ShapeMatcher: Self-Supervised Joint Shape Canonicalization, Segmentation, Retrieval and Deformation

Yan Di[1*], Chenyangguang Zhang[2*], Chaowei Wang[3*], Ruida Zhang[2], Guangyao Zhai[1], Yanyan Li[1], Bowen Fu[2], Xiangyang Ji[2] and Shan Gao†[3]
[1]Technical University of Munich, [2]Tsinghua University, [3]Northwestern Polytechnical University
shangbuhuan13@gmail.com, gaoshan@nwpu.edu.cn

Abstract

In this paper, we present ShapeMatcher, a unified self-supervised learning framework for joint shape canonicalization, segmentation, retrieval and deformation. Given a partially-observed object in an arbitrary pose, we first canonicalize the object by extracting point-wise affine-invariant features, disentangling inherent structure of the object with its pose and size. These learned features are then leveraged to predict semantically consistent part segmentation and corresponding part centers. Next, our lightweight retrieval module aggregates the features within each part as its retrieval token and compare all the tokens with source shapes from a pre-established database to identify the most geometrically similar shape. Finally, we deform the retrieved shape in the deformation module to tightly fit the input object by harnessing part center guided neural cage deformation. The key insight of ShapeMaker is the simultaneous training of the four highly-associated processes: canonicalization, segmentation, retrieval, and deformation, leveraging cross-task consistency losses for mutual supervision. Extensive experiments on synthetic datasets PartNet, ComplementMe, and real-world dataset Scan2CAD demonstrate that ShapeMatcher surpasses competitors by a large margin. Code is released at https://github.com/Det1999/ShapeMaker.

1. Introduction

In recent years, there has been a notable surge in research interest focused on generating high-quality 3D models from scans of complex scenes [6, 15, 16, 21, 35, 49, 73]. This technology encourages extensive applications in both artistic creation [56, 57], robotics [66, 67] and 3D scene perception [41, 69]. Existing methods [41, 54] typically directly utilize deep neural networks to reconstruct 3D models from imperfect scans. However, the presence of noise and occlusions poses a significant challenge in accurately capturing fine-grained geometric structures. To overcome this, Retrieval and Deformation (**R**&**D**) techniques [10,28,40,46,56–58,64] have been developed. These methods generally involve two key steps: first, identifying the most geometrically similar source shape from a pre-curated 3D database; and second, deforming the retrieved shape to achieve precise alignment with the target input. The **R**&**D** approach is particularly effective in producing 3D models enriched with fine details from source shapes.

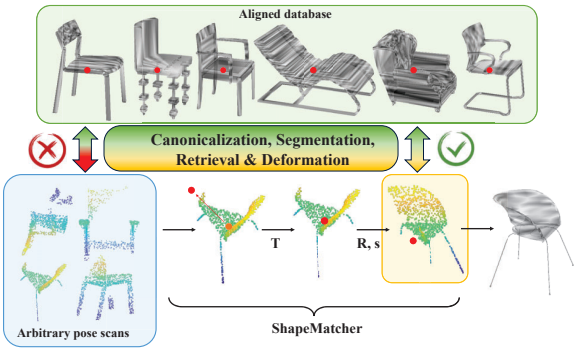

Figure 1. **Illustration of ShapeMatcher.** Objects obtained from real-world scans are typically noisy, partial and exhibit various poses, making it challenging to conduct an effective *R*&*D* process (Red 'X' on the left). To address this issue, we propose ShapeMatcher that first canonicalizes the objects and then segments them into semantic parts, facilitating **R**&**D** processes (Green '✓' on the right).

However, existing **R**&**D** methods usually encounter two primary challenges that make them susceptible to noise, occlusion and pose variations, and difficult to be practically utilized. **1)** Most **R**&**D** techniques [10, 17, 28, 40, 46, 56–58, 64] operate under the assumption that target shapes are aligned in a pre-processed canonical space. Typ-

*Authors with equal contributions. † Corresponding author.

ically, these methods are trained and tested on datasets where shapes have been manually adjusted to this canonical state. However, when these methods are deployed in real-world settings, they necessitate either manual alignment of scanned objects or the use of additional pose estimation networks [14, 18, 48, 71, 72]. Such procedures are not only time-intensive and laborious but also prone to yielding inconsistent results. This limitation significantly impedes the direct application of these methods in real-world scenarios. **2)** Previous methods [56, 57] do not design specially for dealing with partially-observed shapes, making it difficult to handle occluded objects. Although U-RED [17] considers the partial target shapes as input in the $\mathbf{R}\&\mathbf{D}$ process, it directly encodes the shape as a global embedding, which is not robust when dealing with significant occlusion.

To address the aforementioned two challenges, in this paper, we present ShapeMatcher, a novel framework that extends traditional $\mathbf{R}\&\mathbf{D}$ pipeline to joint self-supervised learning of object canonicalization, segmentation, retrieval and deformation. Our core contribution lies in that the four highly-associated processes can be trained simultaneously and supervise each other via constructing several cross-task consistency losses (Fig. 1). Specifically, given a partially-observed object scan in an arbitrary pose, ShapeMatcher processes the objects in four steps. First, we follow [30], which is based on Vector Neurons [13], to extract SE(3)-invariant point-wise features by progressively separating translation and rotation. We further follow [8] to normalize the features to disassociate the object scale. Until here, we successfully obtain affine-invariant point-wise features by disentangling object's inherent structure with its pose and size. This facilitates the *Canonicalization* of the observed object based on these intrinsic characteristics (Fig. 2 (B)). Then we predict semantically consistent part segmentation and corresponding part centers by feeding the learned features into our *Segmentation* module (Fig. 2 (C)). Based on the part segmentation, in the *Retrieval* module (Fig. 2 (D)), we aggregate features within each part and collect them together as a comprehensive retrieval token of the object. For partial objects, we introduce a region-weighted strategy, which assigns a weight to each part according to the point inside it. Parts with more points are assigned higher weights during retrieval, which is proved to be robust to occlusions. We compare the tokens of the target object with each shape in the pre-constructed database to identify the most geometrically similar (most similar tokens) source shape. In the final *Deformation* module (Fig. 2 (E)), the retrieved source shape is deformed to tightly match the target object via part center guided neural cage deformation [64].

To summarize, our main contributions are:

- We introduce ShapeMatcher, a novel self-supervised framework for joint shape canonicalization, segmentation, retrieval and deformation, handling partial target inputs under arbitrary poses. Extensive experiments on the synthetic and real-world datasets demonstrate that ShapeMatcher surpasses existing state-of-the-art approaches by a large margin.
- We demonstrate that the four highly-associated tasks: canonicalization, segmentation, retrieval and deformation, can be effectively trained simultaneously and supervise each other via constructing consistencies.
- We develop the region-weighted retrieval method to mitigate the impact of occlusions in the $\mathbf{R}\&\mathbf{D}$ process.

2. Related Works

Neural Shape Representation. The compact representation of 3D shapes in latent space, based on deep learning, has been a focal point for many researchers. Some attempts, such as [9, 27, 36, 37, 42, 45, 61, 70], employ neural networks to construct an implicit function, while others [1, 38, 52, 62, 68] directly model the shape of objects explicitly using generative models. Another common architecture in 3D shape representation learning, as seen in [12, 29, 44, 60, 63], is to use an encoder-decoder approach to generate latent representation vectors for various shapes. Although these methods have demonstrated impressive representation performance, they often struggle to generate fine-grained shapes when dealing with occlusion and noise.

