Part-level Scene Reconstruction Affords Robot Interaction

Zeyu Zhang[1,2]*, Lexing Zhang[1,3]*, Zaijin Wang[1], Ziyuan Jiao[1,2], Muzhi Han[1,2], Yixin Zhu[4], Song-Chun Zhu[1,3,4], Hangxin Liu[1]†

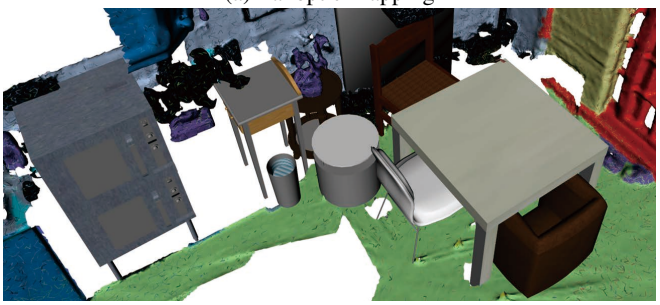
***Abstract*— Existing methods for reconstructing interactive scenes primarily focus on replacing reconstructed objects with CAD models retrieved from a limited database, resulting in significant discrepancies between the reconstructed and observed scenes. To address this issue, our work introduces a part-level reconstruction approach that reassembles objects using primitive shapes. This enables us to precisely replicate the observed physical scenes and simulate robot interactions with both rigid and articulated objects. By segmenting reconstructed objects into semantic parts and aligning primitive shapes to these parts, we assemble them as CAD models while estimating kinematic relations, including parent-child contact relations, joint types, and parameters. Specifically, we derive the optimal primitive alignment by solving a series of optimization problems, and estimate kinematic relations based on part semantics and geometry. Our experiments demonstrate that part-level scene reconstruction outperforms object-level reconstruction by accurately capturing finer details and improving precision. These reconstructed part-level interactive scenes provide valuable kinematic information for various robotic applications; we showcase the feasibility of certifying mobile manipulation planning in these interactive scenes before executing tasks in the physical world.**

I. Introduction

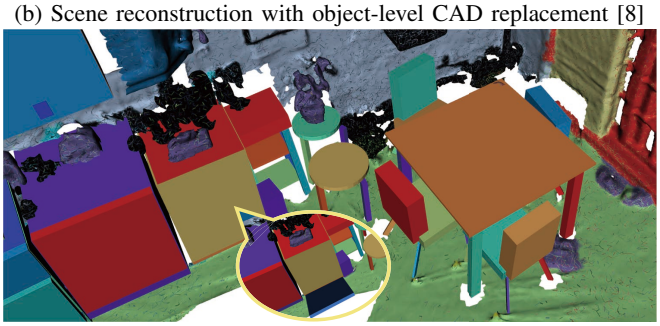
Reconstructing surroundings is critical for robots, enabling them to understand and interact with their environments. However, traditional scene reconstruction methods primarily focus on generating static scenes, represented by sparse landmarks [1, 2], occupancy grids [3], surfels [4, 5], volumetric voxels [6, 7], or semantic objects [2]. These representations lack the ability to capture the dynamic nature of robot operations and limit the complexity of tasks that can be performed, such as interactions with objects beyond pick-and-place. This limitation calls for a new approach that places interactions at the core of scene reconstruction.

By enriching the reconstructed scenes with interactions that allow robots to anticipate action effects and verify their plans without executing them in the physical world, Han *et al*. proposed a novel task of reconstructing interactive scenes that can be imported into ROS-based simulators [8, 9], which is crucial for long-horizon task and motion planning (TAMP) [10–12]. This approach involves using a 3D panoptic mapping method to reconstruct scenes from RGB-D data, segmenting objects, and representing them as 3D meshes (Fig. 1a). The segmented object meshes are then replaced with CAD models from a database, which provide actionable information about how robots can interact with them. This approach facilitates the emulation of complex interactions in simulated environments. Fig. 1b shows a reconstructed interactive scene with objects replaced by CAD models.

(a) Panoptic mapping



(b) Scene reconstruction with object-level CAD replacement [8]



(c) Scene reconstruction with part-level CAD replacement (ours)

Fig. 1: **Reconstructing interactive scenes.** (a) The initial step involves generating a panoptic mapping result with recognized and segmented object instances. (b) Existing methods [8, 9] replace the objects with pre-built CAD models, resulting in significant differences from the observed scenes. (c) In contrast, our proposed approach focuses on part-level reconstruction, replacing object parts with aligned primitive shapes. This yields interactive scenes with finer details and better alignment with the physical environment.

Despite the successful attempt to reconstruct interactive scenes, there are challenges in reproducing the observed scenes with adequate fidelity. As shown in Figs. 1a and 1b, noisy perception and severe occlusions in the scans often result in unsatisfactory CAD replacements. The database's

* Z. Zhang and L. Zhang contributed equally to this work. Emails: zeyuzhang@ucla.edu, zhanglexing@bigai.ai.

† Corresponding author. Email: liuhx@bigai.ai.

[1] National Key Laboratory of General Artificial Intelligence, Beijing Institute for General Artificial Intelligence (BIGAI). [2] Center for Vision, Cognition, Learning, and Autonomy (VCLA), Statistics Department, UCLA. [3] School of Intelligence Science and Technology, Peking University. [4] Institute for Artificial Intelligence, Peking University.

limited number of CAD models further compounds this issue, as they cannot account for the wide variety of objects robots may encounter. As a result, the reconstructed interactive scenes may lack realism and fail to represent the physical scenes accurately.

In this work, we aim to improve the fidelity of reconstructed interactive scenes by extending the approach of Han *et al*. [8, 9]. We propose a part-level reconstruction strategy that focuses on reconstructing scenes by replacing object CAD models at the part level instead of the object level. We employ a semantic point cloud completion network to decompose and complete each noisily segmented object into parts. Next, we perform part-level CAD replacement, including aligning primitive shapes to individual parts and estimating their kinematic relations. This pipeline (see Fig. 2) enables the creation of a kinematics-based scene graph that captures the geometry, semantics, and kinematic constraints of the environment, facilitating more realistic robot interactions. Our part-level reconstructed interactive scenes (see Fig. 1c) closely align with the physical scenes, providing the fidelity that can enable more accurate simulations of robot interactions.

A. Related work

Constructing an effective **scene representation** that facilitates robot mobile manipulation planning is an open problem. Traditional semantic mapping and simultaneous localization and mapping (SLAM) methods produce flat representations primarily suited for navigational tasks [13]. In contrast, graph-based representations such as scene grammar [14–16] and 3D scene graphs [8, 17–19] offer more structural and contextual information, enabling more versatile robot planning capabilities. In particular, Han *et al*. [8, 9] introduced a contact graph that can be automatically converted into a unified robot description format (URDF), providing robots with interpretable kinematic relations [11, 12, 20, 21]. Building upon this work, our approach extends the field by introducing a part-level CAD replacement algorithm for reconstructing interactive scenes.

