Human Correspondence Consensus for 3D Object Semantic Understanding

Yujing Lou[⋆], Yang You[⋆], Chengkun Li[⋆], Zhoujun Cheng, Liangwei Li,
Lizhuang Ma, Weiming Wang, and Cewu Lu[⋆⋆]

Shanghai Jiao Tong University, Shanghai, China
{louyujing,qq456cvb,sjtulck,blankcheng}@sjtu.edu.cn
{liliangwei,ma-lz,wangweiming,lucewu}@sjtu.edu.cn

Abstract. Semantic understanding of 3D objects is crucial in many applications such as object manipulation. However, it is hard to give a universal definition of point-level semantics that everyone would agree on. We observe that people have a consensus on semantic correspondences between two areas from different objects, but are less certain about the exact semantic meaning of each area. Therefore, we argue that by providing human labeled correspondences between different objects from the same category instead of explicit semantic labels, one can recover rich semantic information of an object. In this paper, we introduce a new dataset named **C**orres**P**ondence**Net**. Based on this dataset, we are able to learn dense semantic embeddings with a novel geodesic consistency loss. Accordingly, several state-of-the-art networks are evaluated on this correspondence benchmark. We further show that **C**orres**P**ondence**Net** could not only boost fine-grained understanding of heterogeneous objects but also cross-object registration and partial object matching.

1 Introduction

Object understanding [26, 33, 52] is one of the holy grails in computer vision. Being able to fully understand object semantics is crucial for various applications such as self-driving [8, 35] and attribute transfer [28]. Recently, significant advances have been made in both category-level and instance-level understanding of objects [10, 25]. However, these datasets all require explicit semantic labels with an "oracle" definition and are not suitable for point-level understanding of objects.

One of the key problems with object semantic understanding lies in the ambiguous definitions of semantics. In the past decades, researchers have proposed keypoints [27, 29, 41, 44, 51] and skeletons [4] to explicitly define object semantics. These methods have made success in tasks like human body parsing [22],

[⋆] These authors contributed equally.

[⋆⋆] Cewu Lu is the corresponding author, who is also the member of Qing Yuan Research Institute and MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University, China.

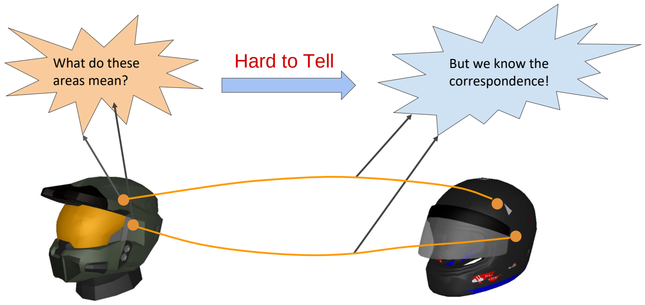

Fig. 1: We observe that it is hard to tell the exact meanings of some areas on an object, while correspondences between different objects are clear.

however, it is hard or even impossible to give consistent definitions of keypoints or skeletons for a general object. Recently, part based representations of objects are also adopted by researchers [10, 50, 33, 21], where an object is decomposed into semantic parts by experts, with a predefined semantic label on each part. The above methods all impose an explicit definition of object semantics, which is inevitably biased or flawed since different people may hold different opinions of what the semantics of an object are.

In this paper, we explore a brand new way to deal with this vagueness in object semantic understanding. Instead of explicitly giving semantic components and labels, we leverage human semantic correspondence consensus between objects to implicitly infer their semantic meanings. This is based on the observation that while it is hard to tell the exact meanings of some sub-object areas, almost everyone would agree on their semantic correspondence across different objects, as shown in Figure 1. Consequently, comprehensive object understanding can be achieved by collecting multiple unambiguous semantic correspondences from a large population.

To that end, we introduce **C**orres**P**ondence**Net** (CPNet): a *diverse* and *high-quality* dataset on top of ShapeNet [10] with *cross-object*, *point-level* 3D semantic correspondence annotations. In this dataset, every annotator gives multiple sets of semantic-consistent points across different intra-class objects, which we call “correspondence sets”, as shown in Figure 2.

Using these correspondence sets, we are able to obtain dense semantic embeddings of an object, perform cross-object semantic registration and partial-to-complete object matching. For dense semantic embedding prediction, a new benchmark with mean Geodesic Error (mGE) is proposed. We leverage a novel geodesic consistency loss to learn this embedding, where points in the same correspondence set are pulled together in the embedding space, while points across different correspondence sets get pushed according to their average geodesic distances. By considering geodesic relationships between different correspondence sets, points with similar semantics are more likely to be grouped together in the embedding space.

In summary, our key contributions are as follows:

- We explore a new way towards 3D object semantic understanding of objects, where explicit definitions are avoided but point-level semantic correspondences across heterogeneous objects are leveraged.
- We introduce *CPNet*, the first human correspondence consensus based dataset for 3D object understanding, which contains 100K+ high-quality semantic-consistent points.
- Based on *CPNet*, we show several applications include dense semantic embedding prediction, cross-object registration and partial-to-complete object matching. We also propose a new benchmark on dense semantic embedding prediction.

The rest of this paper is organized as follows: in Section 2, we discuss some related works; in Section 3, we briefly discuss the importance of human correspondence consensus and introduce our dataset with our annotation methods; in Section 4, we discuss a detailed method on learning dense embeddings based on our dataset; in Section 5, we show some other applications that are naturally driven by our dataset.

2 Related Work

Datasets on Semantic Analysis Big data and deep learning have witnessed several large 2D/3D datasets these years aiming to parse semantic information from objects. In the world of 2D images, SPAIR-71k [32] proposes a large-scale dataset with rich annotations on viewpoints, keypoints and segmentations, which is mainly used for semantic matching between different images. Recently, Ham et al. [19] and Taniai et al. [45] have introduced datasets with groundtruth correspondences. Since then, PF-WILLOW and PF-PASCAL [19] have been used for evaluation in many works. In addition, plenty of datasets on human pose analysis [3, 2] have been proposed recently. These 2D image datasets have their advantages in that they are relatively large and pertain diversity across different scenes and objects.