CAD Model Retrieval and Deformation. Retrieval and Deformation ($\mathbf{R}\&\mathbf{D}$) methods lead another way to recover fine-grained geometric structures. Previous works directly retrieve the most similar CAD models by comparing the similarity of expression vectors in either descriptor space [5, 46] or the latent space of neural networks [4, 10, 22, 34]. Considering the subsequent deformation error, recent efforts introduce deformation-aware embeddings [56] or proposed new optimization objectives [24] to better capture the fine structure of deformed target objects. Nevertheless, these methods yield deteriorated performance when facing partial and pose-agnostic target shapes in real world. [17] achieves an one-to-many retrieval module for addressing the issue caused by partial observations, however, it receives canonicalized target shapes as input, which limit its applicability facing pose-agnostic target shapes in real world. As the retrieved models often exhibit some deviation from the target shape, the deformation module is used to minimize this discrepancy. Traditional approaches [20, 23, 47] aim to fit the target shape by directly optimizing the deformed shape. Neural network based techniques attempt to learn a set of deformation priors from a database of models. They represent deformations as volume warping [25,32], cage deformations [64], vertex offsets [58], or flows [28,65]. These methods typically constrain two shapes are aligned in the same coordinate system, making them challenging to apply in real-world scenarios.

SO(3)-Equivariant Methods. An increasing body of work [2,19,31,55,59] has initiated research on SO(3) equivariance. These efforts are mostly based on steerable convolutional kernels [33]. On the other hand, another set of works achieves equivariance through pose estimation. [43] estimates the object's pose to factor out SO(3) transformations, achieving approximate equivariance. While [51] learns pose estimation in a fully unsupervised manner, the equivariant backbone they employ [50] achieves equivariance primarily through data augmentation, leading to limited generalization. In this paper, we employ Vector Neural Multi-Layer Perceptron [13] as the backbone to get neural invariant features for object canonicalization. It achieves SO(3) equivariance by lifting traditional scalar neurons to vector neurons.

3. Method

Overview. ShapeMatcher consists of 4 modules, corresponding to the 4 highly-associated tasks. Each of the first 3 modules: *Canonicalization, Segmentation* and *Retrieval* modules have two parallel branches, one for complete point cloud (in orange background) and the other for partial object input (in blue background). As shown in Fig. 2, given a partial target shape $S_{tgt} \in \mathbb{R}^{N\times 3}$ in an arbitrary pose, in the *Canonicalization* module, we progressively decouples its inherent shape with rotation $R_{tgt} \in SO(3)$, translation $T_{tgt} \in \mathbb{R}^3$ and the 3D metric size $s_{tgt} \in \mathbb{R}^3$ via VN-MLP [8, 13, 30], yielding the affine-invariant point-wise features F_{tgt}. The object can then be canonicalized via inverse transformation based on intrinsics $\{R_{tgt}, T_{tgt}, s_{tgt}\}$. In the *Segmentation* module, F_{tgt} is fed into a 4-layer MLP network to predict M parts and corresponding part centers $\{K^1_{tgt}, K^2_{tgt}, ..., K^M_{tgt}\}$. The segmentation is semantically consistent across each category and thus can be matched and compared for $\mathbf{R}\&\mathbf{D}$. In the *Retrieval* module, inside each region M^i, we aggregate the features of all points inside it as its retrieval token Q^i. The retrieval token for the object is then represented as $\mathbf{Q_{tgt}} = \{Q^1_{tgt}, Q^2_{tgt}, ..., Q^M_{tgt}\}$. Similarly, during training, we obtain the intrinsics $\{R_{src}, T_{src}, s_{src}\}$, part centers $\{K^1_{src}, K^2_{src}, ..., K^M_{src}\}$ and retrieval tokens $\mathbf{Q_{src}} = \{Q^1_{src}, Q^2_{src}, ..., Q^M_{src}\}$ via the branch for complete point cloud. By comparing $\mathbf{Q_{tgt}}$ and $\mathbf{Q_{src}}$ of each source shape inside the database, we identify the most geometrically similar source shape S_r. In the final *Deformation* module, K_{tgt} and K_{src} are leveraged to guide the neural cage deformation [64] to deform the retrieved S_r towards S_{tgt}, yileding S^{dfm}_{src}.

3.1. Canonicalization

As shown in Fig. 2 (B), the *Canonicalization* module takes the target point cloud S_{tgt} as input and disentangle the inherent structure of S_{tgt} with the intrinsics $\{R_{tgt}, T_{tgt}, s_{tgt}\}$, yielding a point-wise affine-invariant feature F_{tgt}. Specifically, we follow VN-MLP [8, 13, 30] to first decouple translation via VNT [30] and then extract rotation via VNN [13,30]. We further follow [8] to normalize the SE(3)-invariant features obtained above the remove the influence of scaling, yielding F_{tgt} as follows,

$$R_{tgt}, T_{tgt}, F^*_{tgt} = \mathbf{VN\text{-}MLP}(S_{tgt}) \tag{1}$$

$$s_{tgt}, F_{tgt} = normalize(F^*_{tgt}) \tag{2}$$

where F^*_{tgt} denotes the SE(3)-invariant features and F_{tgt} denotes the affine-invariant features. Thereby, the object can be canonicalized with intrinsics as,

$$S^c_{tgt} = s_{tgt} R_{tgt} S_{tgt} + T_{tgt} \tag{3}$$

where S^c_{tgt} denotes the normalized and canonicalized shape of S^{tgt}. During training, in order to ensure that F_{tgt} fully encapsulate the geometric information of S_{tgt}, we integrate a supplementary reconstruction branch which takes F^*_{tgt} as input and reconstruct S^c_{tgt} in the affine-invariant space [30]. Please refer to the Supplementary Material for details. For source shape S_{src} from the database, we follow the same procedures to extract F_{src}.

3.2. Segmentation

Given the affine-invariant features F_{tgt}, we segment the input point cloud S_{tgt} into M semantically consistent parts. We use a 4-layer MLP Θ_l (Fig. 2 (C)) to predict a one-hot segmentation label for each point and use another 4-layer MLP Θ_c to predict M part centers $\{K^1_{src}, K^2_{src}, ..., K^M_{src}\}$. Noteworthy, we don't need any ground truth annotations in this segmentation process. Our experiments show that the network can automatically learn semantically consistent segmentation solely through consistency supervision from the other three tasks. For source shape S_{src}, we follow a similar process to obtain $\{K^1_{src}, K^2_{src}, ..., K^M_{src}\}$.

3.3. Retrieval

The retrieval network aims to identify the model S_{src} from an existing database that bears the closest resemblance to the target object S_{tgt} after deformation. Traditional methods [17, 57] directly extract the global features of objects for retrieval, which typically struggles with heavy occlusion since the global features are susceptible to noise and occlusion, and prone to producing erroneous retrieval results. In contrast, we employ a novel region-weighted retrieval method to explicitly encode independent and semantically consistent regions of the shape. This allows us to accurately handle partial shapes by identifying the visible regions to retrieve models most similar to the target.