In the domain of object modeling, **part-based** approaches leverage computer vision techniques to track movements among object parts [22, 23], exploit contextual relations from large datasets [24–26] or develop data-efficient learning methods [27, 28]. These approaches aim to recognize and segment object parts, enhancing the understanding of complex object structures, but they do not yield a holistic representation of a scene that encompasses multiple objects.

Part-level **primitive shapes**, such as spheres, cylinders, and cuboids, have been utilized to simplify the modeling of complex objects in images [29, 30] and scenes [31, 32]. However, these part-level representations typically focus on the static aspect of the perceived environment, lacking the essential kinematic information required for robots to actively interact with their surroundings. This absence of kinematic information is a common limitation in the robotics community. To bridge this crucial information gap, we propose a novel part-level framework that leverages primitive shapes and estimates their kinematic relations.

B. Overview

This paper is organized as follows: Sec. II presents our kinematics-based scene graph representation. Sec. III introduces the part-level CAD replacement algorithm. In Sec. IV, we demonstrate the efficacy of the proposed method in various settings. Finally, Sec. V concludes the paper.

II. Kinematics-based Scene Representation

We extend the contact graph (cg) introduced by Han *et al*. [8] to represent 3D indoor scenes by incorporating scene entity parts and their kinematic information. The $cg = (pt, E)$ consists of a parse tree (pt) and a set of proximal relations (E). The parse tree organizes scene entity nodes (V) hierarchically based on supporting relations (S), while the proximal relations capture the relationships between objects. Each object node in V includes attributes describing its semantics and geometry while supporting and proximal relations impose constraints to ensure physically plausible object placements.

To enhance the cg, we introduce an additional attribute, denoted as pt^p, to each object node $v \in V$. This attribute represents a per-object part-level parse tree (pt^p), which organizes part entities (V^p) along with their kinematic relations ($\mathcal{J}$). The part entities and kinematic relations are defined as follows.

The set of **part entity nodes**, denoted as $V^p = v^p$, represents all part entities within an object. Each part entity, $v^p = \langle l, c, M, \Pi \rangle$, encodes a unique part instance label (l), a part semantic label (c) such as "table leg," a geometry model (M) in the form of a triangular mesh or point cloud, and a set of surface planes (Π). The surface planes are represented as $\Pi = (\boldsymbol{\pi}^k, U^k)$, where U^k is a list of 3D vertices defining a polygon that outlines the plane $\boldsymbol{\pi}^k$. The plane $\boldsymbol{\pi}^k$ is represented by a homogeneous vector $[\boldsymbol{n}^{k^T}, d^k]^T \in \mathbb{R}^4$ in projective space. The unit plane normal vector is denoted as $\boldsymbol{n}_i^k$, and the equation $\boldsymbol{n}^{k^T} \cdot \boldsymbol{u} + d^k = 0$ describes the constraint satisfied by any point $\boldsymbol{u} \in \mathbb{R}^3$ on the plane.

The set of **kinematic relations**, $\mathcal{J} = J_{p,c}$, represents the parametric joints between part entities within an object. A joint, $J_{p,c} = \langle t_{p,c}, T_{p,c}, \mathcal{F}p, c \rangle$, exists between a parent part (v_p) and a child part (v_c). The joint encodes the joint type (tp, c), the parent-to-child transformation ($T_{p,c}$), and the joint axis ($\mathcal{F}_{p,c} \in \mathbb{R}^3$).

In this paper, we consider three types of joints:

- `fixed joint`: Represents a rigid connection between two parts, such as a table top and a table leg.
- `prismatic joint`: Indicates that one part can slide along a single axis with respect to the other, as seen in an openable drawer and a cabinet base.
- `revolute joint`: Represents a joint where one part can rotate around a single axis in relation to another part, like the door and base of a microwave.

Establishing a kinematic relation between two parts v_p and v_c requires them to be in contact with each other by

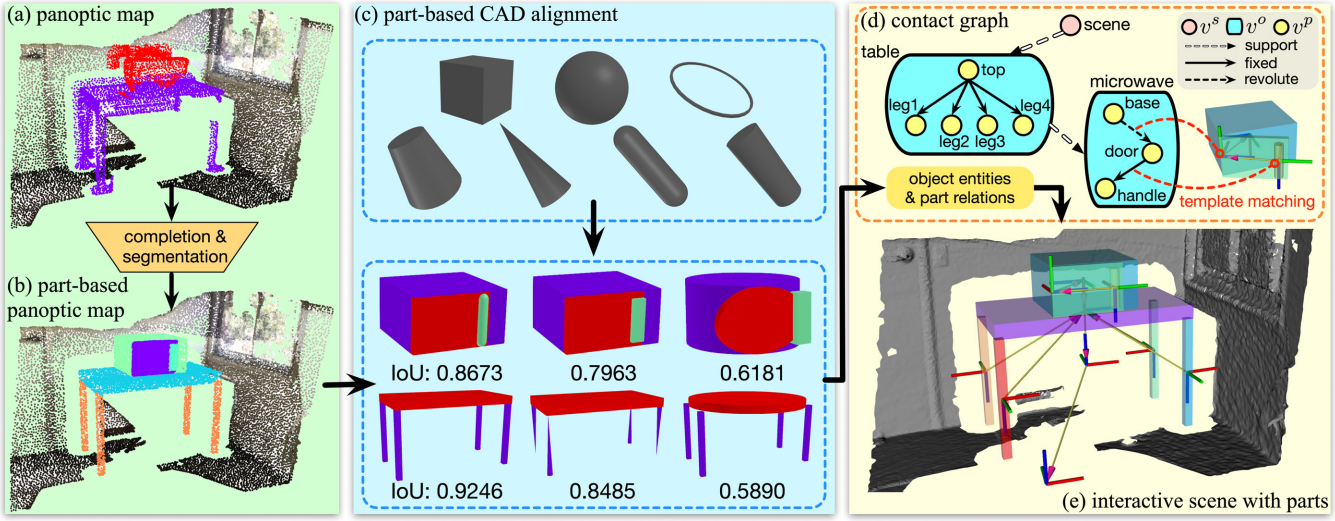


Fig. 2: **System architecture for part-level interactive scene reconstruction.** (a) The initial step involves completing and segmenting the point clouds of the noisily segmented 3D objects, resulting in (b) a part-based panoptic map. (c) Each completed object part is replaced with the most aligned primitive shape. The optimal combination of part alignments, determined by the highest IoU, is selected to (d) estimate the kinematic relations among the parts. (e) The replaced object parts and their relations are compiled into a URDF representation, capturing the kinematics of objects and the scene. This URDF can be imported into various simulators for TAMP tasks.