On the other hand, there exists a rich set of 3D model datasets that try to directly process meshes or point clouds. There are generally two types of them: ones that focus on rigid models and some others that focus on non-rigid models. For rigid model analysis, ShapeNet Core 55 [10] is proposed to help object-level classification while ShapeNet part dataset [50] pushes it one step forward with intra-object part classification. As a followup, PartNet [33] comes up with a much more complete and manually defined hierarchical structures of parts. Alternatively, dataset proposed by Dutagaci et al. [14] focuses on sparse semantic keypoints on objects. For non-rigid (deformable) models, FAUST [7] and TOSCA [9] provide dense correspondence labels for humans and animals, respectively. These methods leverage the clear anatomy structure underlying humans and animals and can be applied to pose transfer, pose synthesis, etc.

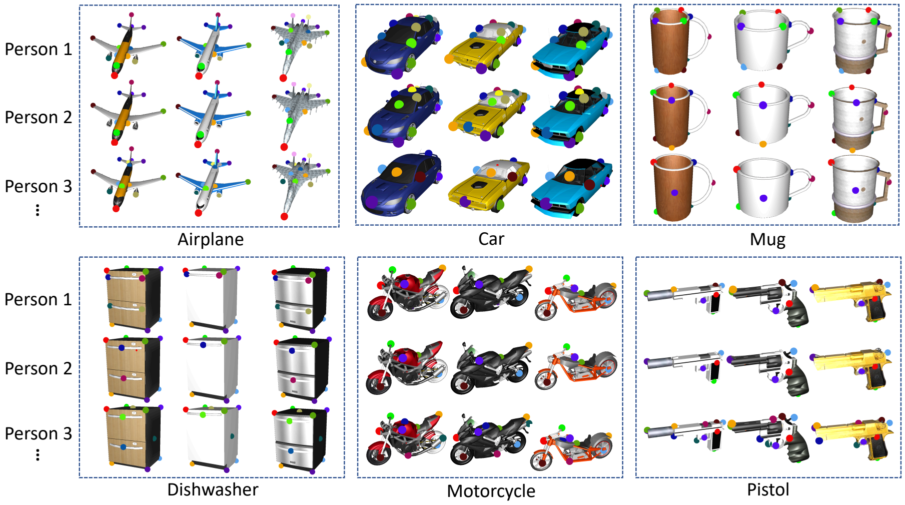

Fig. 2: **CPNet dataset**. Each person annotates multiple sets of corresponding points. Points in the same correspondence set are in the same color. It can be seen that people could have his/her own understanding of semantic points as long as they are consistent across different models within the same category.

Methods on Object Semantic Understanding In the last decade, plenty of methods have been proposed to find semantic correspondences between paired images. Earlier methods like Okutomi et al. [34], Horn et al. [20] and Matas et al. [31] propose to find semantic correspondences within the same scene. Semantic flows like SIFT flow [30] and ProposalFlow [19] further explore to find dense correspondence across different scenes. Kulkarni et al. [24] and Zhou et al. [53] utilize a synthesis 3D model as a medium to enforce semantic cycle-consistency. Florence et al. [16] and Schmidt et al. [42] leverage an unsupervised method to learn consistent dense embeddings across different objects.

When it comes to the domain of 3D shapes, Blanz et al. [6] and Allen et al. [1] are the pioneers on finding 3D correspondence between human faces and bodies. Recently, 3D dense semantic correspondence has been boosted by a variety of deep learning methods. Halimi et al. [18], Groueix et al. [17] and Roufosse et al. [39] propose unsupervised methods on learning dense correspondences between humans and animals. Deep functional dictionaries [43] gives a small dictionary of basis functions for each shape, a dictionary whose span includes the semantic functions provided for that shape. Perhaps, closest to this paper, is the method of Huang et al. [21]. It utilized expert-defined corresponding shape parts to generate a synthetic dense point correspondence dataset and then extracts local descriptors by a neural network. However, it is ambiguous to clearly define object parts while we do not leverage any expert-defined part labels. In addition, their assumption of dense one-to-one correspondence within the same part fails in many common objects.

3 CorresPondenceNet

Understanding semantics from arbitrary objects is of great importance. However, explicitly expressing semantics in a well defined format is extremely hard as the definition of semantics is vague and diverse.

We observe that people are pretty sure about the correspondence between two areas but less sure about what each area means in semantics. As shown in Figure 1, almost everyone would agree on the lined correspondences between two helmets. However, it is hard to tell the exact semantic meanings of the colored areas.

Unlike all previous methods where an explicit definition of keypoints or parts is given, we instead focus on sparse correspondences annotated by humans, based on the assumption that all the corresponding points labeled by the same person share the same semantic meaning.

Therefore, we propose a new dataset called **C**orres**P**ondenceNet (CPNet). CPNet has a collection of 25 categories, 2,334 models based on ShapeNetCore with 104,861 points. Each model is annotated with a number of semantic points from multiple annotators, as shown in Figure 2. Unlike other 2D or 3D keypoint datasets which manually set a keypoint template and let annotators to follow, semantic points in our dataset are not deliberately defined by anyone. The key is that every annotator can have his/her own understanding of semantic points, as long as they are consistent across different models within the same category. In the following subsections, we discuss how we collect models, how we annotate models and annotation types in details. Table 1 gives the detailed statistics of our dataset.

3.1 Dataset Collections

Our dataset is based on ShapeNetCore [10]. ShapeNetCore is a subset of the full ShapeNet dataset with single clean 3D models and manually verified category and alignment annotations. There are 51,300 unique 3D models from 55 common object categories in ShapeNetCore. We select 25 categories that are mostly seen in daily life to build our dataset. To keep a balanced dataset, for each category we keep at most 100 models. For those categories with less than 100 models, all the models are selected.