Specifically, the part segmentation network Θ_l takes F_{tgt} as input to predict M regions $F_{seg} = [C_1, C_2, C_3, ..., C_N]^\top$

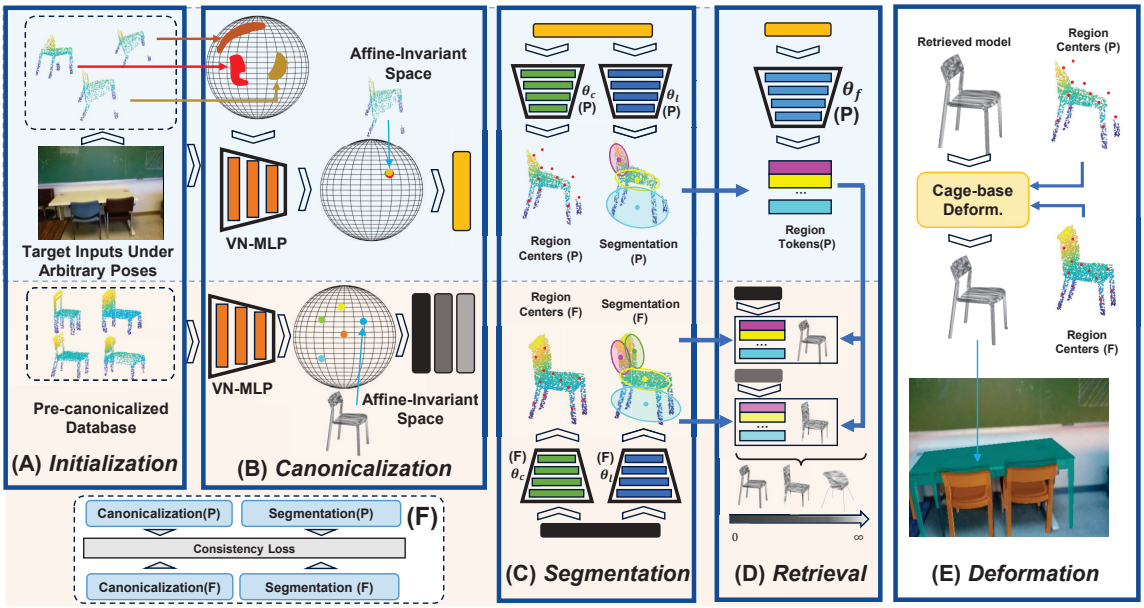

Figure 2. **The pipeline of ShapeMatcher.** Given a target point cloud obtained from a single-view scan and a pre-established database (A), ShapeMatcher generates the fine-grained reconstruction result using the joint 4 modules including *Canonicalization* (B), *Segmentation* (C), *Retrieval* (D) and *Deformation* (E), where the first three contains the partial branch for target processing and the full branch for source processing. Specifically, the target and source inputs are first canonicalized into the same affine-invariant space (B). Then, the semantic-consistent region segmentation is yielded from the affine-invariant features (C). The segmented regions are fed to the region-weight retrieval module (C) and the part center guided neural cage deformation module (E) for occlusion-robust $\mathbf{R}\&\mathbf{D}$ process. During training, the partial-full consistency losses (F) are enforced for the two branches.

of S^c_{tgt}, where $F_{seg} \in \mathbb{R}^{N\times M}$, $C_i \in \mathbb{R}^M$ represents the probability of point i belonging to each part center. Then we use another 4-layer feature aggregator Θ_f (Fig. 2 (D)) to extract the retrieval tokens Q of all parts as follows,

$$F_{cls} = F^{\top}_{seg} * \Theta_f(F_{tgt}), \tag{4}$$

$$Q_{tgt} = F_{cls}/(\sum_{n=1}^{N} F^{(n)}_{seg}). \tag{5}$$

where $Q_{tgt} \in \mathbb{R}^{M\times C}$ contains the C-dimensional retrieval tokens for all the M regions. Here we employ a soft assignment strategy where each point inside S_{tgt} is estimated M values describing the probabilities belonging to each of the M parts. Therefore, we first aggregate features F_{cls} on all points belonging to each part and then normalize F_{cls} using the sum of probabilities of points in each part, as in Eq. 4 and Eq. 5. Following a similar strategy, we can obtain Q_{src} for each source model in the pre-curated database. We just need to compare Q_{tgt} with the retrieval tokens Q_{src} of all source shapes using the weighted $\mathcal{L}_1$ distance,

$$Dis = \omega \sum \mathcal{L}_1(Q_{tgt} - Q_{src}) \tag{6}$$

where vector $\omega \in \mathbb{R}^{1\times M}$ stores the ratio of point number of each part with respect to the total point number N. Intuitively, parts with smaller point numbers contribute less in calculating the distance score, which reduces the influence of noise and occlusion. The source shape S_r with the smallest distance score is identified as the best retrieval.

3.4. Deformation

The *Deformation* module aims to deform the retrieved shape S_r to tightly match the target shape S_{tgt}. We utilize the neural cage scaffolding strategy as in [26, 64]. First, the neural cages C_{src} for S_r is pre-calculated. We utilize the part centers (K_{tgt} and K_{src}) to control the vertice offsets $C_{src2tgt}$ of the neural cage C_{src} to match S_{tgt}. In particular, we employ a neural network Θ_I to predict an influence vector $I \in \mathbb{R}^{N_c\times M}$ for each point concerning all cage vertices by $I = \Theta_I(concat(F_{tgt}, F_{src}))$, where N_c denotes the number of vertices used in C_{src}. $C_{src2tgt}$ is computed through the influence vectors I and the differences between region centers (Fig. 2 (E)):

$$C_{src2tgt} = C_{src} + \sum_{i=1}^{M} I_i(K^{(i)}_{src} - K^{(i)}_{tgt}), \tag{7}$$

Finally, we employ a sparse cage scaffolding strategy [26, 64] to achieve the deformation field of S_{src}. The deformed

shape $S_{src2tgt}$ of S_{src} can be expressed as follows:

$$S_{src2tgt} = S_{src} + \Psi(C_{src}, C_{src2tgt}), \tag{8}$$

where Ψ computes the displacement of each point in S_{src} by evaluating the differences between C_{src} and $C_{src2tgt}$, thereby achieving deformation.

3.5. ShapeMatcher: Joint Training

Our core insight in ShapeMatcher is that the four highly-associated tasks: *Canonicalization, Segmentation, Retrieval* and *Deformation* can be trained simultaneously and supervise each other via introducing cross-task consistency terms. We mainly introduce two types of losses here, *i.e.* partial-full consistency losses and task-oriented loss. For more details, please refer to the Supplementary Material.

Task-Oriented Loss. In the *Canonicalization*, we mainly use Chamfer Distance to constrain the canonicalized S^c_{tgt} and $\hat{S^c}_{tgt}$ predicted in the affine-invariant space by the supplementary reconstruction branch, to enforce the affine-invarianty of F^*_{tgt} in Sec. 3.1, by

$$\mathcal{L}_{can} = dis_{cham}(S^c_{tgt}, \hat{S^c}_{tgt}) + orth(R_{tgt}), \tag{9}$$

with $orth(R_{tgt})$ serving the purpose of enforcing the orthogonality of matrices.