satisfying the following constraints:

$$\begin{aligned} &\exists\ (\boldsymbol{\pi}_p^i, U_p^i) \in \Pi_p,\ (\boldsymbol{\pi}_c^j, U_c^j) \in \Pi_c, \\ \text{s.t. Align}&\left(\boldsymbol{\pi}_p^i, \boldsymbol{\pi}_c^j\right) \stackrel{\text{def}}{=} \text{abs}\left({\boldsymbol{n}_p^i}^T \boldsymbol{n}_c^j\right) \geqslant \theta_a, \\ \text{Dist}&\left(\boldsymbol{\pi}_p^i, \boldsymbol{\pi}_c^j\right) \stackrel{\text{def}}{=} \frac{1}{|U_c^j|} \sum_{\boldsymbol{u} \in U_c^j} {\boldsymbol{n}_p^i}^T \boldsymbol{u} + d_p^i \leqslant \theta_d, \\ \text{Cont}&\left(U_p^i, U_c^j\right) \stackrel{\text{def}}{=} \text{A}\left(U_p^i \cap \text{proj}_{p,i}(U_c^j)\right) / \text{A}\left(U_c^j\right) \geqslant \theta_c, \end{aligned} \tag{1}$$

where:

- Align $\left(\boldsymbol{\pi}_p^i, \boldsymbol{\pi}_c^j\right)$ defines the alignment between two surface planes, abs$(\cdot)$ computes the absolute value, and θ_a is the threshold to determine a good alignment ($\theta_a = 1$ for a perfect alignment where two planes are parallel);
- Dist $\left(\boldsymbol{\pi}_p^i, \boldsymbol{\pi}_c^j\right)$ defines the distance between the surface planes by averaging the distances from vertices of polygon U_c^j (that outlines the surface plane $\boldsymbol{\pi}_c^j$) to plane $\boldsymbol{\pi}_p^i$, $|U|$ is the number of vertices, and θ_d is the maximum distance allowed;
- Cont $\left(U_p^i, U_c^j\right)$ defines the contact ratio, A$(\cdot)$ computes the area of a polygon, θ_c is the minimum sufficient contact ratio, $\cap$ computes the intersection between two polygons, and $\text{proj}_{p,i}(U_c^j)$ projects polygon U_c^j onto the plane $\boldsymbol{\pi}_p^i$ by projecting each vertex in U_c^j onto $\boldsymbol{\pi}_p^i$:

$$\hat{\boldsymbol{u}}_c^j = \boldsymbol{u}_c^j - {\boldsymbol{n}_p^i}^T (\boldsymbol{u}_c^j - \boldsymbol{u}_p^i)\ \boldsymbol{n}_p^i,\ \forall \boldsymbol{u}_c^j \in U_c^j, \tag{2}$$

where $\hat{\boldsymbol{u}}_c^j$ is the projected point of $\boldsymbol{u}_c^j$ on the plane $\boldsymbol{\pi}_p^i$, and $\boldsymbol{u}_p^i$ is an arbitrary point on $\boldsymbol{\pi}_p^i$.

By definition, a cg augmented with object parts and kinematics sufficiently defines objects' semantics, geometry, and articulations in a scene. Crucially, such a representation is also naturally compatible with the kinematic tree and could be seamlessly converted to a URDF for various downstream applications. Leveraging cg, a robot can reason about the action outcomes when it interacts with (articulated) objects.

III. Part-level CAD Replacement

We aim to replace the segmented and completed part entities (as shown in Fig. 2b) with best-aligned primitive shapes while estimating their kinematic relations, and construct a part-level contact graph cg as defined in Sec. II.

A. Individual part alignment

For each individual part, we select a primitive shape with a sufficient level of similarity and calculate a 6D transformation to align the shape to the part. Given a part entity with a point cloud segment P, we find an optimal primitive shape M^* from a finite set of primitive candidates $\mathcal{M}^c$ and an optimal 6D transformation $T_{ind}^* \in SE(3)$ that aligns M^* with P. We obtain $\mathcal{M}^c$ based on pre-defined primitive shape templates and a 3D scaling vector estimated from the minimum-volume oriented 3D bounding box of P [33]. The optimization problem can be formulated as follows:

$$M^*, T_{ind}^* = \min_{M_i \in \mathcal{M}^c, T \in SE(3)} \frac{1}{|h(M_i)|} \sum_{\boldsymbol{u} \in h(M_i)} d_P(T_i \circ \boldsymbol{u}), \tag{3}$$

where $h(M_i)$ is a set of evenly sampled points on the surface of the CAD model M_i, $d_P(\boldsymbol{u})$ is the distance from a sampled point $\boldsymbol{u}$ to the closest point in P, and $T_i \circ \boldsymbol{u}$ is the position of point $\boldsymbol{u}$ after applying transformation T_i.

To solve this optimization problem, we compute the optimal transformation T_i^* for each primitive candidate M_i using the iterative closest point method [34]. Then M^* is the primitive candidate with the smallest minimum total distance among all candidates $\{M_i\}$, and T_{ind}^* is the corresponding optimal transformation in $\{T_i^*\}$.

B. Kinematic relation estimation

After replacing part entities with primitive shapes based on individual shape alignment results, we estimate the parent-child contact relations and kinematics (*i.e.*, parametric joints)

between parts to obtain a per-object part-level parse tree pt^p. To initialize a part node v^p:

1) We acquire its part-level semantic label c, instance label l, and point cloud P.
2) We replace its point cloud P with a primitive shape M, as described in Sec. III-A.
3) We extract surface planes Π from M by iteratively applying RANSAC [35].