3.2 Annotation Process

We hire 80 professional annotators in total. Each model is annotated by at least 10 persons to enrich the dataset.

Template Creation For each category, every annotator is allowed to create 1 to 6 templates with his/her own understanding of semantic points. To ensure a broader range of point coverage, we plot a heatmap for each template to indicate which region has been marked often by others. Annotators are encouraged to mark semantic points in those regions that are less explored. As shown in

	Airplane	Bathtub	Bed	Bench	Bottle	Bus	Cap	Car	Chair	Dishwasher	Display	Earphone	Faucet
$\mathrm{N_P}$	5527	6033	6464	5421	4489	6404	949	7938	6140	5343	4509	904	1612
$\mathrm{N_A}$	10	10	10	10	10	10	10	10	10	10	10	10	10
$\mathrm{N_M}$	100	100	100	100	100	100	38	100	100	77	100	58	100
$\mathrm{C_{min}}$	35	40	40	30	41	50	20	64	50	60	20	14	10
$\mathrm{C_{med}}$	54	60	60	50	45	64	25	80	70	70	50	15	15
$\mathrm{C_{max}}$	72	96	80	70	46	81	30	82	78	84	51	21	22

	Guitar	Helmet	Knife	Lamp	Laptop	Motorcycle	Mug	Pistol	Rocket	Skateboard	Table	Vessel	All
$\mathrm{N_P}$	2832	1500	2109	1683	2987	3878	7668	3358	2315	3822	4008	5214	104861
$\mathrm{N_A}$	10	10	10	10	10	10	10	10	10	10	10	10	-
$\mathrm{N_M}$	100	95	100	100	100	100	100	100	66	100	100	100	2334
$\mathrm{C_{min}}$	19	27	10	13	20	30	66	17	21	20	39	40	-
$\mathrm{C_{med}}$	30	35	12	15	30	40	77	35	32	40	40	54	-
$\mathrm{C_{max}}$	32	37	15	21	36	40	78	41	49	43	44	56	-

Table 1: **CPNet statistics.** $\mathrm{N_P}$ gives the number of annotated points of each category; $\mathrm{N_A}$ gives the number of annotators for each category; $\mathrm{N_M}$ is the number of models in each category; $\mathrm{C_{min}}$, $\mathrm{C_{med}}$, $\mathrm{C_{max}}$ give minimum, median and maximum number of correspondence sets per instance in each category.

Figure 3(a), red regions indicate frequent annotations while blue means the opposite. Therefore, annotators should avoid red regions in order to get a better coverage.

Templates are then listed to guide the annotations of rest models, so that he/she is able to keep the consistency. Consider an airplane as an example, if one annotator marks the nose as No.1 semantic point, then he/she is supposed to mark all the noses on other airplanes as No.1. It does not matter if another annotator marks the nose as No.2 semantic point, or even neglecting it, as long as the annotator keeps his/her own rules across all the models. For a certain point that does not exist on all the models such as a point of propeller, one can just skip the models without it. The annotator is free to choose any points from his/her perspective.

Each annotator is asked to mark at most 16 semantic points per model. All points are annotated on raw meshes, which is more accurate than those annotated on point clouds. Moreover, it is straightforward to extend these annotations to point clouds by sampling from the mesh while fixing the locations of semantic points.

Handling Symmetries In case of any central/rotational symmetry, we extend our single semantic point $p_{i,j}^{(n)}$ to a single *hyperpoint*, which contains all the points that are centrally/rotationally symmetric. This step is done manually by marking out those symmetric points. During training, *hyperpoint* are treated as normal points. When generating a positive/negative point pair, we randomly sample a point within the *hyperpoint*.

Cross Validation As we mentioned before, we do not define semantic labels. However, this makes strict vetting process impossible. In order to make our

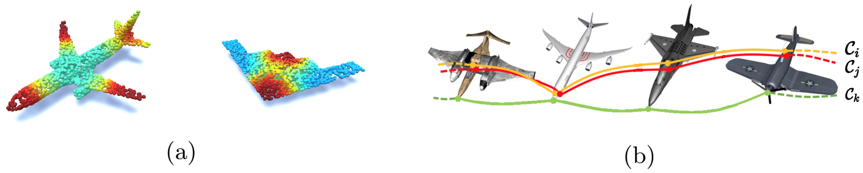

Fig. 3: (a) Example coverage heatmaps. Red regions indicate frequent annotations while blue means the opposite. We encourage annotators to annotate on blue regions. (b) **Correspondence sets across different airplanes.** $\mathcal{C}_i$, $\mathcal{C}_j$ and $\mathcal{C}_k$ denote three semantic correspondence sets respectively.

dataset trustworthy, we introduce a cross-validation process. To be more specific, for each annotated correspondence, we ask at-least ten other annotators to verify if it is reasonable or not. If more than 80% annotators agree it is reasonable, then this correspondence is kept, otherwise it is rejected. The rationale for cross-validation lies in our prior that most people have a consistent common sense on whether a given semantic correspondence exists across different objects.

3.3 Annotation Type

Denote all the models as $\mathbf{M} = \{\mathcal{M}_i\}$, where $\mathcal{M}_i$ represents a single model. Each mdoel $\mathcal{M}_i$ is associated with a set of semantic points $\mathcal{P}_i = \{p_{i,j}^{(n)}\}$ where i, j, n denote the j-th semantic point of the n-th annotator on the i-th model.

In addition, we ask each annotator to give consistent points across different models, so that $p_{i_1,j}^{(n)}$ and $p_{i_2,j}^{(n)}$ have the same semantic meaning. Therefore, we define a set of correspondence sets $\Omega = \{\mathcal{C}_j | j = 1, \cdots, N_\Omega\}$, where each correspondence set $\mathcal{C}_j = \{p_{i,j} | i = 1, \cdots, N_\mathbf{M}\}$ contains all the points with the same semantic label. Note that we dropped the index of the annotator since distinct point correspondence from the same person can be treated the same as those from different persons.