In the *Segmentation*, to keep consistency between the part segmentation and the predicted part center, we jointly train Θ_l and Θ_c with the following loss, which enforce that each predicted part center K^i_{tgt} approximately lies in the center of all points belonging to the part M_i,

$$\mathcal{L}_{seg} = \sum_{m=1}^{M} \|K^{(m)}_{tgt} - (F^{\top}_{seg} * S^c_{tgt})^{(m)}\|_2 \tag{10}$$

To train the *Retrieval* and *Deformation* simultaneously, for an input target S_{tgt}, we randomly select a source model S_{src} from the database for training. Specifically, to eliminate the influence of occlusion, we do not directly use the global Chamfer Distance of S_{tgt} and S_{src} as ground truth. Instead, we employ a regional supervision strategy, ensuring that occluded areas do not contribute to the training of retrieval network. Taking the i-th region as an example, S^i_{tgt} represents all points in S_{tgt} that belong to the i-th region. We calculate the average of the nearest distances D_i from each point in S^i_{tgt} to the deformed shape S^{dfm}_{src} to enforce the learning of the regional retrieval tokens by

$$\mathcal{L}_{retrieval} = \frac{1}{M}\sum_{i=1}^{M} MSE(Q^{(i)}_{tgt} - Q^{(i)}_{src}, D_i). \tag{11}$$

The deformation loss is achieved by directly constraining the Chamfer Distance between S_{tgt} and S^{dfm}_{src}, expressed as:

$$\mathcal{L}_{deform} = dis_{cham}(S_{tgt}, S^{dfm}_{src}) + \|I\|_2 \tag{12}$$

where we regularize I using the L2 norm.

Partial-Full Consistency Losses. In the first two modules: *Canonicalization, Segmentation*, the full branch serves as a guidance to enhance the learning of the partial branch. Therefore, in each module, we can enforce corresponding consistency terms between the outputs of the two parallel branches (Fig. 2 (F)).

During the consistency training process, for randomly selected full input S_{full}, we generate a mask $U_{f2p} \in \mathbb{R}^N$ to crop it to simulate the situation of a partial input $S_{partial} = S_{full}U_{f2p}$.

In the *Canonicalization* module, to enforce the consistency in the affine-invariant space between the two branches, we apply the same transformation U_{f2p} to S^c_{full} in the affine-invariant space as before and then use the chamfer distance to constrain its distance to $S^c_{partial}$:

$$\mathcal{L}_{ccan} = dis_{cham}(S^c_{partial}, S^c_{full}U_{f2p}) \tag{13}$$

Similarly, in the *Segmentation* module, for the consistency constraint of the part center prediction network Θ_c, we directly use the Chamfer Distance to constrain the region centers detected by the two branches:

$$\mathcal{L}_{ccen} = dis_{cham}(K_{partial}, K_{full}). \tag{14}$$

In the segmentation network Θ_l, we mask the segmentation results of the full branch $F^{(full)}_{seg}$ and compare them with the results of the partial branch $F^{(partial)}_{seg}$:

$$\mathcal{L}_{cseg} = dis_{cham}(F^{(full)}_{seg}U_{f2p}, F^{(partial)}_{seg}). \tag{15}$$

Joint Training. Generally, the joint training of ShapeMatcher is divided into three stages. First, we train the full branch by $\mathcal{L}_{can}$ and $\mathcal{L}_{seg}$ for construction of *Canonicalization* and *Segmentation* ability. Second, the partial branch is introduced and trained by both the task-oriented losses for *Canonicalization* and *Segmentation* $\mathcal{L}_{can}$ and $\mathcal{L}_{seg}$ and the partial-full consistency loss terms $\mathcal{L}_{ccan}$, $\mathcal{L}_{ccen}$ and $\mathcal{L}_{cseg}$. Finally, after training *Canonicalization* and *Segmentation* of the both branches, $\mathcal{L}_{retrieval}$ and $\mathcal{L}_{deform}$ are adopted for joint *Retrieval* and *Deformation* training simultaneously utilizing the both branches to handle partial target inputs and full source inputs respectively.

4. Experiments

4.1. Experimental Setup

In this section, we mainly focus on **R&D** experiments, which better reflects the overall performance of the system. The ablations and analysis also demonstrate the effectiveness of considering joint *Canonicalization* and *Part Segmentation*.

Datasets. We evaluate the effectiveness of our joint framework using three datasets: two synthetic datasets,

PartNet [39] and ComplementMe [53], and one real-world dataset, Scan2CAD [3]. For datasets PartNet and ComplementMe, we follow the same database splits as in [57], separating their target inputs into training and testing sets. In our training process, we exclusively employ mesh models and do not utilize part segmentations as in [57], since the process of ShapeMatcher is fully self-supervised and does not need any additional annotations. The shapes used in PartNet and ComplementMe datasets are sourced from ShapeNet [7]. PartNet comprises 1,419 source models in the database, with 11,433 target models in the training set and 2,861 in the testing set. In ComplementMe, the numbers are 400, 11,311 and 2,825 respectively. In the synthetic cases, three categories of tables, chairs, and cabinets are evaluated on both datasets. Scan2CAD [3] is a real-world dataset developed based on ScanNet [11] with capacity of 14,225 objects. The input point cloud data on Scan2CAD is generated by reverse-projecting the depth images. In the real-world cases, we conduct training on the categories of tables, chairs, and cabinets from PartNet and directly testing on Scan2CAD.

Baselines. Both baseline methods, Uy *et al*. [57] and U-RED [17], are trained using the same data partitioning strategy stated above. To ensure fairness in comparison with ShapeMatcher, we augment the training data with pose variations, keeping other hyperparameters consistent with the original paper. During testing, we evaluate scenarios where target observations with arbitrary poses are directly used as input. Additionally, we test scenarios where the inputs are transformed using an offline pose estimation method [18], simulating the two-stage route of traditional methods with pre-canonicalizing (Uy *et al*. [57] + PE and U-RED [17] + PE). For experiments on Scan2CAD, we directly use the baseline models trained on PartNet with the 25% occlusion setting to conduct zero-shot testing, since real-world ground-truth models are inaccessible for training.

Evaluation Metrics. We utilize Chamfer Distance (CD) on the magnitude of 10^{-2} to assess both full shape scenarios and partial shape scenarios. We calculate the metrics following [57] to use the best result among the top 10 candidate objects. The final average metrics are obtained by averaging the results across all instances.

Implementation Details. During training, we uniformly sample objects to obtain point clouds with $M = 2500$ points to represent shapes. We directly generate partial point clouds from the corresponding full point clouds by random cropping for the partial branch inputs. We apply random pose augmentation to the input shape, specifically with random translations $T_{rand} \in [-0.1, 0.1]$ and random rotations $R_{rand} \in [-1, 1]$ on three Eulerian angles respectively. We set the initial learning rate to $1e-3$ and train ShapeMatcher for 200 epochs in every training stage of Sec. 3.5. Regarding the weight of the loss, in the first stage considering only the full branch, $\mathcal{L}_{can}$ and $\mathcal{L}_{seg}$ are equally weighted. In the second stage introducing the partial branch, we primarily emphasize the partial-full consistency losses, assigning significant weights to $\mathcal{L}_{ccan}$, $\mathcal{L}_{ccen}$ and $\mathcal{L}_{cseg}$ with weights set as 5, 2, and 2 respectively, while keeping the remaining weights default at 1. In the final stage for joint $\mathbf{R}\&\mathbf{D}$, both $\mathcal{L}_{retrieval}$ and $\mathcal{L}_{deform}$ are equally weighted.