For a set of part entity nodes V^p corresponding to an object, we estimate the structure of pt^p, *i.e.*, the optimal parent-child contact relations among the parts of an object $S^{p*} = \{s_{p,c}\}$ in terms of Eq. (1). We formulate an optimization problem to maximize the overall contact scores $\text{Cont}(\cdot,\cdot)$ while satisfying the constraints in Eq. (1):

$$\begin{aligned} S^{p*} &= \underset{S^p}{\text{argmax}} \sum_{s_{p,c}\in S^p} \max_{i,j} \left(\text{Cont}\left(U_p^i, U_c^j\right)\right), \\ \text{s.t. } &\text{Align}\left(\boldsymbol{\pi}_p^i, \boldsymbol{\pi}_c^j\right) \geqslant \theta_a, \ \forall p, c, \ \exists i, j, \\ &\text{Dist}\left(\boldsymbol{\pi}_p^i, \boldsymbol{\pi}_c^j\right) \leqslant \theta_d, \ \forall p, c, \ \exists i, j. \end{aligned} \tag{4}$$

We solve this optimization problem in two steps:

1) We construct a directed graph with nodes in V^p. By traversing all pairs of nodes, our algorithm adds an edge $s_{p,c}$ from node v_p to node v_c to the graph if they satisfy the constraints in Eq. (4), with the edge's weight set to $\max_{i,j}\left(\text{Cont}\left(U_p^i, U_c^j\right)\right)$.
2) We find the optimal parent-child relations S^{p*}. Although the constructed graph entails all possible contact relations among entities, it may not be in the form of a parse tree since the indegree of a node could be greater than 1 (*i.e.*, a node has multiple parents), violating the definition of a rooted tree. Finding the optimal parent-child relations is equivalent to finding a directed spanning tree of maximum weight in the constructed graph, known as an arborescence problem. We adopt Edmonds' algorithm [36] to solve this problem.

Next, we estimate parameterized joints $\mathcal{J}$ for all parent-child relations in S^{p*} by matching the primitive parts to a library of articulated templates. This involves determining the joint types, joint axes, and joint poses based on their semantic labels, parent-child relations, and geometries. For example, a microwave door should be connected to its base with a revolute joint, which is usually located at the rim of its base. Fig. 3 presents a complete example of estimating the kinematic relations among table parts.

C. Spatial refinement among parts

We can further perform a refinement process to adjust transformations in $\mathcal{J}$ so that parts forming parent-child pairs are better aligned. This step reduces penetration between parts.

The spatial refinement algorithm, detailed in Alg. 1, performs refinements between parts in a top-down manner, given the input parse tree pt^p of an object, to avoid conflicts. The function `getEdgeTransform` retrieves the relative transformation $T_{p,c}$ from the parse tree. Then, `getAlignedPlanes` pairwise compares surface planes in

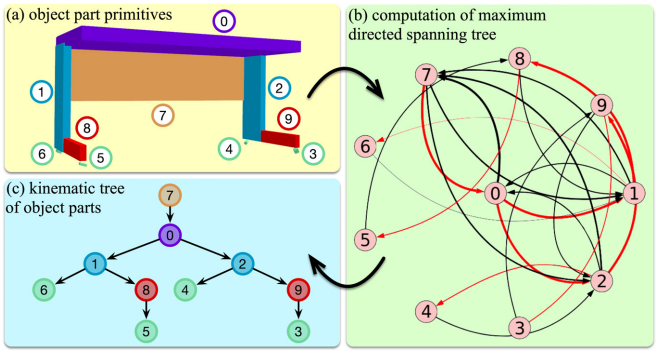


Fig. 3: **Kinematic relation estimation among parts.** (a) The set of primitive shapes that best match the part entities of a table. (b) Based on the largest contact score $\text{Cont}\left(U_p^i, U_c^j\right)$ between every pair of parts (indicated by the edge's weight), the most probable connectivity between parts can be found by computing the maximum directed spanning tree, *i.e.*, the red edges. (c) The computed kinematic relations among the parts from parent to child.

v_p and v_c, and selects roughly-aligned normal vectors of planes for downstream transformation refinement. Next, the function `refineTF` refines the rotation of v_c by computing a translation-free refinement transformation T_c^r that aligns a set of normal vectors X_c to another set X_p. Finally, `updateEdgeTransform` makes necessary updates to elements in pt^p (*i.e.*, $T_{p,c}$ in $\mathcal{J}$) using the refined transformation.

The optimization problem in `refineTF` is formulated as:

$$\begin{aligned} T^* &= \underset{T\in SE(3)}{\text{argmin}} \sum_{\boldsymbol{u}_i^p\in X_p, \boldsymbol{u}_i^c\in X_c} ||T\circ\boldsymbol{u}_i^c - \boldsymbol{u}_i^p||_2^2, \\ \text{s.t. } T &= \begin{bmatrix} R_{3\times3} & \mathbf{0} \\ \mathbf{0}^T & 1 \end{bmatrix}, \end{aligned} \tag{5}$$

where $R_{3\times3}$ is a rotation matrix, $\boldsymbol{u}_i^p$ and $\boldsymbol{u}_i^c$ are a correspondent pair of normal vectors close to each other in direction. This optimization problem is equivalent to a point set registration problem, for which we find the optimal solution using the Kabsch algorithm [37].

Finally, we combine the part-based representation pt^p for all objects into a single contact graph cg of the scene. Following Han *et al.* [8, 9], we build object entity nodes and estimate the inter-object supporting and proximal relations. Then, we refine the pose of each whole object based on the supporting relations. The resulting cg effectively organizes parts of all objects in the scene with kinematic information and can be converted into a kinematic tree in URDF format to directly support robot interactions.

IV. Experiments

Experiments demonstrate that our system successfully reconstructs part-level fine-grained interactive scenes from partial scans, yielding more details of the observed scenes compared with the baseline [9] that reconstructs scenes with object-level CAD replacement.

Dataset augmentation: Due to the lack of ground-truth object geometries and part segmentation in ScanNet, we augment the dataset with the information of the CAD models in PartNet [24] based on the annotations in Scan2CAD [38]. The kinematic joints of articulated objects are further ac-

Algorithm 1: Spatial refinement among parts

Input : a part-level parse tree pt^p
Output: pt^p with refined transformations

$q \leftarrow$ `Queue()`
// add children of root of pt to queue q
foreach $v_c \in pt^p.root.children$ **do**
  q.`push`(v_c)
while q *is not empty* **do**
  $v_c \leftarrow q$.`pop`()
  $v_p \leftarrow v_c.parent$
  // get transformation from v_p to v_c
  $T_{pc} \leftarrow pt^p$.`getEdgeTransform`(v_p, v_c)
  // find normal vectors of nearly aligned planes
  $X_p, X_c \leftarrow$ `getAlignedPlanes`(v_p, v_c)
  // compute the refinement transformation of v_c
  $T_c^r \leftarrow$ `refineTF`(X_c, X_p)
  // update the transformation from v_p to v_c
  $T_{pc} \leftarrow T_{pc} T_c^r$
  // update pt^p with the refined transformation
  pt^p.`updateEdgeTransform`(v_p, v_c, T_{pc})
  // add children of v_c to queue q
  foreach $v_{cc} \in v_c.children$ **do**
    q.`push`(v_{cc})
return pt^p

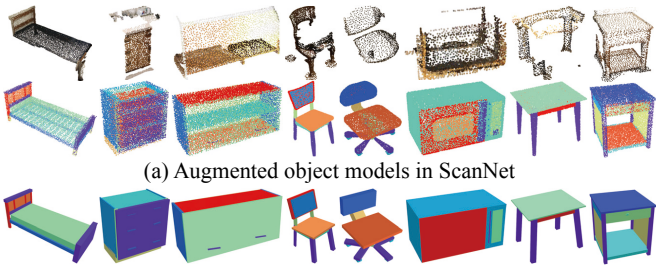
quired from PartNet-Mobility [39]. Fig. 4a shows some examples of augmented object models in ScanNet.

