\
&\text{either, } p(\mathsf{c}_i) = q(\mathsf{c}'_j) = 0,\\
&\text{or, } p(\mathsf{c}_i) > 0 \text{ and } q(\mathsf{c}'_j) > 0\\
&\text{for all, } \mathsf{c}_i \in \mathcal{L}, \mathsf{c}'_j \in \mathcal{L}'.
\end{aligned} \tag{S1}$$

Condition (S1) guarantees that $\mathcal{L} = \mathcal{L}'$ and thus we obtain a symmetric deformation space. □

Note that for condition (S1) to be true, we do not require the two distributions to be equal, however, it is sufficient and desirable to have so. Therefore, Proposition 1 in the main text highlights such sufficient and desirable case. It is particularly meaningful when we are learning to predict the coefficients through stochastic methods such as a neural network training. In our network architecture indeed one can expect the distributions of these two vectors to be similar given the data exhibits such a symmetric deformation space, since the prediction branches of c_i and c'_i are very similar. Alternatively, one may also try to enforce the condition using a KL divergence loss.

S1.2 Symmetry Plane Parametrization.

As mentioned in Sec. 5.3 in the main paper, we observe that handling misaligned data with unsupervised methods can lead to some rotation ambiguities. More specifically, we observe that different combination of basis shapes can result in different alignments.

As we show in Fig. 5 in the text, predicting the symmetry plane of the object category allows to have more control over the predicted instance poses. We came up with the idea of learning an additional common parameter, $\mathsf{R}_{\mathcal{C}}$, which is directly related to the symmetry plane. By adding this category-specific parameter, the network learns a common rotation for all the objects in the

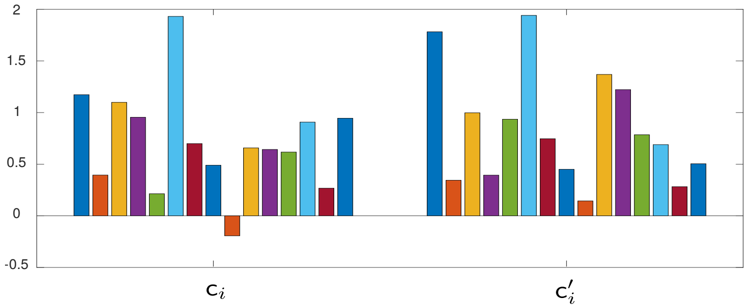


Fig. S1: **Coefficients distribution**. Mean values of c_i components (left) and c'_i components (right) for the Dynamic FAUST [56]. The mean of the variances for the different components are: c_i : 0.54, c'_i : 0.50. The figure shows that the network learns similar distribution for the coefficients c_i and c'_i.

category. As a consequence, the instance-wise rotation, R_i, can be thought like an offset from the reference basis alignment. Several evaluations confirmed that this strategy helps the learning process, reducing the rotation ambiguities.

S2 Additional Experiments

S2.1 Keypoints correspondence

We provide a complete overview for all the object categories evaluated regarding the keypoints correspondences across instances in Fig. S2. This demonstrates the ability of our model to capture and model the inter-subject shape variations and intra-subject deformations in a category.

S2.2 Segmentation Label Transfer

As demonstrated in Sec. 5.2 in the main paper, our predicted keypoints correspond to semantically meaningful locations. Therefore, here we explore the utility of the proposed category-specific keypoints for the segmentation label transfer task. In this experiment, for every point in the original shape $\mathsf{s}_{ij} \in \mathsf{S}_i$, we find its closest category-specific keypoint $\mathsf{p}_{ik} \in \mathsf{P}_i$, and transfer the corresponding semantic label to it. We assume the keypoints labels are known and correspond to those in Fig. 4 in the paper.

Some qualitative results are shown in Fig. S3. Our method achieves full correspondence between instances, therefore avoiding placing keypoints in less representative parts. An example is the engine, in grey, in the case of airplanes. This is reflected in the label transfer since there is no distinction of these parts. Besides that, only with eight keypoints in the example, we achieve reasonable results, close to the ground truth data.