4.2. Synthetic Cases

To validate the ability of ShapeMatcher tackling the challenge of arbitrary poses and occlusions, we first use synthetic datasets to simulate this scenario. We evaluate all methods [17, 57] where object observations with arbitrary poses are directly used as input. Additionally, we also report results where inputs are transformed and canonicalized using an offline pose estimation method [18] for baseline methods (Uy *et al*. + PE and U-RED + PE). Moreover, we analyze inference time of ShapeMatcher against the $\mathbf{R}\&\mathbf{D}$ baselines in the Supplementary Material.

We conduct two types of inputs for evaluation: full inputs using the PartNet and ComplementMe datasets, and partial input tests using 10%, 25%, and 50% occlusion rates on the PartNet dataset. The results of the full input tests are detailed in Table 1. In PartNet, our ShapeMatcher significantly outperforms the current leading competitors. For the Chamfer Distance on three categories, ShapeMatcher measures at 0.197, 0.150, and 0.519, maintaining the leading position. Even when the processed PE results are used as input, the baselines' results still fall short of ShapeMatcher. This demonstrates the effectiveness of adopting the affine-invariant features in the joint *Canonicalization* step. Results from ComplementMe supports the same conclusion, where ShapeMatcher reports significantly better results compared to the baseline methods. ShapeMatcher surpasses the top-performing Uy *et al.* + PE by 85.2%. Such superior results yielded by ShapeMatcher demonstrates that the joint consideration of all four steps improves the matching accuracy a lot.

For evaluation on partial inputs, we control the occlusion rates of partial point clouds by controlling the position of the cropping planes onto the full point clouds. The evaluation on partial inputs are presented in Table 2. Concretely, ShapeMatcher outperforms the current top method handling partial inputs U-RED by 5.018, 5.666 and 7.241 at the occlusion rate of 10%, 25% and 50% respectively. As the occlusion rate increases, the superiority of the ShapeMatcher method grows. Considering PE adopting, the same trend is exhibited. ShapeMatcher surpasses the U-RED + PE by 0.525, 0.949, 2.434 under three occlusion rates. It demonstrates that the proposed region-weighted retrieval brings strong robustness of our method against occlusion.

As shown in Fig. 3 and 4, our results shows more geo-

PartNet [39]				
Method	Chair	Table	Cabinet	Average
Uy *et al.* [57]	4.269	6.302	4.118	5.271
Uy *et al.* [57] + PE	1.507	3.006	1.070	2.219
U-RED [17]	5.331	4.980	9.141	5.463
U-RED [17] + PE	1.025	0.359	1.423	0.725
Ours	**0.197**	**0.150**	**0.519**	**0.200**
ComplementMe [53]				
Method	Chair	Table	Cabinet	Average
Uy *et al.* [57]	4.018	5.480	–	4.825
Uy *et al.* [57] + PE	1.439	2.454	–	1.999
U-RED [17]	8.575	5.800	–	7.044
U-RED [17] + PE	4.954	0.847	–	2.688
Ours	**0.253**	**0.328**	–	**0.294**

Table 1. The Chamfer Distance metrics for joint **R**&**D** results on full shapes under arbitrary poses.

Occlusion	Method	Chair	Table	Cabinet	Average
10%	Uy *et al.* [57]	4.372	6.395	4.179	5.365
	Uy *et al.* + PE [57]	1.523	2.982	**1.133**	2.219
	U-RED [17]	6.025	5.375	5.269	5.640
	U-RED [17] + PE	1.207	1.012	1.669	1.147
	Ours	**0.676**	**0.481**	1.212	**0.622**
25%	Uy *et al.* [57]	4.654	6.927	4.750	5.795
	Uy *et al.* + PE [57]	1.803	3.195	1.607	2.481
	U-RED [17]	5.196	7.215	8.164	6.442
	U-RED [17] + PE	1.684	1.387	2.795	1.625
	Ours	**0.878**	**0.643**	**1.071**	**0.776**
50%	Uy *et al.* [57]	6.070	9.322	7.929	7.841
	Uy *et al.* + PE [57]	3.314	5.030	4.584	4.272
	U-RED [17]	8.696	8.387	7.613	8.455
	U-RED [17] + PE	4.722	2.015	7.903	3.628
	Ours	**1.197**	**1.079**	**1.872**	**1.194**

Table 2. The Chamfer Distance metrics for joint **R**&**D** results on partial shapes under arbitrary poses of PartNet dataset.

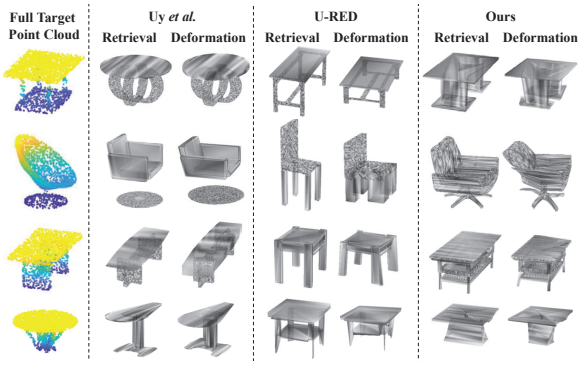

Figure 3. Qualitative **R&D** results with full target inputs on PartNet.

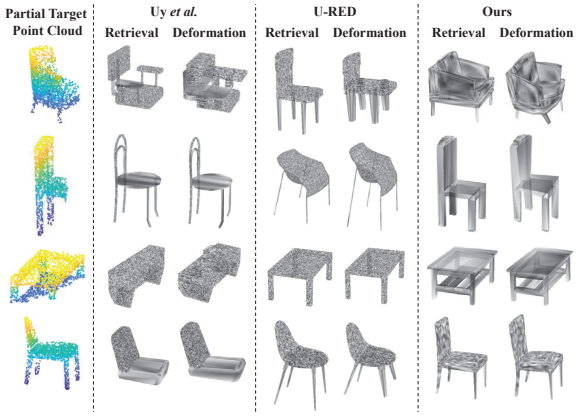

Figure 4. Qualitative **R&D** results with partial target inputs on the occlusion rate of 25% on PartNet.

metric resemblance to the targets compared to other methods. This is attributed to the suitable joint consideration of the four highly-associated processes, which accurately decouples the input poses, mapping them to a consistent space for accurate $R\&D$. The region-weighted retrieval we employ explicitly eliminates the influence of occluded areas, allowing for a more precise matching with the source model.