Implementation details: To detect the 3D objects from the point cloud of a scanned scene, we adopt the MLCVNet [40] as the front end of our system, which outputs a 3D bounding box for each detected object. This model was pre-trained on the ScanNet dataset following the same train/test split described in Xie *et al.* [40]. After retrieving the object point cloud inside the bounding box, we used StructureNet [41] to decompose the object into parts, which incorporated point cloud completion and outlier removal during the decomposition process. Of note, our system is modularized for future integration of more powerful 3D detection/completion models.

A. Part-level CAD replacement from partial 3D scan

Protocols: We evaluate our part-level CAD replacement against a baseline that replaces interactive CADs at the object level [9] based on two criteria, geometric similarity and plausibility of kinematic estimation. The evaluations were conducted using the synthetic scans from SceneNN [42] and real-world scans from ScanNet [43]. Specifically, we picked 7 scenes in SceneNN that were used in the baseline [9] and 8 scenes from ScanNet for the evaluations.

Geometric similarity: We use Chamfer distance and intersection over union (IoU) metrics to quantitatively evaluate the reconstruction results. Chamfer distance measures the point-wise distance between the surface structures of the reconstructed objects and the ground-truth scans, indicating their overall geometric similarity. IoU reflects how well the



(a) Augmented object models in ScanNet

(b) Assembled objects from primitive shapes

Fig. 4: **Examples of augmented objects in ScanNet.** (a) Incomplete objects in ScanNet (top) are augmented by corresponding objects in PartNet with part segmentation (bottom). (b) Objects could be assembled from primitive shapes in terms of part segmentation.

reconstructed objects align with the ground-truth objects in terms of poses and sizes. Objects are normalized to a unit box for Chamfer distance computation, and replaced objects are voxelized into a 32^3 grid for comparison with the ground-truth voxel grid for IoU.

There were three types of input scans studied in evaluations: original RGB-D scans (*original*), scans with point cloud completion and part decomposition (*completed*), and annotated scans in the augmented dataset that serve as the ground-truth (*annotation*). Our method was not evaluated on the *original* scans as they lack part-level information. Also, sequences in the SceneNN dataset cannot be augmented as *annotation* scans; thus, they are not evaluated in the corresponding setup either.

The results are summarized in Tab. I. Looking at the *original* and *completed* groups, our part-based method outperforms the baseline with lower Chamfer distance and higher IoU for most sequences. This indicates the effectiveness of part-level CAD replacement for reconstructing interactive scenes and the importance of unitizing a point cloud completion model to handle noisy and incomplete scans. On the other hand, reconstructions from augmented scans with object part segmentation (see Fig. 4b for some examples) are significantly improved (*completed vs. annotation* groups), suggesting that perception noise remains a primary challenge.

Kinematic structure of object parts: Evaluating the plausibility of the estimated kinematic structure of object parts is challenging due to its ambiguity. The same object can be represented by different kinematic structures (see Fig. 5a). To address this, we manually annotate the kinematic structures of different objects. For each pair of parts in an object, we connect them with an undirected edge if we believe there is a contact relation between them. We use the mean average precision (mAP) metric to measure the alignment between the human-annotated kinematic structure and the estimated structure based on the undirected contact relations between parts (see Fig. 3b). The mAP metric summarizes how accurately the relations between parts (edges) are predicted.

Tab. II summarizes the results. Our method successfully estimates the kinematic structures of 5 object categories with high articulation, achieving mAP values close to 1.0, indicating a nearly perfect match.

TABLE I: **Quantitative comparison of geometric similarity using Chamfer distance (Cdist, the lower the better) and IoU (the higher the better)** Bold values indicate the best results between object-level baseline [9] and our part-level CAD replacement using original and completed inputs, while underlined values indicate the best results using the annotated inputs.

	CAD replacement	input format	SceneNN seq. ID 011	030	061	078	086	096	223	ScanNet seq. ID 0002	0003	0092	0157	0215	0335	0560	0640
Cdist.	object-level	original	**0.189**	0.759	0.431	0.634	0.588	0.508	0.462	0.573	0.776	0.392	0.559	0.379	0.604	0.329	0.752
		completed	0.329	0.378	0.483	0.413	0.601	0.329	0.619	0.580	0.710	0.321	0.554	0.256	0.663	0.307	0.651
		annotation	-	-	-	-	-	-	-	0.416	0.590	0.282	0.321	0.143	0.519	0.322	0.554
	part-level	completed	0.205	**0.207**	**0.310**	**0.187**	**0.210**	**0.177**	**0.169**	**0.202**	**0.163**	**0.216**	**0.239**	**0.192**	**0.174**	**0.190**	**0.183**
		annotation	-	-	-	-	-	-	-	0.101	0.119	0.092	0.087	0.076	0.086	0.098	0.089
IoU	object-level	original	0.109	0.034	0.063	0.028	0.042	0.047	0.021	0.021	0.013	0.034	0.028	0.033	0.021	0.101	0.012
		completed	0.030	0.034	0.087	0.033	0.016	0.052	0.040	0.014	0.076	0.128	0.027	0.065	0.017	0.057	0.018
		annotation	-	-	-	-	-	-	-	0.056	0.100	0.116	0.170	0.196	0.067	0.133	0.119
	part-level	completed	**0.125**	**0.118**	**0.215**	**0.157**	**0.156**	**0.134**	**0.113**	**0.191**	**0.224**	**0.131**	**0.089**	**0.192**	**0.179**	**0.159**	**0.190**
		annotation	-	-	-	-	-	-	-	0.383	0.540	0.478	0.665	0.361	0.548	0.467	0.614

TABLE II: **mAP of the estimated kinematic structures among object parts.**

Category	Chair	Table	Microwave	Cabinet	Bed
mAP	0.9247	0.8292	0.9741	0.9592	0.9785

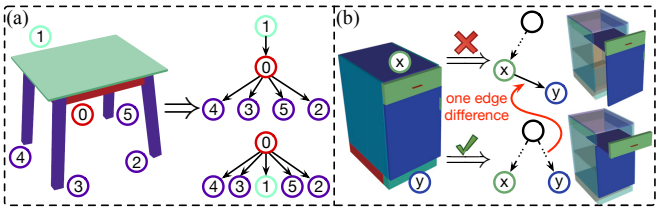


Fig. 5: **Evaluation of kinematic relations.** (a) Different kinematic trees represent the parts of a table. (b) One error in kinematic structure estimation results in undesired articulation.