Each annotated point contains attributes about (1) xyz coordinate, (2) color, (3) face index and (4) uv coordinate. By providing these attributes, methods based on either point clouds or meshes can be applied easily.

We thus release four different versions of our correspondence dataset for those who are interested: 1) correspondences without any symmetries; 2) correspondences with only central symmetries; 3) correspondences with only rotational symmetries; 4) correspondences with both central and rotational symmetries.

4 Learning Dense Semantic Embeddings

We now propose a method on learning dense semantic embeddings from human labeled correspondences across various intra-class models.

4.1 Problem Statement

Given a set of 3D models $\mathbf{M} = \{\mathcal{M}_i | i = 1, \cdots, N_\mathbf{M}\}$ and a set of correspondence sets $\Omega = \{\mathcal{C}_j | j = 1, \cdots, N_\Omega\}$ defined in Section 3.3, our goal is to produce a set of pointwise embeddings for each model $\mathcal{M}_i$. The embeddings encode semantic information across different models and points with similar semantics are close in embedding space. We define f as an embedding function, such that $f(p)$ gives the embedding for point p on the model. In practice, we approximate f with a deep neural network and explain how to optimize f as follows.

4.2 Method Details

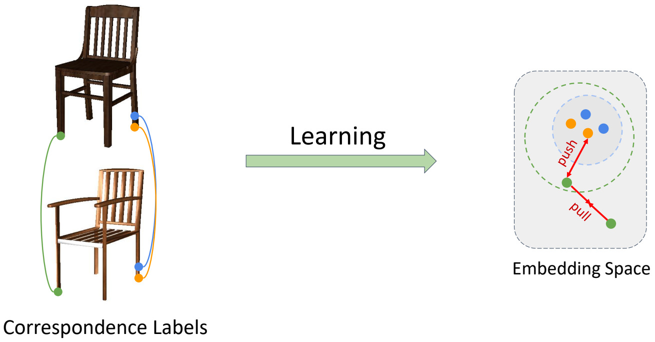

Fig. 4: Given correspondence sets, we pull the points in the same correspondence set and push points from different correspondence sets adaptively, according to their average geodesic distances. The blue and orange correspondence sets are close so that they can stay close in embedding space, while the orange and green ones are far away in average geodesic distance so their embeddings are pushed further from each other.

Pull Loss It is natural to come up with a pull loss since we would like to ensure the semantic consistency within every correspondence set. As illustrated in Figure 3(b), the points with the same color belong to the same correspondence set and reveal similar semantic information. For one specific correspondence $\mathcal{C}_k$ like the green line shown in Figure 3(b), we aim to pull the embedding vectors of the points within it. Any two of points in the same correspondence set form a positive pair. The pairwise embedding distances are then summed over all positive pairs to form our pull loss:

$$L_{pull} = \frac{1}{N_{pos}} \sum_k \sum_{p,q \in \mathcal{C}_k, p \neq q} \|f(p) - f(q)\|_2, \tag{1}$$

where N_{pos} is the number of all possible positive point pairs.

Geodesic Consistency Loss The pull loss in Equation 1 enforces the points in the same correspondence set to have similar embeddings. However, there is a trivial solution where f outputs a constant embedding (e.g. $\mathbf{0}$) for all points, which is a global optimum when minimizing L_{pull} only. Such a trivial solution is due to the ignorance of an important principle: we ought to ensure that those points with distinct semantics to have a large embedding distance. Therefore, a push loss guided by geodesic consistency is proposed to fulfill this goal. We leverage a prior to determine whether two different correspondence sets have distinct semantics: if all pairs of points from these two sets have large geodesic distances on models, they are more likely to reveal different semantic information.

Based on this insight, we design a distance measure $\mathbf{d}$ for a pair of correspondence sets $\mathcal{C}_i$ and $\mathcal{C}_j$:

$$\mathbf{d}(\mathcal{C}_i, \mathcal{C}_j) = \frac{1}{N_{\mathbf{M}}} \sum_{k} \sum_{p,q \in \mathcal{M}_k} \mathbf{d}_{geo}(p, q), \;\; s.t. \; p \in \mathcal{C}_i, q \in \mathcal{C}_j, \tag{2}$$

where $\mathbf{d}_{geo}(p, q)$ is the geodesic distance between point p and q. This distance measure $\mathbf{d}$ represents the average geodesic distance between point pairs from two correspondence sets.

Then, the push loss can be written as,

$$L_{push} = \frac{1}{N_{neg}} \sum_{i \neq j} \sum_{p \in \mathcal{C}_i} \sum_{q \in \mathcal{C}_j} \max\{0, \mathbf{d}(\mathcal{C}_i, \mathcal{C}_j) - \|f(p) - f(q)\|_2\}, \tag{3}$$

where N_{neg} is the number of all possible negative pairs formed by points from different correspondence sets.

In Equation 3, the push loss is only activated when $\|f(p) - f(q)\|_2$ is smaller than $\mathbf{d}(\mathcal{C}_i, \mathcal{C}_j)$. In other words, the larger $\mathbf{d}(\mathcal{C}_i, \mathcal{C}_j)$ is, the further $f(p)$ and $f(q)$ are separated in the embedding space. This is based on the observation that some points in two correspondence sets may have similar semantic information (like the red and orange lines in Figure 3(b)) while some have totally different meanings (like the orange and green lines in Figure 3(b)). Therefore, only for those correspondence sets with a large average geodesic distance, a large distance between their embeddings is expected.

Our final loss is,

$$L = L_{pull} + \lambda L_{push}, \tag{4}$$

where λ is a weight factor. Our method is summarized in Figure 4.

Hard Negative Mining In practice, negative pairs to be pushed are combinatorially more than positive pairs to be pulled, since negative pairs are sampled from different correspondence sets. In such case, we borrow the idea from [12] to utilize hard negative mining. Within each batch, only those negative pairs with smallest embedding distances are taken into consideration, matching the number of positive pairs.