4.3. Real-world Cases

We test the effectiveness of ShapeMatcher on real-world datasets. In such case, ShapeMatcher is trained on the synthetic PartNet with 25% occlusion and directly tested on the partial scans of the real-world dataset Scan2CAD without manual pose adjustments. Table 3 displays our results, where our method significantly outperforms existing competitors. Particularly, compared to the U-RED, in three categories, the reported Chamfer Distance are reduced by 92%, 96%, and 94% respectively. In comparison to Uy *et al.*, the reported Chamfer Distance are reduced by 91%, 98%, and 92%. Even compared with the use of PE on the two baseline methods, there remains a significant leap forward of ShapeMatcher. This zero-shot experiments on real-world further prove that the procedure of joint *Canonicalization*, *Segmentation*, *Retrieval* and *Deformation* possesses a strong domain adaptation ability, showcasing the great potential of ShapeMatcher in real-world applications. The visualizations on Scan2CAD are provided in the supplementary material.

4.4. Ablation Studies

We conduct ablation experiments on PartNet, mainly on two aspects. First, in the *Canonicalization*, we investigate the importance of disentangling different pose intrinsics in Table 4, and demonstrate the effectiveness of joint considering *Canonicalization*. Second, in Table 5, we ablate the region-weighted *Retrieval* and the part center guided *Deformation*. Moreover, the robustness against occlusion is

Method	Chair	Table	Cabinet	Average
Uy *et al.* [57]	4.886	7.605	8.335	6.181
Uy *et al.* [57] + PE	3.362	6.657	7.261	4.905
U-RED [17]	5.490	5.131	10.091	5.945
U-RED [17] + PE	2.893	3.164	5.957	3.354
Ours	**0.423**	**0.186**	**0.654**	**0.375**

Table 3. The Chamfer Distance metrics for joint **R**&**D** results in real-world Scan2CAD [3].

analyzed in Table 2.

Canonicalization Capability. To study the impact of different intrinsics of object poses in the *Canonicalization* process, we conduct ablations on decoupling of translations, rotations and scales. The results are presented in Table 4. Specifically, in row (1), we make no adjustments to the input poses. Thanks to the regional-level $R\&D$ process, it shows decent performance. However, there is still a noticeable gap compared to row (4), indicating the significance of our proposed joint *Canonicalization*. In row (2), we solely decouple the input translations, resulting in a decrease of 14% in reported metrics. Moving to row (3), upon this foundation, we add decoupling for rotation, leading to a substantial decrease in reported Chamfer Distance, averaging at 0.244. In row (4), we introduce scale decoupling, resulting in another decrease in the reported metrics. It is evident that accurate decoupling of rotation is a crucial aspect of the success of the *Canonicalization* process. Moreover, it demonstrates that to integrate *Canonicalization* and $R\&D$ process is indispensable for the ShapeMatcher process.

Deformation and Retrieval Ability. To validate the effectiveness of our proposed region-weighted *Retrieval* and the part center guided neural cage *Deformation*, we conduct an ablation study on the PartNet dataset with the 25% occlusion rate. The results are presented in Table 5. In row (1), we conduct experiments using global retrieval and global deformation. This means we directly use an MLP network to extract overall point cloud features as the retrieval vector [57]. In the deformation network, similarly, we directly use an MLP network to generate neural cage offsets for deformation [26, 64]. Due to the lack of extraction of local information, the reported Chamfer Distance is more than twice of row (4). In row (2), we employ the global retrieval and the part center guided neural cage deformation. This improvement allows much more tightly-matched deformation by the retrieved source model, resulting in a 14% decrease in reported metrics. In row (3), we conduct experiments using the regional retrieval and the global deformation. The proposed regional-weighted retrieval handles occluded objects, reducing the impact of occluded parts and resulting in a substantial decrease in Chamfer Distance, down to 0.973.

Occlusion Robustness. We test shapes at different occlusion levels by altering the occlusion ratio in the input.

	Trans.	Rot.	Scal.	Chair	Table	Cabinet	Average
(1)				0.571	0.502	1.233	0.590
(2)	✓			0.468	0.442	1.096	0.506
(3)	✓	✓		0.213	0.200	0.674	0.244
(4)	✓	✓	✓	**0.197**	**0.150**	**0.519**	**0.200**

Table 4. Ablations on the *Canonicalization* process, we demonstrate the effectiveness of the joint *Canonicalization* by ablating different pose intrinsics. Here, Trans. denotes the decoupling of translation, Rot. represents the rotation, and Scal. signifies the scale.

	Gl. R.	Re. R.	Gl. D.	Re. D.	Chair	Table	Cabinet	Average
(1)	✓		✓		1.672	1.446	1.800	1.570
(2)	✓			✓	1.539	1.305	1.641	1.431
(3)		✓	✓		1.042	0.874	1.223	0.973
(4)		✓		✓	**0.878**	**0.643**	**1.071**	**0.776**

Table 5. Ablations of the $R\&D$ process. GL. R. denotes the global feature based retrieval [57], Re. R. represents the proposed region-weighted retrieval, GL. D. signifies direct neural cage deformation using global features [26, 64], and Re. D. denotes the adopted regional part center guided neural cage deformation.

For each specific occlusion ratio, we deliberately crop a portion of the complete point cloud to simulate occlusion. We test scenarios with occlusion ratios of 10%, 25%, and 50%, and the results are presented in Table 2. Observably, as the occluded regions increased, the Chamfer Distance significantly rises for the baseline methods. Taking the U-RED + PE as an example, its reported metrics increase from 1.147 to 3.628, doubling in value, as the occluded area expands. In contrast, our method increases by less than 1-fold, which exhibits strong robustness against occlusion.

5. Conclusion

In this paper, we present ShapeMatcher, a unified self-supervised learning framework for joint shape canonicalization, segmentation, retrieval and deformation. Given a partially-observed object in an arbitrary pose, we first canonicalize the object by extracting point-wise affine-invariant features. Then, the affine-invariant features are leveraged to predict semantically consistent part segmentation and corresponding part centers. Afterwards, the lightweight region-weighted retrieval module aggregates the features within each part as its retrieval token and compare all the tokens with source shapes from a pre-established database to identify the most geometrically similar shape. Finally, we deform the retrieved shape in the deformation module to tightly fit the input object by harnessing part center guided neural cage deformation. Extensive experiments on synthetic datasets PartNet, ComplementMe, and real-world dataset Scan2CAD demonstrate that ShapeMatcher surpasses competitors by a large margin. In the future, we plan to further applicate I-RED to various downstream tasks like robotic grasping. **Limitations.** are discussed in the Supplementary Material.