Discussions: The results in Tab. I provide an indirect assessment of the kinematic transformation among parts, while Tab. II verifies the accuracy of estimating their parent-child relations. Although the results appear promising individually, the complex nature of kinematic relations means that even a small error can lead to significant issues. Fig. 5 showcases some typical cases, highlighting the ongoing challenge of estimating kinematic relations.

B. Interactive scene reconstruction

Fig. 6 provides a qualitative comparison of interactive scene reconstructions from ScanNet using object-level [9] and part-level (ours) approaches. The reconstructed scenes enable robot TAMP by leveraging the encoded kinematic relations. Our method achieves a more precise reconstruction (Fig. 6c) compared to the baseline [9] (Fig. 6b), as indicated by the ground-truth segmentation of the 3D scans (Fig. 6a).

We highlight successful and failed samples in Fig. 6d to better understand our method's performance. Failures often occur due to outliers in the part decomposition module, leading to incorrect part replacement and alignment, or when the completion module struggles with overly incomplete input point clouds (*e.g.*, a single surface of a fridge) due to the limited information available. Many of these failure cases stem from perceptual limitations when dealing with unobserved or partially observed environments [44].

Furthermore, we demonstrate that the reconstructed interactive scenes can be converted to URDF and imported into ROS for robot-scene interactions (Fig. 6e). The resulting contact graph containing object part geometry and kinematic relations acts as a bridge between the robot's scene perception, understanding, and TAMP [11, 21].

V. Conclusion and Discussion

In this work, we developed a system for reconstructing interactive scenes by replacing object point clouds with CAD models that enable robot interactions. In contrast to previous approaches focused on object-level CAD replacement [8, 9], our system takes a part-level approach by decomposing objects and aligning primitive shapes to them. We achieved a more detailed and precise representation of the scene by estimating the kinematic relations between adjacent parts, including joint types (fixed, prismatic, or revolute) and parameters.

To handle noisy and partial real scans, our system incorporates a point cloud completion module to recover observed object parts before performing CAD replacement. The estimated kinematics of objects and the scene are aggregated and composed into a graph-based representation, which can be converted to a URDF. This representation allows for reconstructing interactive scenes that closely match the actual scenes, providing a "mental space" [44, 45] for robots to engage in TAMP and anticipate action effects before execution. This capability is crucial for the success of robots in long-horizon sequential tasks.

Moreover, our system has potential applications beyond interactive scene reconstruction. It can be utilized to digitize real environments for virtual reality, creating in-situ simulations for robot planning and training [46, 47], and facilitate the understanding of human-object interactions [48–50], among other downstream applications.

In conclusion, our part-level CAD replacement system significantly enhances the reconstruction of interactive scenes by capturing finer details and improving precision. The resulting scenes serve as a foundation for robot cognition and planning, enabling robots to navigate complex tasks successfully. Additionally, the versatility of our system opens up possibilities for various applications in virtual reality, robot planning, training, and human-object interaction studies.

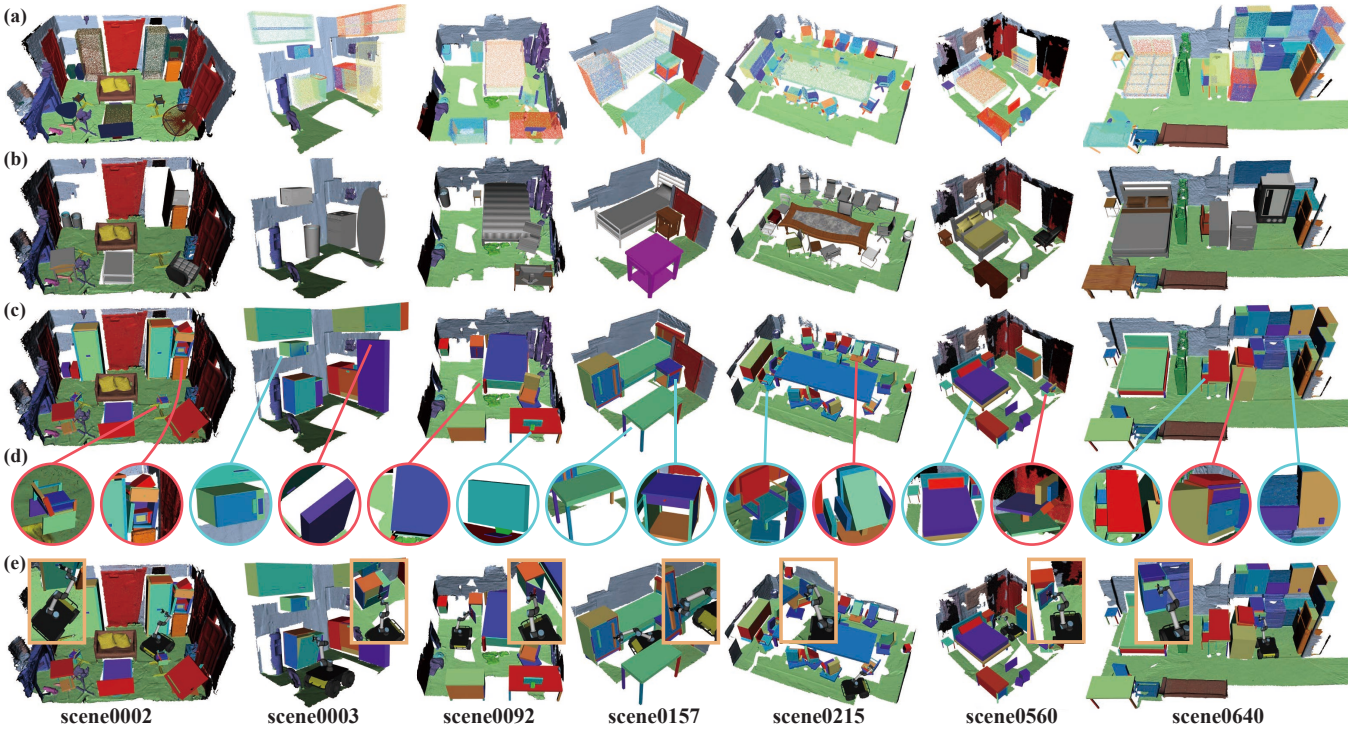


Fig. 6: **Qualitative comparisons of reconstructed interactive scenes at the object level [9] and the part level (ours)**. (a) Ground-truth part-level segmentation of 7 real-world scans augmented from ScanNet [43]. (b) Object-level CAD replacement preserves object semantics and overall dimensions but fails to reflect geometries accurately. (c) Our part-level CAD replacement better reflects object geometries by assembling objects from primitive shapes. (d) Successful and failed part replacements/assemblies are highlighted with blue and red circles, respectively. (e) The resulting interactive scenes enable fine-grained robot interactions using TAMP).