Algorithm 1 mean Geodesic Error calculation

```
Input: model set $\Omega$, an embedding function $f$ to be evaluated
Output: mean Geodesic Error (mGE) $\varepsilon$ of $f$
$\varepsilon = 0$
for $\mathcal{C}_i$ in $\Omega$ do
    for $p$, $q$ in $\mathcal{C}_i$ do
        $x = \arg\min_{x \in \mathcal{M}_q} \|f(x) - f(p)\|_2$, where
            $\mathcal{M}_q$ denotes the model that point $q$ lies on.
        $\varepsilon = \varepsilon + \mathbf{d}_{geo}(q, x)$
    end for
end for
$\varepsilon = \frac{\varepsilon}{N_{\Omega} N_{\mathbf{M}}^2}$
```

4.3 Mean Geodesic Error

Since we are dealing with a new dense embedding prediction task, existing metrics on classification or part segmentation can not benchmark it well. Therefore, we introduce mean Geodesic Error (mGE), a new metric on dense correspondence, to evaluate predicted semantic embeddings. Unlike mean Euclidean Error that is used in Huang at el. [21], geodesic distance is more suitable for 3d objects as it is restricted to lie on object surfaces. mGE is calculated individually for each category and measures how well the generated embedding vectors fit with annotated correspondence sets. We also provide results for mean Euclidean Error in the supplementary material. Algorithm 1 presents the calculation procedure of mGE for a given embedding function f. Intuitively, for each annotated points on a model, we find their corresponding points that minimize the embedding distance on other models. After that, the geodesic distances between these points and human labeled corresponding points are accumulated. It is easy to verify that if all the embeddings are identical within the same correspondence set but are distinct across different correspondence sets, mGE = 0, which means that the predicted semantic embeddings are consistent with human labels.

4.4 Experiments

In this section, we demonstrate that our proposed method can effectively learn point-wise dense embeddings from human labeled correspondences. We evaluate the embeddings with mGE error. Seven state-of-the-art neural network backbones are benchmarked. These backbones are point cloud [37, 38, 49], graph [48, 13] and voxel [11, 46] based neural networks. We additionally compare our approach, which is based on implicit correspondences, with that based on explicit part-level supervision.

Evaluation and Results We split our dataset into train (70%), validation (15%) and test (15%) set. Train and validation sets are used during training and all the

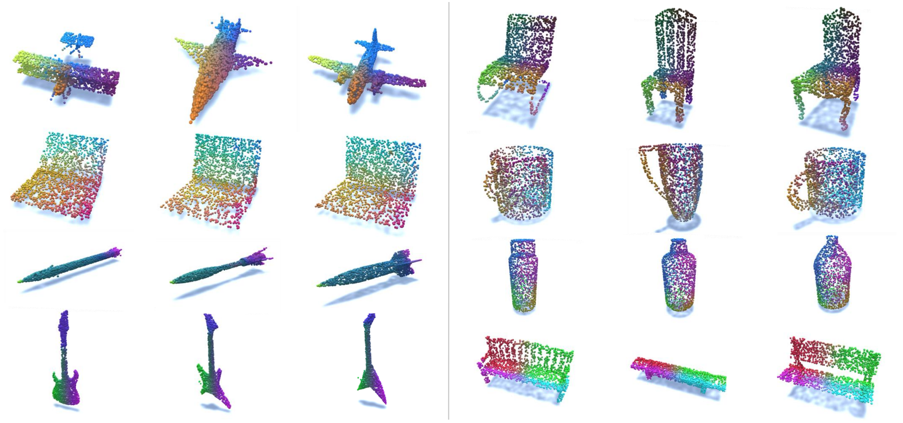

Fig. 5: **Predicted semantic embeddings for PointConv.** Same colors indicate similar embeddings.

results are reported on the test set. We use ADAM optimizer [23] with initial learning rate $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and batch size 4. The learning rate is multiplied by 0.9 every 10 epochs and the hyperparameter λ in Equation 4 is set to 1. The output point embedding vector is 128-dimensional for all neural networks.

Table 2 gives mGE of all the compared architectures. SHOT fails to predict correct semantic correspondences across objects, whose performance is just slightly better than random point embeddings. The reason is that SHOT only considers local geometric properties, without aggregation of the global structure and semantic information. In contrast, all deep learning based methods using our geodesic consistency loss achieve much smaller mGE. Among them, DGCNN, PointNet, RS-Net and PointConv are relatively superior to the other nets on extracting semantic correspondence information. The visualization of learned embeddings by PointConv is shown in Figure 5. From Figure 5, we can see that consistent pointwise embeddings are generated across heterogeneous objects. We get reasonable dense embeddings of all points on objects by fitting sparse correspondence annotations.

Comparison to Part-level Supervision To further illustrate the advantage of our proposed semantic correspondence sets, we compare our method with that supervised by part-level annotations.

We train a PointNet using correspondence labels and part labels respectively. For PointNet trained on part labels, we use the same experiment settings for part segmentation as the original paper [37] and extract features from the last but one layer as point embeddings. Then given a point on a source model, we use embeddings to find its corresponding point on the target model and results are shown in Figure 6. Qualitatively, we can see that when trained on our correspondence

	Airplane	Bathtub	Bed	Bench	Bottle	Bus	Cap	Car	Chair	Dishwasher	Display	Earphone	Faucet
PointNet	0.063	0.141	**0.078**	0.066	0.090	**0.055**	0.093	0.070	**0.088**	**0.103**	**0.071**	0.151	0.163
PointNet++	0.053	0.170	0.118	0.071	0.138	0.118	0.123	0.075	0.114	0.148	0.112	0.122	0.179
RS-Net	0.052	0.153	0.121	0.091	**0.082**	0.059	0.101	**0.064**	0.097	0.145	0.081	0.115	0.167
PointConv	0.053	0.133	0.128	0.072	0.100	0.076	0.121	0.079	0.126	0.144	0.085	0.103	0.161
DGCNN	**0.046**	**0.118**	0.125	**0.058**	0.088	0.060	**0.085**	0.073	0.106	0.116	0.086	**0.091**	**0.143**
GraphCNN	0.069	0.153	0.126	0.089	0.166	0.099	0.122	0.112	0.147	0.157	0.132	0.136	0.163
Minkowski	0.085	0.149	0.150	0.112	0.147	0.113	0.155	0.102	0.162	0.177	0.179	0.116	0.185
SHOT	0.230	0.485	0.580	0.568	0.380	0.410	0.340	0.386	0.508	0.515	0.430	0.495	0.258
Random	0.308	0.492	0.564	0.544	0.431	0.404	0.484	0.401	0.515	0.507	0.483	0.599	0.355