References

[1] Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas Guibas. Learning representations and generative models for 3d point clouds. In *International conference on machine learning*, pages 40–49. PMLR, 2018. 2

[2] Brandon Anderson, Truong Son Hy, and Risi Kondor. Cormorant: Covariant molecular neural networks. *Advances in neural information processing systems*, 32, 2019. 3

[3] Armen Avetisyan, Manuel Dahnert, Angela Dai, Manolis Savva, Angel X Chang, and Matthias Nießner. Scan2cad: Learning cad model alignment in rgb-d scans. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 2614–2623, 2019. 6, 8

[4] Armen Avetisyan, Angela Dai, and Matthias Nießner. End-to-end cad model retrieval and 9dof alignment in 3d scans. In *Proceedings of the IEEE/CVF International Conference on computer vision*, pages 2551–2560, 2019. 2

[5] Frederic Bosche and Carl T Haas. Automated retrieval of 3d cad model objects in construction range images. *Automation in Construction*, 17(4):499–512, 2008. 2

[6] Anh-Quan Cao and Raoul de Charette. Monoscene: Monocular 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3991–4001, 2022. 1

[7] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 6

[8] Yunlu Chen, Basura Fernando, Hakan Bilen, Matthias Nießner, and Efstratios Gavves. 3d equivariant graph implicit functions. In *European Conference on Computer Vision*, pages 485–502. Springer, 2022. 2, 3

[9] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5939–5948, 2019. 2

[10] Manuel Dahnert, Angela Dai, Leonidas J Guibas, and Matthias Nießner. Joint embedding of 3d scan and cad objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8749–8758, 2019. 1, 2

[11] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 6

[12] Angela Dai, Charles Ruizhongtai Qi, and Matthias Nießner. Shape completion using 3d-encoder-predictor cnns and shape synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5868–5877, 2017. 2

[13] Congyue Deng, Or Litany, Yueqi Duan, Adrien Poulenard, Andrea Tagliasacchi, and Leonidas J Guibas. Vector neurons: A general framework for so (3)-equivariant networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12200–12209, 2021. 2, 3

[14] Yan Di, Fabian Manhardt, Gu Wang, Xiangyang Ji, Nassir Navab, and Federico Tombari. So-pose: Exploiting self-occlusion for direct 6d pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12396–12405, 2021. 2

[15] Yan Di, Henrique Morimitsu, Shan Gao, and Xiangyang Ji. Monocular piecewise depth estimation in dynamic scenes by exploiting superpixel relations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4363–4372, 2019. 1

[16] Yan Di, Henrique Morimitsu, Zhiqiang Lou, and Xiangyang Ji. A unified framework for piecewise semantic reconstruction in dynamic scenes via exploiting superpixel relations. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10737–10743. IEEE, 2020. 1

[17] Yan Di, Chenyangguang Zhang, Ruida Zhang, Fabian Manhardt, Yongzhi Su, Jason Rambach, Didier Stricker, Xiangyang Ji, and Federico Tombari. U-red: Unsupervised 3d shape retrieval and deformation for partial point clouds. *arXiv preprint arXiv:2308.06383*, 2023. 1, 2, 3, 6, 7, 8

[18] Yan Di, Ruida Zhang, Zhiqiang Lou, Fabian Manhardt, Xiangyang Ji, Nassir Navab, and Federico Tombari. Gpv-pose: Category-level object pose estimation via geometry-guided point-wise voting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6781–6791, 2022. 2, 6

[19] Carlos Esteves, Christine Allen-Blanchette, Ameesh Makadia, and Kostas Daniilidis. Learning so (3) equivariant representations with spherical cnns. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 52–68, 2018. 3

[20] Vignesh Ganapathi-Subramanian, Olga Diamanti, Soeren Pirk, Chengcheng Tang, Matthias Niessner, and Leonidas Guibas. Parsing geometry using structure-aware shape templates. In *2018 International Conference on 3D Vision (3DV)*, pages 672–681. IEEE, 2018. 2

[21] Georgia Gkioxari, Jitendra Malik, and Justin Johnson. Mesh r-cnn. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9785–9795, 2019. 1

[22] Can Gümeli, Angela Dai, and Matthias Nießner. Roca: Robust cad model retrieval and alignment from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4022–4031, 2022. 2

[23] Qi-Xing Huang, Bart Adams, Martin Wicke, and Leonidas J Guibas. Non-rigid registration under isometric deformations. In *Computer Graphics Forum*, volume 27, pages 1449–1457. Wiley Online Library, 2008. 2

[24] Vladislav Ishimtsev, Alexey Bokhovkin, Alexey Artemov, Savva Ignatyev, Matthias Niessner, Denis Zorin, and Evgeny Burnaev. Cad-deform: Deformable fitting of cad models to 3d scans. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pages 599–628. Springer, 2020. 2

[25] Dominic Jack, Jhony K Pontes, Sridha Sridharan, Clinton Fookes, Sareh Shirazi, Frederic Maire, and Anders Eriksson. Learning free-form deformations for 3d object reconstruction. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6,*

2018, Revised Selected Papers, Part II 14, pages 317–333. Springer, 2019. 2

[26] Tomas Jakab, Richard Tucker, Ameesh Makadia, Jiajun Wu, Noah Snavely, and Angjoo Kanazawa. Keypointdeformer: Unsupervised 3d keypoint discovery for shape control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12783–12792, 2021. 4, 8

[27] Wonbong Jang and Lourdes Agapito. Codenerf: Disentangled neural radiance fields for object categories. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12949–12958, 2021. 2

[28] Chiyu Jiang, Jingwei Huang, Andrea Tagliasacchi, and Leonidas J Guibas. Shapeflow: Learnable deformation flows among 3d shapes. *Advances in Neural Information Processing Systems*, 33:9745–9757, 2020. 1, 2

[29] Jincen Jiang, Xuequan Lu, Lizhi Zhao, Richard Dazaley, and Meili Wang. Masked autoencoders in 3d point cloud representation learning. *IEEE Transactions on Multimedia*, 2023. 2

[30] Oren Katzir, Dani Lischinski, and Daniel Cohen-Or. Shape-pose disentanglement using se (3)-equivariant vector neurons. In *European Conference on Computer Vision*, pages 468–484. Springer, 2022. 2, 3

[31] Risi Kondor, Zhen Lin, and Shubhendu Trivedi. Clebsch–gordan nets: a fully fourier space spherical convolutional neural network. *Advances in Neural Information Processing Systems*, 31, 2018. 3

[32] Andrey Kurenkov, Jingwei Ji, Animesh Garg, Viraj Mehta, JunYoung Gwak, Christopher Choy, and Silvio Savarese. Deformnet: Free-form deformation network for 3d shape reconstruction from a single image. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 858–866. IEEE, 2018. 2

[33] Leon Lang and Maurice Weiler. A wigner-eckart theorem for group equivariant convolution kernels. *arXiv preprint arXiv:2010.10952*, 2020. 3

[34] Yangyan Li, Hao Su, Charles Ruizhongtai Qi, Noa Fish, Daniel Cohen-Or, and Leonidas J Guibas. Joint embeddings of shapes and images via cnn image purification. *ACM transactions on graphics (TOG)*, 34(6):1–12, 2015. 2

[35] Jiachen Liu, Pan Ji, Nitin Bansal, Changjiang Cai, Qingan Yan, Xiaolei Huang, and Yi Xu. Planemvs: 3d plane reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8665–8675, 2022. 1

[36] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019. 2

[37] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2

[38] Kaichun Mo, Paul Guerrero, Li Yi, Hao Su, Peter Wonka, Niloy Mitra, and Leonidas J Guibas. Structurenet: Hierarchical graph networks for 3d shape generation. *arXiv preprint arXiv:1908.00575*, 2019. 2

[39] Kaichun Mo, Shilin Zhu, Angel X Chang, Li Yi, Subarna Tripathi, Leonidas J Guibas, and Hao Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 909–918, 2019. 6, 7