Limitations and future work: Reconstructing interactive scenes, particularly at the part level, poses significant challenges that require substantial research efforts. We acknowledge the following limitations and identify potential avenues for future work.

First, real-world indoor scenes are inherently complex and are often subject to clustering, occlusions, and sensor noise. Even with scan completion methods, 2.5D RGB-D scans may still be noisy or incomplete, hindering a comprehensive understanding of the scene. Addressing this limitation requires further advancements in scan completion techniques to improve the quality of the input data.

Second, estimating an object's kinematics solely from static observations during the reconstruction process is inherently ambiguous. Existing approaches often rely on object motion cues to disambiguate kinematic relationships. However, these methods may struggle to scale effectively in larger-scale real scenes. Future research should explore novel strategies for resolving kinematic ambiguity, potentially by leveraging both static and dynamic cues or exploiting tactile information [51] to enhance the overall accuracy.

Third, our current system treats the interior structure of contained spaces (*e.g.*, the space inside a cabinet) as a solid due to its unobservable nature. Human cognition excels at filling in perceptual gaps, but our system lacks this capability. Building upon our presented system, future work could integrate advanced perception and reasoning models to endow robots with similar cognitive abilities, enabling them to operate better within complex environments. Additionally, it would be valuable to develop methods that allow robots to actively probe the environment and refine reconstructed scenes, leading to more robust and detailed representations.

Acknowledgement: This work is supported in part by the National Key R&D Program of China (2021ZD0150200) and the Beijing Nova Program.

REFERENCES

[1] A. Pronobis and P. Jensfelt, "Large-scale semantic mapping and reasoning with heterogeneous modalities," in *International Conference on Robotics and Automation (ICRA)*, 2012.

[2] S. Yang and S. Scherer, "Cubeslam: Monocular 3-d object slam," *Transactions on Robotics (T-RO)*, vol. 35, no. 4, pp. 925–938, 2019.

[3] M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (slam) problem," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 229–241, 2001.

[4] J. McCormac, A. Handa, A. Davison, and S. Leutenegger, "Semanticfusion: Dense 3d semantic mapping with convolutional neural networks," in *International Conference on Robotics and Automation (ICRA)*, 2017.

[5] D.-C. Hoang, A. J. Lilienthal, and T. Stoyanov, "Panoptic 3d mapping and object pose estimation using adaptively weighted semantic information," *IEEE Robotics and Automation Letters (RA-L)*, vol. 5, no. 2, pp. 1962–1969, 2020.

[6] M. Grinvald, F. Furrer, T. Novkovic, J. J. Chung, C. Cadena, R. Siegwart, and J. Nieto, "Volumetric instance-aware semantic mapping and 3d object discovery," *IEEE Robotics and Automation Letters (RA-L)*, vol. 4, no. 3, pp. 3037–3044, 2019.

[7] J. McCormac, R. Clark, M. Bloesch, A. Davison, and S. Leutenegger, "Fusion++: Volumetric object-level slam," in *International Conference on 3D Vision (3DV)*, 2018.

[8] M. Han, Z. Zhang, Z. Jiao, X. Xie, Y. Zhu, S.-C. Zhu, and H. Liu, "Reconstructing interactive 3d scenes by panoptic mapping and cad model alignments," in *International Conference on Robotics and Automation (ICRA)*, 2021.

[9] M. Han, Z. Zhang, Z. Jiao, X. Xie, Y. Zhu, S.-C. Zhu, and H. Liu, "Scene reconstruction with functional objects for robot autonomy," *International Journal of Computer Vision (IJCV)*, vol. 130, no. 12, pp. 2940–2961, 2022.

[10] L. P. Kaelbling and T. Lozano-Pérez, "Hierarchical task and motion planning in the now," in *International Conference on Robotics and Automation (ICRA)*, 2011.

[11] Z. Jiao, Y. Niu, Z. Zhang, S.-C. Zhu, Y. Zhu, and H. Liu, "Sequential manipulation planning on scene graph," in *International Conference on Intelligent Robots and Systems (IROS)*, 2022.

[12] Y. Su, J. Li, Z. Jiao, M. Wang, C. Chu, H. Li, Y. Zhu, and H. Liu, "Sequential manipulation planning for over-actuated unmanned aerial manipulators," in *International Conference on Intelligent Robots and Systems (IROS)*, 2023.

[13] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *Transactions on Robotics (T-RO)*, vol. 32, no. 6, pp. 1309–1332, 2016.

[14] S. Huang, S. Qi, Y. Xiao, Y. Zhu, Y. N. Wu, and S.-C. Zhu, "Cooperative holistic scene understanding: Unifying 3d object, layout, and camera pose estimation," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[15] C. Jiang, S. Qi, Y. Zhu, S. Huang, J. Lin, L.-F. Yu, D. Terzopoulos, and S.-C. Zhu, "Configurable 3d scene synthesis and 2d image rendering with per-pixel ground truth using stochastic grammars," *International Journal of Computer Vision (IJCV)*, vol. 126, no. 9, pp. 920–941, 2018.

[16] Y. Chen, S. Huang, T. Yuan, S. Qi, Y. Zhu, and S.-C. Zhu, "Holistic++ scene understanding: Single-view 3d holistic scene parsing and human pose estimation with human-object interaction and physical commonsense," in *International Conference on Computer Vision (ICCV)*, 2019.

[17] I. Armeni, Z.-Y. He, J. Gwak, A. R. Zamir, M. Fischer, J. Malik, and S. Savarese, "3d scene graph: A structure for unified semantics, 3d space, and camera," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[18] J. Wald, H. Dhamo, N. Navab, and F. Tombari, "Learning 3d semantic scene graphs from 3d indoor reconstructions," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[19] A. Rosinol, A. Gupta, M. Abate, J. Shi, and L. Carlone, "3D dynamic scene graphs: Actionable spatial perception with places, objects, and humans," in *Robotics: Science and Systems (RSS)*, 2020.

[20] Z. Jiao, Z. Zhang, W. Wang, D. Han, S.-C. Zhu, Y. Zhu, and H. Liu, "Efficient task planning for mobile manipulation: a virtual kinematic chain perspective," in *International Conference on Intelligent Robots and Systems (IROS)*, 2021.

[21] Z. Jiao, Z. Zhang, X. Jiang, D. Han, S.-C. Zhu, Y. Zhu, and H. Liu, "Consolidating kinematic models to promote coordinated mobile manipulations," in *International Conference on Intelligent Robots and Systems (IROS)*, 2021.

