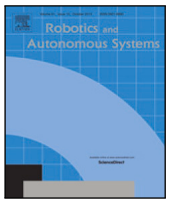
Robotics and Autonomous Systems 201 (2026) 105442

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot



Generalizable task-oriented object grasping through LLM-guided ontology and similarity-based planning

Hao Chen *, Takuya Kiyokawa, Weiwei Wan, Kensuke Harada

Graduate School of Engineering Science, The University of Osaka, 1-3, Machikaneyamacho, Toyonaka, Osaka, 560-8531, Japan

ARTICLE INFO

Keywords:
Task-oriented grasping
Ontology
Planning under uncertainty

ABSTRACT

Task-oriented grasping (TOG) is more challenging than simple object grasping because it requires precise identification of object parts and careful selection of grasping areas to ensure effective and robust manipulation. While recent approaches have trained large-scale vision-language models to integrate part-level object segmentation with task-aware grasp planning, their instability in part recognition and grasp inference limits their ability to generalize across diverse objects and tasks. To address this issue, we introduce a novel, geometry-centric strategy for more generalizable TOG that does not rely on semantic features from visual recognition, effectively overcoming the viewpoint sensitivity of model-based approaches. Our main proposals include: (1) an object-part-task ontology for functional part selection based on intuitive human commands, constructed using a Large Language Model (LLM); (2) a sampling-based geometric analysis method for identifying the selected object part from observed point clouds, incorporating multiple point distribution and distance metrics; and (3) a similarity matching framework for imitative grasp planning, utilizing similar known objects with pre-existing segmentation and grasping knowledge as references to guide the planning for unknown targets. We validate the high accuracy of our approach in functional part selection, identification, and grasp generation through real-world experiments. Additionally, we demonstrate the method's generalization capabilities to novel-category objects by extending existing ontological knowledge, showcasing its adaptability to a broad range of objects and tasks.

1. Introduction

Robotic grasping is advancing beyond conventional pick-and-place operations toward more human-like behaviors, where understanding *part affordance* [1,2] is crucial for achieving task-oriented grasping (TOG). For instance, a robot is expected to identify and grasp the *handle* of a cup in a *pouring* task. In real-world scenarios, however, human instructions are often intuitive and semantically rich, extending far beyond simple action labels like *pouring*. As a result, the accurate interpretation of user intentions becomes essential for both human-guided manipulation and human–robot collaboration tasks. Moreover, the reliable recognition of object functional parts from partial observation while ensuring the generation of high-quality grasps remains a significant challenge, leaving this research problem largely unsolved.

More than a decade ago, the concept of *semantic grasping* was introduced [3,4] to enable high-level manipulation considering task constraints, rather than focusing solely on basic grasping. However, at the time, even with the aid of vision and tactile sensors, analytical approaches struggled to generate high-quality grasps without human supervision and showed limited generalization. With the rapid progress in deep learning and image segmentation, more advanced strategies based on visual recognition have emerged [5–7], demonstrating notable performance in planning task-oriented grasps across a wide variety of objects. Nevertheless, these vision-based methods typically represent target tasks using predefined cues such as *grasp from the handle* or simplistic labels like *handover* or *cut*, lacking the ability to translate contextual human instructions into specific robotic actions. This interaction challenge has been widely addressed in recent years with the advent of vision-language models (VLMs) [8–10]. By leveraging high-dimensional language embeddings and attention-based learning frameworks, robots are now increasingly capable of responding to natural human commands. However, several limitations persist in current research: (1) low grasping DoF [8]; (2) reliance on complete mesh models or point clouds [9]; and (3) significant performance degradation when generalizing to unseen objects [10].

To overcome these limitations, we propose a novel strategy for TOG that integrates an object-part-task ontology guided by large language models (LLMs) with a grasp detection method based on similarity

* Corresponding author.
E-mail address: chen@sys.es.osaka-u.ac.jp (H. Chen).

https://doi.org/10.1016/j.robot.2026.105442
Received 30 June 2025; Received in revised form 26 January 2026; Accepted 21 March 2026
Available online 25 March 2026