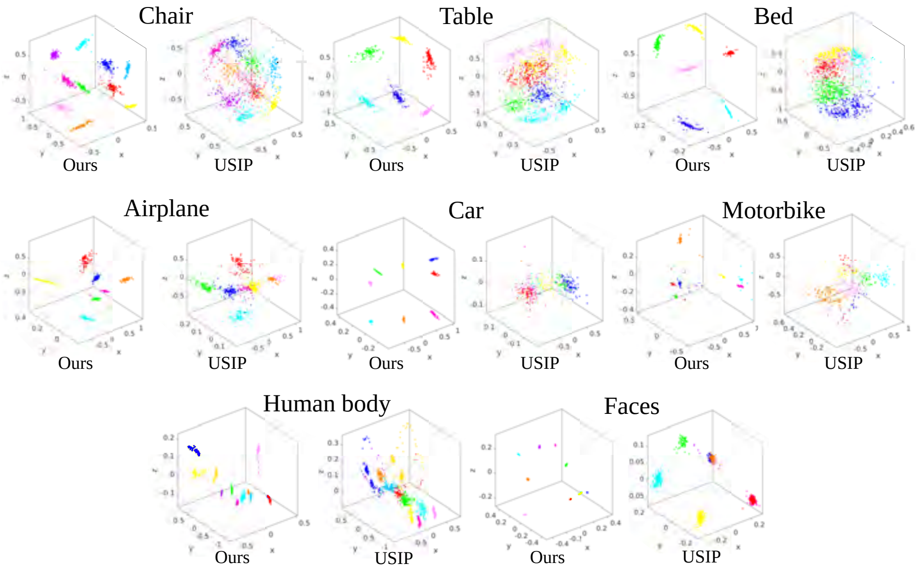


Fig. S2: **Keypoints correspondence across instances**. We cluster the keypoints predicted for all the instances of a category to show their geometric consistency. Note how our keypoints get neatly clustered creating a general 3D shape template.

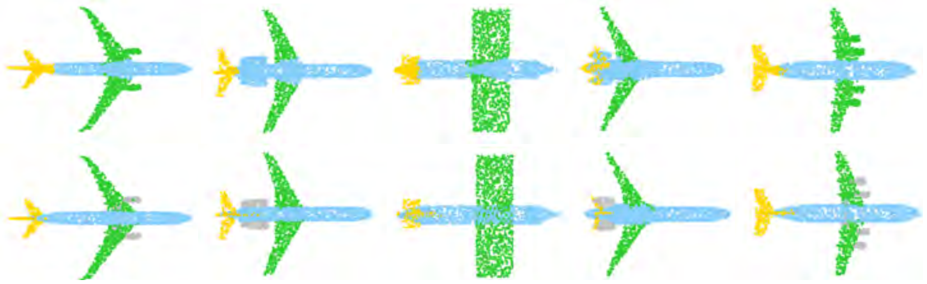


Fig. S3: **First row:** results of performing semantic label transfer with our keypoints. **Second row:** ground truth. This is evaluated in ShapeNet part dataset [51] using eight keypoints for the label transfer.

S2.3 Real Data

In this section, we show the performance of our method for real data in Fig. S4. For this experiment, the network is trained on the chair category from the ModelNet10 dataset [52] and tested on real chairs from the SUNRGBD dataset [60]. To generate the real data dataset from [60], we crop the points inside the ground truth 3D bounding boxes provided by the authors. Real data entail additional challenges. This is not only because shapes appear incomplete and noisy, but also because other objects may cause occlusions, e.g. part of a table occluding a chair. As illustrated in Fig. S4, even though real data is fairly challenging, our network can still produce corresponding meaningful keypoints.

Being able to generalize to previously unseen real objects as demonstrated in Fig. S4 is crucial and really useful for many tasks such as guide for shape completion or shape generation.