[22] Y. Weng, H. Wang, Q. Zhou, Y. Qin, Y. Duan, Q. Fan, B. Chen, H. Su, and L. J. Guibas, "Captra: Category-level pose tracking for rigid and articulated objects from point clouds," in *International Conference on Computer Vision (ICCV)*, 2021.

[23] J. Huang, H. Wang, T. Birdal, M. Sung, F. Arrigoni, S.-M. Hu, and L. J. Guibas, "Multibodysync: Multi-body segmentation and motion estimation via 3d scan synchronization," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[24] K. Mo, S. Zhu, A. X. Chang, L. Yi, S. Tripathi, L. J. Guibas, and H. Su, "Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[25] A. Bokhovkin, V. Ishimtsev, E. Bogomolov, D. Zorin, A. Artemov, E. Burnaev, and A. Dai, "Towards part-based understanding of rgb-d scans," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[26] C. Xu, Y. Chen, H. Wang, S.-C. Zhu, Y. Zhu, and S. Huang, "Partafford: Part-level affordance discovery from 3d objects," *arXiv preprint arXiv:2202.13519*, 2022.

[27] X. Liu, J. Zhang, R. Hu, H. Huang, H. Wang, and L. Yi, "Self-supervised category-level articulated object pose estimation with part-level se (3) equivariance," in *International Conference on Learning Representations (ICLR)*, 2023.

[28] G. Liu, Q. Sun, H. Huang, C. Ma, Y. Guo, L. Yi, H. Huang, and R. Hu, "Semi-weakly supervised object kinematic motion prediction," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[29] S.-C. Zhu, C.-E. Guo, Y. Wang, and Z. Xu, "What are textons?," *International Journal of Computer Vision (IJCV)*, vol. 62, pp. 121–143, 2005.

[30] Y. N. Wu, Z. Si, H. Gong, and S.-C. Zhu, "Learning active basis model for object detection and recognition," *International Journal of Computer Vision (IJCV)*, vol. 90, no. 2, pp. 198–235, 2010.

[31] A. T. Miller, S. Knoop, H. I. Christensen, and P. K. Allen, "Automatic grasp planning using shape primitives," in *International Conference on Robotics and Automation (ICRA)*, 2003.

[32] Y. Lin, C. Tang, F.-J. Chu, and P. A. Vela, "Using synthetic data and deep networks to recognize primitive shapes for object grasping," in *International Conference on Robotics and Automation (ICRA)*, 2020.

[33] D. Dimitrov, C. Knauer, K. Kriegel, and G. Rote, "Bounds on the quality of the pca bounding boxes," *Computational Geometry*, vol. 42, no. 8, pp. 772–789, 2009.

[34] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 14, no. 2, pp. 239–256, 1992.

[35] Y. Taguchi, Y.-D. Jian, S. Ramalingam, and C. Feng, "Point-plane slam for hand-held 3d sensors," in *International Conference on Robotics and Automation (ICRA)*, 2013.

[36] J. Edmonds *et al.*, "Optimum branchings," *Journal of Research of the National Bureau of Standards B*, vol. 71, no. 4, pp. 233–240, 1967.

[37] W. Kabsch, "A solution for the best rotation to relate two sets of vectors," *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, vol. 32, no. 5, pp. 922–923, 1976.

[38] A. Avetisyan, M. Dahnert, A. Dai, M. Savva, A. X. Chang, and M. Nießner, "Scan2cad: Learning cad model alignment in rgb-d scans," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[39] F. Xiang, Y. Qin, K. Mo, Y. Xia, H. Zhu, F. Liu, M. Liu, H. Jiang, Y. Yuan, H. Wang, *et al.*, "Sapien: A simulated part-based interactive environment," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[40] Q. Xie, Y.-K. Lai, J. Wu, Z. Wang, Y. Zhang, K. Xu, and J. Wang, "Mlcvnet: Multi-level context votenet for 3d object detection," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[41] K. Mo, P. Guerrero, L. Yi, H. Su, P. Wonka, N. J. Mitra, and L. J. Guibas, "Structurenet: hierarchical graph networks for 3d shape generation," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 6, pp. 1–19, 2019.

[42] B.-S. Hua, Q.-H. Pham, D. T. Nguyen, M.-K. Tran, L.-F. Yu, and S.-K. Yeung, "Scenenn: A scene meshes dataset with annotations," in *International Conference on 3D Vision (3DV)*, 2016.

[43] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[44] Y. Zhu, T. Gao, L. Fan, S. Huang, M. Edmonds, H. Liu, F. Gao, C. Zhang, S. Qi, Y. N. Wu, *et al.*, "Dark, beyond deep: A paradigm shift to cognitive ai with humanlike common sense," *Engineering*, vol. 6, no. 3, pp. 310–345, 2020.

[45] L. Fan, M. Xu, Z. Cao, Y. Zhu, and S.-C. Zhu, "Artificial social intelligence: A comparative and holistic view," *CAAI Artificial Intelligence Research*, vol. 1, no. 2, pp. 144–160, 2022.

[46] X. Xie, H. Liu, Z. Zhang, Y. Qiu, F. Gao, S. Qi, Y. Zhu, and S.-C. Zhu, "Vrgym: A virtual testbed for physical and interactive ai," in *Proceedings of the ACM TURC*, 2019.

[47] F. Xia, W. B. Shen, C. Li, P. Kasimbeg, M. E. Tchapmi, A. Toshev, R. Martín-Martín, and S. Savarese, "Interactive gibson benchmark: A benchmark for interactive navigation in cluttered environments," *IEEE Robotics and Automation Letters (RA-L)*, vol. 5, no. 2, pp. 713–720, 2020.

[48] P. Li, T. Liu, Y. Li, Y. Zhu, Y. Yang, and S. Huang, "Gendexgrasp: Generalizable dexterous grasping," in *International Conference on Robotics and Automation (ICRA)*, 2023.

[49] N. Jiang, T. Liu, Z. Cao, J. Cui, Y. Chen, H. Wang, Y. Zhu, and S. Huang, "Chairs: Towards full-body articulated human-object interaction," *arXiv preprint arXiv:2212.10621*, 2022.

[50] H. Liu, Z. Zhang, Z. Jiao, Z. Zhang, M. Li, C. Jiang, Y. Zhu, and S.-C. Zhu, "Reconfigurable data glove for reconstructing physical and virtual grasps," *Engineering*, 2023.

[51] W. Li, M. Wang, J. Li, Y. Su, D. K. Jha, X. Qian, K. Althoefer, and H. Liu, "L^3 f-touch: A wireless gelsight with decoupled tactile and three-axis force sensing," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 5148–5155, 2023.