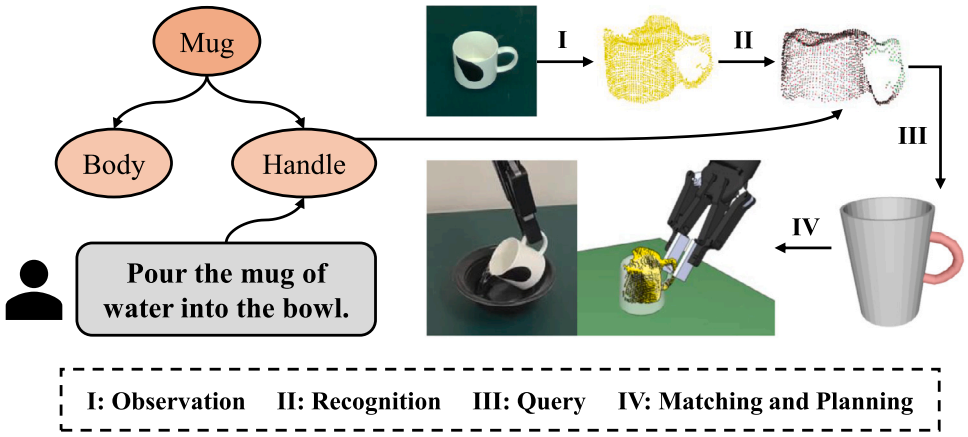


Fig. 1. Demonstration of our method applied to a *pouring* task with an unseen mug.

matching with known templates. This approach enables accurate interpretation of intuitive human instructions and robust 6-DoF grasp generation by leveraging references from similar pre-existing instances. An example application of our proposed method is shown in Fig. 1. Given a previously unseen object and a natural human instruction, we first process the semantic information by associating the human instruction with an existing ontology and identifying the functional part of the target object in the given task. Then, we utilize the geometric information obtained from single-view observation and employ a sampling-based method to recognize the corresponding functional part, represented as a point cluster. Based on this recognition result, we query a similar model template containing prior segmentation and grasping knowledge as a reference. Finally, through a matching and planning process assisted by the template, a high-quality 6-DoF grasp pose is generated and executed to successfully accomplish the task. The effectiveness of our method is validated by extensive real-world experiments, notably demonstrating strong generalization to novel-category objects via the scalable integration of LLM-guided ontology and similarity-based planning.

A recent study, ShapeGrasp [11], introduces a similar TOG framework that also leverages LLMs and knowledge graphs for grasp inference. A common challenge we have addressed is the accurate segmentation of objects into functionally meaningful parts. While ShapeGrasp relies heavily on empirically defined thresholds for object geometric decomposition, our method achieves significantly greater robustness through a sampling-based strategy that avoids extensive manual parameter tuning. In addition, their approach is limited to top-down grasps positioned at object centroids, whereas our framework generates full 6-DoF grasps that are explicitly aware of contact stability. The idea of using similar models as templates originates from our earlier work [12], which addressed novel object pick-and-place by referencing an existing object database. However, its exclusive reliance on global object similarity is insufficient to address the nuanced demands of TOG scenarios, which constitutes the central focus of this research.

Our main contributions can be summarized as:

- We propose an object-part-task ontology guided by LLMs that efficiently and accurately maps intuitive human instructions to the corresponding functional parts of target objects.
- We introduce a geometry-based part recognition method that is robust to viewpoint variations by leveraging similar model templates as references for functional part matching and imitative grasp planning.
- We optimize the quality of generated grasps from similar references through a combination of local-to-global point cloud registration and stability-aware positional adjustment.

2. Related works

2.1. Unseen object grasping

Grasping research has explored strategies for handling unseen objects for decades, driven by the need to address the complexity of real-world environments, which are diverse, dynamic, and only partially understood. Among existing studies, model-based approaches using deep learning [13–17] or reinforcement learning [18–20] dominate the field over model-free approaches based on geometric analysis [21,22]. However, each approach has its strengths and limitations. Model-based methods exhibit high performance in specific grasping tasks through large-scale training, but are computationally expensive and sensitive to environmental changes. In contrast, model-free methods are cost-effective and robust across varying environments, but struggle to generalize to diverse object types.

To harness the benefits of model-free methods while enhancing generalizability, we investigate a promising direction based on similarity matching [12,23,24]. The core idea of this approach is to leverage prior knowledge of similar known objects to guide the grasping of unknown targets. Inspired by [24], we find that integrating similarity matching with ontological relationships can effectively extend the method's capabilities to tackle TOG beyond basic object handling.

2.2. Task-oriented object grasping

High-level manipulation tasks require robots not only to grasp objects successfully, but also to grasp at appropriate positions to ensure both safety and functionality. To this end, TOG has been extensively studied, evolving from analytical approaches [3,4] to vision-based approaches [5–7], and more recently to vision-language-based approaches [8–10]. Here, we highlight a few recent notable works. CAGE [25] is a representative affordance-aware grasping system considering both object and task constraints. GCNGrasp [7] constructs a knowledge graph to train a Graph Convolutional Network, enabling the generalization of task-oriented grasps from predefined instances to novel concepts. OS-TOG [26] and Robo-ABC [27] propose matching frameworks that utilize database objects with labeled affordances to guide the grasping of novel objects. While conceptually similar to our approach, their methods are limited to 2D image matching, which restricts their ability to transfer 6-DoF grasping knowledge. Although all of these approaches effectively achieve generalized TOG, they do not incorporate language models and are therefore limited to simplistic task labels such as *handover* or *cut*.