[40] Liangliang Nan, Ke Xie, and Andrei Sharf. A search-classify approach for cluttered indoor scene understanding. *ACM Transactions on Graphics (TOG)*, 31(6):1–10, 2012. 1

[41] Yinyu Nie, Xiaoguang Han, Shihui Guo, Yujian Zheng, Jian Chang, and Jian Jun Zhang. Total3dunderstanding: Joint layout, object pose and mesh reconstruction for indoor scenes from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 55–64, 2020. 1

[42] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174, 2019. 2

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

[44] Yuchen Rao, Yinyu Nie, and Angela Dai. Patchcomplete: Learning multi-resolution patch priors for 3d shape completion on unseen categories. *Advances in Neural Information Processing Systems*, 35:34436–34450, 2022. 2

[45] Edoardo Remelli, Artem Lukoianov, Stephan Richter, Benoit Guillard, Timur Bagautdinov, Pierre Baque, and Pascal Fua. Meshsdf: Differentiable iso-surface extraction. *Advances in Neural Information Processing Systems*, 33:22468–22478, 2020. 2

[46] Adriana Schulz, Ariel Shamir, Ilya Baran, David IW Levin, Pitchaya Sitthi-Amorn, and Wojciech Matusik. Retrieval on parametric shape collections. *ACM Transactions on Graphics (TOG)*, 36(1):1–14, 2017. 1, 2

[47] Olga Sorkine and Marc Alexa. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, volume 4, pages 109–116. Citeseer, 2007. 2

[48] Yongzhi Su, Yan Di, Guangyao Zhai, Fabian Manhardt, Jason Rambach, Benjamin Busam, Didier Stricker, and Federico Tombari. Opa-3d: Occlusion-aware pixel-wise aggregation for monocular 3d object detection. *IEEE Robotics and Automation Letters*, 8(3):1327–1334, 2023. 2

[49] Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15598–15607, 2021. 1

[50] Weiwei Sun, Wei Jiang, Eduard Trulls, Andrea Tagliasacchi, and Kwang Moo Yi. Acne: Attentive context normalization

for robust permutation-equivariant learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11286–11295, 2020. 3

[51] Weiwei Sun, Andrea Tagliasacchi, Boyang Deng, Sara Sabour, Soroosh Yazdani, Geoffrey Hinton, and Kwang Moo Yi. Canonical capsules: Unsupervised capsules in canonical pose. *arXiv preprint arXiv:2012.04718*, 1, 2021. 3

[52] Yongbin Sun, Yue Wang, Ziwei Liu, Joshua Siegel, and Sanjay Sarma. Pointgrow: Autoregressively learned point cloud generation with self-attention. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 61–70, 2020. 2

[53] Minhyuk Sung, Hao Su, Vladimir G Kim, Siddhartha Chaudhuri, and Leonidas Guibas. Complementme: Weakly-supervised component suggestions for 3d modeling. *ACM Transactions on Graphics (TOG)*, 36(6):1–12, 2017. 6, 7

[54] Maxim Tatarchenko, Stephan R Richter, René Ranftl, Zhuwen Li, Vladlen Koltun, and Thomas Brox. What do single-view 3d reconstruction networks learn? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3405–3414, 2019. 1

[55] Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018. 3

[56] Mikaela Angelina Uy, Jingwei Huang, Minhyuk Sung, Tolga Birdal, and Leonidas Guibas. Deformation-aware 3d model embedding and retrieval. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 397–413. Springer, 2020. 1, 2

[57] Mikaela Angelina Uy, Vladimir G Kim, Minhyuk Sung, Noam Aigerman, Siddhartha Chaudhuri, and Leonidas J Guibas. Joint learning of 3d shape retrieval and deformation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11713–11722, 2021. 1, 2, 3, 6, 7, 8

[58] Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. 3dn: 3d deformation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1038–1046, 2019. 1, 2

[59] Maurice Weiler, Mario Geiger, Max Welling, Wouter Boomsma, and Taco S Cohen. 3d steerable cnns: Learning rotationally equivariant features in volumetric data. *Advances in Neural Information Processing Systems*, 31, 2018. 3

[60] Haozhe Xie, Hongxun Yao, Xiaoshuai Sun, Shangchen Zhou, and Shengping Zhang. Pix2vox: Context-aware 3d reconstruction from single and multi-view images. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2690–2698, 2019. 2

[61] Qiangeng Xu, Zexiang Xu, Julien Philip, Sai Bi, Zhixin Shu, Kalyan Sunkavalli, and Ulrich Neumann. Point-nerf: Point-based neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5438–5448, 2022. 2

[62] Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge Belongie, and Bharath Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4541–4550, 2019. 2

[63] Shuo Yang, Min Xu, Haozhe Xie, Stuart Perry, and Jiahao Xia. Single-view 3d object reconstruction from shape priors in memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3152–3161, 2021. 2

[64] Wang Yifan, Noam Aigerman, Vladimir G Kim, Siddhartha Chaudhuri, and Olga Sorkine-Hornung. Neural cages for detail-preserving 3d deformations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 75–83, 2020. 1, 2, 3, 4, 8

[65] Michela Zaccaria, Fabian Manhardt, Yan Di, Federico Tombari, Jacopo Aleotti, and Mikhail Giorgini. Self-supervised category-level 6d object pose estimation with optical flow consistency. *IEEE Robotics and Automation Letters*, 8(5):2510–2517, 2023. 2

[66] Guangyao Zhai, Xiaoni Cai, Dianye Huang, Yan Di, Fabian Manhardt, Federico Tombari, Nassir Navab, and Benjamin Busam. Sg-bot: Object rearrangement via coarse-to-fine robotic imagination on scene graphs. *arXiv preprint arXiv:2309.12188*, 2023. 1

[67] Guangyao Zhai, Dianye Huang, Shun-Cheng Wu, HyunJun Jung, Yan Di, Fabian Manhardt, Federico Tombari, Nassir Navab, and Benjamin Busam. Monograspnet: 6-dof grasping with a single rgb image. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1708–1714. IEEE, 2023. 1

[68] Guangyao Zhai, Evin Pinar Örnek, Shun-Cheng Wu, Yan Di, Federico Tombari, Nassir Navab, and Benjamin Busam. Commonscenes: Generating commonsense 3d indoor scenes with scene graphs. *arXiv preprint arXiv:2305.16283*, 2023. 2

[69] Cheng Zhang, Zhaopeng Cui, Yinda Zhang, Bing Zeng, Marc Pollefeys, and Shuaicheng Liu. Holistic 3d scene understanding from a single image with implicit representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8833–8842, 2021. 1

[70] Chenyangguang Zhang, Yan Di, Ruida Zhang, Guangyao Zhai, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. Ddf-ho: Hand-held object reconstruction via conditional directed distance field. *arXiv preprint arXiv:2308.08231*, 2023. 2

[71] Ruida Zhang, Yan Di, Zhiqiang Lou, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. Rbp-pose: Residual bounding box projection for category-level pose estimation. In *European Conference on Computer Vision*, pages 655–672. Springer, 2022. 2

[72] Ruida Zhang, Yan Di, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. Ssp-pose: Symmetry-aware shape prior deformation for direct category-level object pose estimation. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7452–7459. IEEE, 2022. 2

[73] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12786–12796, 2022. 1