	Guitar	Helmet	Knife	Lamp	Laptop	Motorcycle	Mug	Pistol	Rocket	Skateboard	Table	Vessel	Average
PointNet	0.066	0.169	0.066	**0.221**	0.163	0.085	0.072	**0.091**	0.151	**0.059**	**0.042**	**0.101**	0.101
PointNet++	0.083	0.180	0.079	0.226	0.182	0.089	0.106	0.117	0.153	0.095	0.093	0.140	0.123
RS-Net	**0.061**	0.166	0.064	0.243	0.170	**0.074**	**0.063**	0.098	0.133	0.072	0.103	0.120	0.108
PointConv	0.082	0.177	0.089	0.237	**0.116**	0.089	0.094	0.107	**0.124**	0.061	0.076	0.128	0.110
DGCNN	0.064	**0.160**	**0.052**	**0.221**	0.131	0.085	0.095	0.099	0.127	**0.059**	0.064	0.118	**0.099**
GraphCNN	0.115	0.178	0.117	0.245	0.160	0.121	0.132	0.115	0.170	0.089	0.098	0.191	0.136
Minkowski	0.123	0.195	0.100	0.252	0.203	0.140	0.151	0.126	0.154	0.101	0.112	0.154	0.146
SHOT	0.311	0.389	0.193	0.390	0.551	0.350	0.413	0.343	0.276	0.395	0.606	0.374	0.407
Random	0.329	0.410	0.426	0.452	0.547	0.369	0.488	0.408	0.315	0.396	0.544	0.377	0.446

Table 2: **Mean Geodesic Error (mGE) results**.

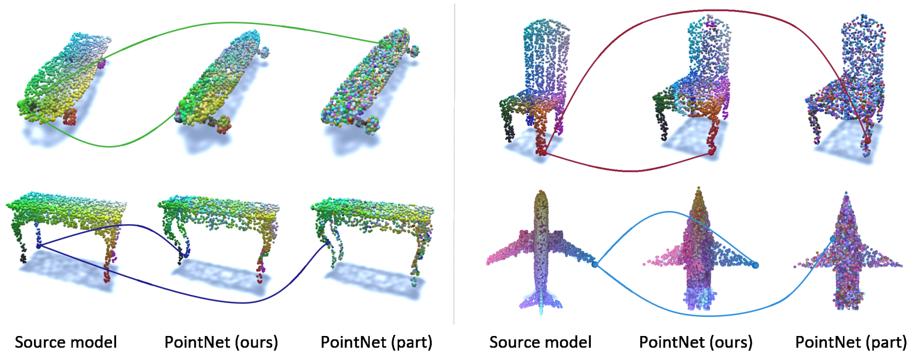

Fig. 6: **Comparison between our method and part-level supervision.** Given a point on the source model, we find its closest point in embedding space on the target model and post-process the founded correspondences with PMF [47] to ensure bijectiveness. The corresponding points are in the same color.

labels, points of the same semantic have similar embeddings while part-level supervision fails to give consistent semantic embeddings across objects. In addition, we compare them quantitatively using mGE, as shown in Table 3. Clearly, PointNet trained on our correspondence labels achieves better performance. On the contrary, with only part-level supervision, points in the same part are hard to be distinguished from each other, resulting in inferior performance. Note that the number of training data for part-level supervision (10240) is seven times more than that for correspondence based supervision (1362).

	Air.	Cap	Car	Chair	Earphone	Guitar	Knife	Lamp
PointNet	**0.063**	**0.093**	**0.070**	**0.088**	0.151	**0.066**	0.066	**0.221**
PointNet(Part)	0.166	0.271	0.245	0.227	**0.140**	0.083	**0.065**	0.282

	Laptop	Motor	Mug	Pistol	Rocket	Skate.	Table	Average
PointNet	0.163	**0.085**	**0.072**	**0.091**	**0.151**	**0.059**	**0.042**	**0.099**
PointNet(Part)	**0.112**	0.222	0.182	0.189	0.228	0.322	0.282	0.201

Table 3: **Comparison of the results trained on human labeled correspondences and part annotations using PointNet.**

5 Other Applications

5.1 Cross-Object Registration

We demonstrate cross-object registration at category-level could benefit from the learnt embeddings, as illustrated in Figure 7.

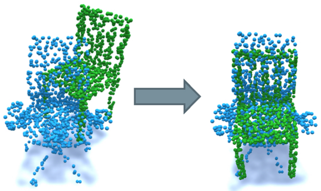

Fig. 7: **Cross-object registration visualization.**

	Chair	Airplane	Mug	Pistol
FPFH	77.1°/0.285	41.3°/0.163	25.9°/0.14	9.1°/0.095
SHOT	72.0°/0.262	44.8°/0.172	91.3°/0.33	21.2°/0.121
Part	20.1°/**0.155**	**24.4°/0.147**	80.6°/0.35	67.75°/0.306
Ours	**14.6°**/0.157	37.0°/0.225	**17.1°/0.137**	**5.3°/0.089**

Table 4: **Comparison of cross-object registration.**

Experiment Settings Given two shapes S and S' in the same category with aligned orientations and overlapped centroids, we randomly rotate and translate S'. Both shapes are normalized in a unit sphere. The objective is to find a rotation matrix $\mathbf{R} \in \mathbb{R}^{3\times3}$ and a translation vector $\mathbf{t} \in \mathbb{R}^3$ that best align S to S'. Initial rotation and translation on S' are seen as ground truth. We use RANSAC[15] with embeddings for global registration and ICP[5] to refine. As a comparison, we also evaluate registration results from SHOT, FPFH[40] and PointNet part segmentation embeddings. 840 shape pairs from 4 common categories of CPNet test set are evaluated under three levels of perturbation similar to [36]: $Easy(10°, 0.1)$, $Medium(20°, 0.3)$, $Hard(45°, 0.5)$. Table 4 gives relative rotational and translational errors. Our embeddings are robust in registration and give reliable semantic correspondences.