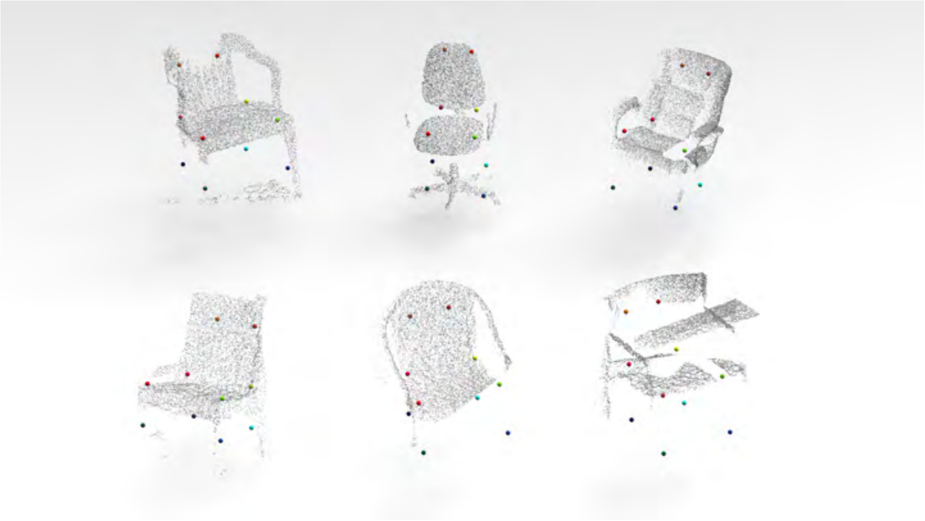


Fig. S4: Results in real chairs from SUNRGBD dataset [60] training with CAD chairs from ModelNet10 dataset [52].

S3 Qualitative results

In this section, we provide additional qualitative results on various object categories from the datasets evaluated in the paper; ModelNet10 [52] in Fig. S5, ShapeNet parts [51] in Fig. S6, Dynamic FAUST [56] in Fig. S7 and Basel Face Model 2017 [57] in Fig. S8.

Again, we note that our network predicts corresponding keypoints between instances of the same category and consistently associates the same keypoint with the same semantic part. For instance, for the chair object category, the keypoint colored in pink is always associated with the chair back, the keypoint colored in cyan is associated with the front left leg, etc.

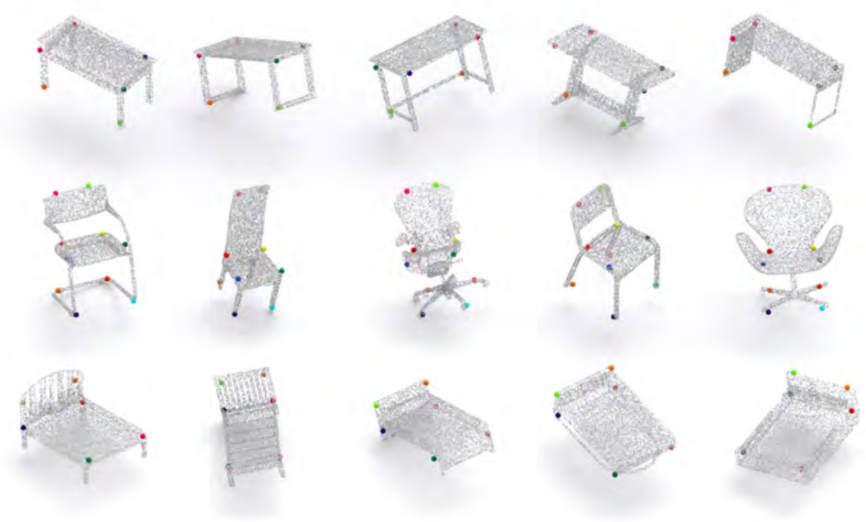


Fig. S5: Qualitative results in table, chair and bed categories from ModelNet10 dataset [52].

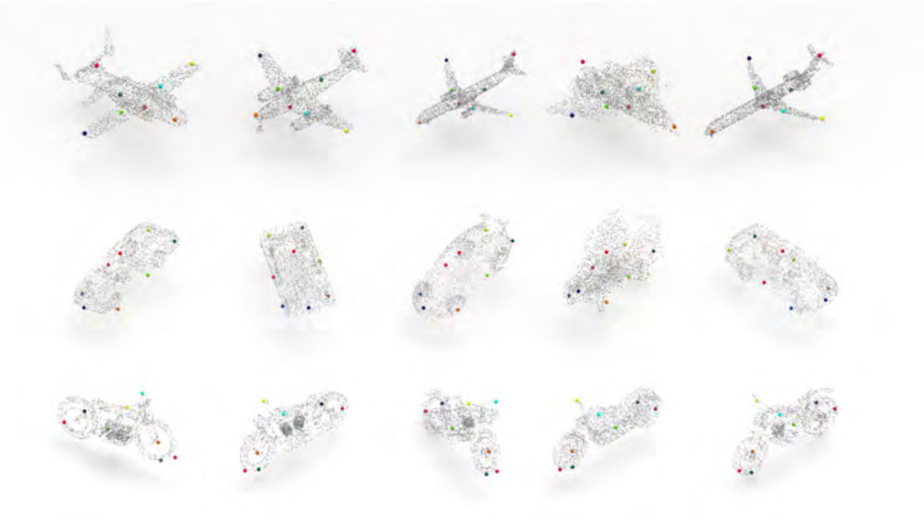


Fig. S6: Qualitative results in airplane, car and motorbike categories from ShapeNet parts dataset [51].

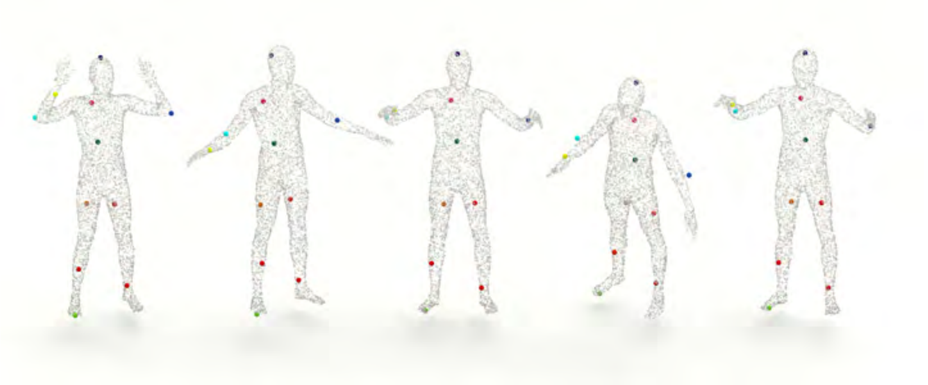


Fig. S7: Qualitative results in human bodies from Dynamic FAUST dataset [56].

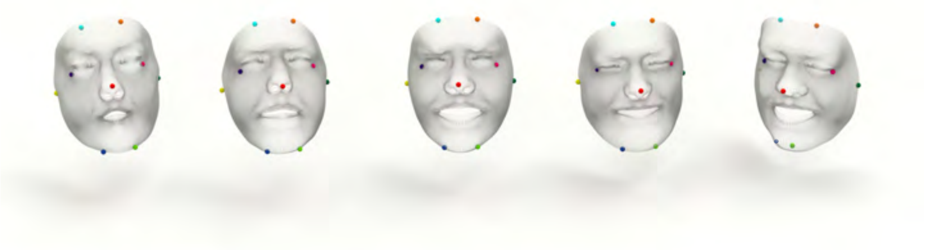


Fig. S8: Qualitative results in faces from Basel Face Model 2017 dataset [57].