5.2 Partial Object Matching

In real applications, occlusion and incompletion of 3D models are pretty common, which makes accurate semantic point matching a tough task.

We conduct experiments to qualitatively show that the learnt embeddings with our method can generalize well to partial objects and thus can be used to find correspondences between partial and complete objects. Given our dataset, we train the network with complete objects and apply the network on their partial counterparts synthetically by removing some parts. Figure 8 shows the embeddings of partial and complete object pairs. Our method predicts reliable semantic embeddings even under severe erosion.

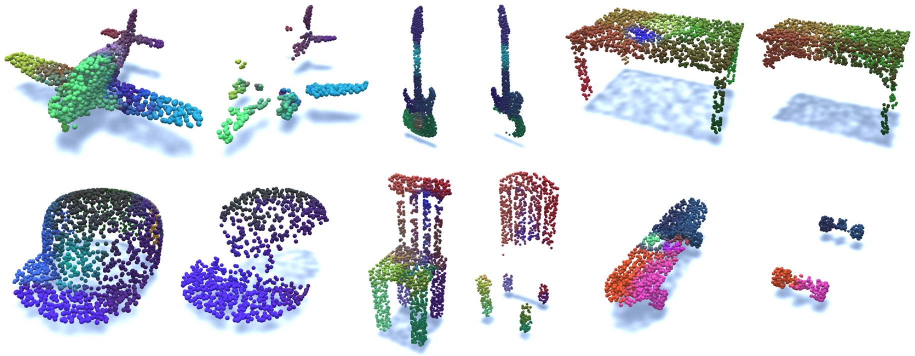

Fig. 8: **Partial object matching**. Each pair includes the complete and partial scans of the different objects within the same category. Same colors indicate same embeddings.

6 Conclusion

In this paper, we explored a new way towards semantic understanding of 3D objects. Instead of explicitly defining semantic labels on an object, we leveraged an observation that while semantic meanings on a single object can be ambiguous and hard to depict, the correspondences of certain points across objects are clear. We thus built a dataset named **C**orres**P**ondence**Net** (CPNet) based on human labeled correspondences, and proposed a method on learning dense semantic embeddings of objects. Mean Geodesic Error is introduced to evaluate our method with various backbones. Some other applications like cross-object registration and partial object matching are also introduced to better illustrate CPNet's potentiality in boosting general object semantic understandings.

7 Acknowledgements

This work is supported in part by the National Key R&D Program of China, No. 2017YFA0700800, National Natural Science Foundation of China under Grants 61772332, SHEITC (2018-RGZN-02046) and Shanghai Qi Zhi Institute.

References

1. Allen, B., Curless, B., Curless, B., Popović, Z.: The space of human body shapes: reconstruction and parameterization from range scans. In: ACM transactions on graphics (TOG). vol. 22, pp. 587–594. ACM (2003)
2. Andriluka, M., Iqbal, U., Ensafutdinov, E., Pishchulin, L., Milan, A., Gall, J., B., S.: PoseTrack: A benchmark for human pose estimation and tracking. In: CVPR (2018)
3. Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2d human pose estimation: New benchmark and state of the art analysis. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2014)
4. Au, O.K.C., Tai, C.L., Chu, H.K., Cohen-Or, D., Lee, T.Y.: Skeleton extraction by mesh contraction. In: ACM transactions on graphics (TOG). vol. 27, p. 44. ACM (2008)
5. Besl, P.J., McKay, N.D.: Method for registration of 3-d shapes. In: Sensor fusion IV: control paradigms and data structures. vol. 1611, pp. 586–606. International Society for Optics and Photonics (1992)
6. Blanz, V., Vetter, T., et al.: A morphable model for the synthesis of 3d faces. In: Siggraph. vol. 99, pp. 187–194 (1999)
7. Bogo, F., Romero, J., Loper, M., Black, M.J.: Faust: Dataset and evaluation for 3d mesh registration. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3794–3801 (2014)
8. Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L.D., Monfort, M., Muller, U., Zhang, J., et al.: End to end learning for self-driving cars. arXiv preprint arXiv:1604.07316 (2016)
9. Bronstein, A.M., Bronstein, M.M., Kimmel, R.: Numerical geometry of non-rigid shapes. Springer Science & Business Media (2008)
10. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: Shapenet: An information-rich 3d model repository. arXiv preprint arXiv:1512.03012 (2015)
11. Choy, C., Gwak, J., Savarese, S.: 4d spatio-temporal convnets: Minkowski convolutional neural networks. arXiv preprint arXiv:1904.08755 (2019)
12. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection (2005)
13. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. In: Advances in neural information processing systems. pp. 3844–3852 (2016)
14. Dutagaci, H., Cheung, C.P., Godil, A.: Evaluation of 3d interest point detection techniques via human-generated ground truth. The Visual Computer **28**(9), 901–917 (2012)
15. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM **24**(6), 381–395 (1981)
16. Florence, P.R., Manuelli, L., Tedrake, R.: Dense object nets: Learning dense visual object descriptors by and for robotic manipulation. arXiv preprint arXiv:1806.08756 (2018)
17. Groueix, T., Fisher, M., Kim, V.G., Russell, B.C., Aubry, M.: 3d-coded: 3d correspondences by deep deformation. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 230–246 (2018)
18. Halimi, O., Litany, O., Rodolà, E., Bronstein, A., Kimmel, R.: Self-supervised learning of dense shape correspondence. arXiv preprint arXiv:1812.02415 (2018)

19. Ham, B., Cho, M., Schmid, C., Ponce, J.: Proposal flow: Semantic correspondences from object proposals. IEEE transactions on pattern analysis and machine intelligence **40**(7), 1711–1725 (2017)
20. Horn, B.K., Schunck, B.G.: " determining optical flow": A retrospective (1993)
21. Huang, H., Kalogerakis, E., Chaudhuri, S., Ceylan, D., Kim, V.G., Yumer, E.: Learning local shape descriptors from part correspondences with multiview convolutional networks. ACM Trans. Graph. **37**(1) (Nov 2017). https://doi.org/10.1145/3137609, https://doi.org/10.1145/3137609
22. Kalayeh, M.M., Basaran, E., Gökmen, M., Kamasak, M.E., Shah, M.: Human semantic parsing for person re-identification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1062–1071 (2018)
23. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
24. Kulkarni, N., Gupta, A., Tulsiani, S.: Canonical surface mapping via geometric cycle consistency. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2202–2211 (2019)
25. Kundu, A., Li, Y., Rehg, J.M.: 3d-rcnn: Instance-level 3d object reconstruction via render-and-compare. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3559–3568 (2018)
26. Leng, B., Liu, Y., Yu, K., Zhang, X., Xiong, Z.: 3d object understanding with 3d convolutional neural networks. Information sciences **366**, 188–201 (2016)
27. Leutenegger, S., Chli, M., Siegwart, R.: Brisk: Binary robust invariant scalable keypoints. In: 2011 IEEE international conference on computer vision (ICCV). pp. 2548–2555. Ieee (2011)
28. Liao, J., Yao, Y., Yuan, L., Hua, G., Kang, S.B.: Visual attribute transfer through deep image analogy. arXiv preprint arXiv:1705.01088 (2017)
29. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
30. Liu, C., Yuen, J., Torralba, A.: Sift flow: Dense correspondence across scenes and its applications. IEEE transactions on pattern analysis and machine intelligence **33**(5), 978–994 (2010)
31. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide-baseline stereo from maximally stable extremal regions. Image and vision computing **22**(10), 761–767 (2004)
32. Min, J., Lee, J., Ponce, J., Cho, M.: Spair-71k: A large-scale benchmark for semantic correspondence. arXiv preprint arXiv:1908.10543 (2019)
33. Mo, K., Zhu, S., Chang, A.X., Yi, L., Tripathi, S., Guibas, L.J., Su, H.: Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 909–918 (2019)
34. Okutomi, M., Kanade, T.: A multiple-baseline stereo. IEEE Transactions on Pattern Analysis & Machine Intelligence (4), 353–363 (1993)
35. Paden, B., Čáp, M., Yong, S.Z., Yershov, D., Frazzoli, E.: A survey of motion planning and control techniques for self-driving urban vehicles. IEEE Transactions on intelligent vehicles **1**(1), 33–55 (2016)
36. Pomerleau, F., Colas, F., Siegwart, R., et al.: A review of point cloud registration algorithms for mobile robotics. Foundations and Trends® in Robotics **4**(1), 1–104 (2015)

37. 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)
38. 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. pp. 5099–5108 (2017)
39. Roufosse, J.M., Sharma, A., Ovsjanikov, M.: Unsupervised deep learning for structured shape matching. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1617–1627 (2019)
40. Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms (fpfh) for 3d registration. In: 2009 IEEE international conference on robotics and automation. pp. 3212–3217. IEEE (2009)
41. Salti, S., Tombari, F., Spezialetti, R., Di Stefano, L.: Learning a descriptor-specific 3d keypoint detector. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2318–2326 (2015)
42. Schmidt, T., Newcombe, R., Fox, D.: Self-supervised visual descriptor learning for dense correspondence. IEEE Robotics and Automation Letters **2**(2), 420–427 (2016)
43. Sung, M., Su, H., Yu, R., Guibas, L.J.: Deep functional dictionaries: Learning consistent semantic structures on 3d models from functions. In: Advances in Neural Information Processing Systems. pp. 485–495 (2018)
44. 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. pp. 2059–2070 (2018)
45. Taniai, T., Sinha, S.N., Sato, Y.: Joint recovery of dense correspondence and cosegmentation in two images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4246–4255 (2016)
46. Tombari, F., Salti, S., Di Stefano, L.: Unique signatures of histograms for local surface description. In: European conference on computer vision. pp. 356–369. Springer (2010)
47. Vestner, M., Litman, R., Rodolà, E., Bronstein, A., Cremers, D.: Product manifold filter: Non-rigid shape correspondence via kernel density estimation in the product space. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3327–3336 (2017)
48. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph cnn for learning on point clouds. ACM Transactions on Graphics (TOG) **38**(5), 146 (2019)
49. Wu, W., Qi, Z., Li, F.: Pointconv: Deep convolutional networks on 3d point clouds. CoRR **abs/1811.07246** (2018), http://arxiv.org/abs/1811.07246
50. Yi, L., Kim, V.G., Ceylan, D., Shen, I., Yan, M., Su, H., Lu, C., Huang, Q., Sheffer, A., Guibas, L., et al.: A scalable active framework for region annotation in 3d shape collections. ACM Transactions on Graphics (TOG) **35**(6), 210 (2016)
51. You, Y., Lou, Y., Li, C., Cheng, Z., Li, L., Ma, L., Lu, C., Wang, W.: Keypointnet: A large-scale 3d keypoint dataset aggregated from numerous human annotations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13647–13656 (2020)
52. Zhou, B., Zhao, H., Puig, X., Xiao, T., Fidler, S., Barriuso, A., Torralba, A.: Semantic understanding of scenes through the ade20k dataset. International Journal of Computer Vision **127**(3), 302–321 (2019)

53. Zhou, T., Krahenbuhl, P., Aubry, M., Huang, Q., Efros, A.A.: Learning dense correspondence via 3d-guided cycle consistency. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 117–126 (2016)