To address more complex human instructions, VLMs have increasingly been adopted for contextual understanding. LERF-TOGO [28] integrates CLIP embeddings [29] with DINO features [30] to train a

multi-scale VLM capable of generating grasps for specific object parts based on language queries. GraspSplats [31] introduces a Gaussian-based feature representation for real-time object motion tracking and enables dynamic language-guided manipulation. FoundationGrasp [32] leverages open-ended knowledge from foundation models (both LLMs and VLMs) to learn generalizable TOG skills. While these approaches successfully associate contextual language inputs with spatial motion outputs, they struggle to achieve high task success rates due to the complexity inherent in integrating vision and language within cross-domain learning frameworks. To cope with this issue, we draw inspiration from ontology-based grasping methods [33,34], which defines multiple layers of knowledge representations (e.g., object, task, affordance) to support part-aware object manipulation. Rather than fusing all features into a unified model, we adopt a more modular approach: processing visual and verbal inputs separately and linking them through ontological reasoning over a predefined knowledge graph.

2.3. LLMs in task-oriented grasping

LLMs are a crucial tool in our approach. In recent years, the emergence of high-performance LLMs such as *ChatGPT* and *Claude* has significantly advanced TOG strategies. For instance, GraspGPT [35] leverages LLMs to generate text descriptions for both objects and tasks, which are then encoded with object point clouds and grasp poses to select high-quality task-oriented grasps through a transformer decoder. ATLA [36] utilizes LLMs to generate rich semantic knowledge, accelerating tool learning in diverse manipulation tasks. Grasp-Anything [37] employs LLMs to generate scene descriptions, which are integrated with image inputs using a diffusion model trained on a large dataset and applied to language-driven grasp detection. While these approaches commonly use LLMs to generate instructive text to enhance the learning process, we find that directly regressing grasp poses from text embeddings can lead to unstable performance due to the diversity of LLM descriptions and the uncertainty in real-world observations. Therefore, we propose a sequential strategy that first queries the target object parts using the strong interpretability of LLMs, and then focuses on object geometry for grasp planning. This separation effectively mitigates the complexity of handling linguistic and visual features simultaneously.

3. Methods

3.1. Overview

Our goal is to enable a robot to execute task-oriented grasps based on intuitive human instructions. Given an instruction, the robot uses an RGB-D camera to detect the target object, identify the relevant functional part, and plan feasible grasps that fulfill the intended task. The overall process consists of three core components: *Human Instruction Interpretation*, *Functional Part Recognition*, and *Task-Oriented Grasp Planning*. A high-level overview of our TOG system is illustrated in Fig. 2. On the left side, we process RGB-D images captured by a wrist-mounted camera through mask extraction and pixel-to-point projection, resulting in a partial point cloud of the target object. On the right side, we process human instructions using an LLM-guided ontology to identify the relevant functional part, which is used to guide the subsequent matching and planning stages. Meanwhile, we prepare a database of model templates, each segmented into multiple functional parts according to the defined ontology. In addition, each template is accompanied by a corresponding segmented point cloud and a set of preplanned robust grasps.

Based on the identified functional part and the pre-existing template point clouds, we adopt a sampling-clustering-matching strategy to recognize the corresponding part (a point cluster) within the observed object point cloud. We then perform local-to-global point cloud registration using both the full template point cloud and the selected point cluster to align the observed object with the best-matching model template, yielding a transformation matrix. This matrix is used to transfer preplanned grasping knowledge from the model template to the observed object. Finally, a dedicated optimization process generates robust task-oriented grasps tailored to the given instructions and object geometry for execution.

3.2. Template database preparation

The model template database used in our pipeline consists of three key components: pre-segmented object meshes, corresponding point clouds, and pre-planned grasps. While the point clouds and pre-planned grasps can be automatically generated using voxel grid downsampling and an existing antipodal grasp sampling method [38], the only manual step required in constructing the database is the part-aware segmentation of object meshes. However, this process is neither difficult nor time-consuming, due to the minimal requirement for precise labeling—the template serves merely as a reference rather than a direct target. Unlike pixel-level 2D image segmentation, our 3D mesh segmentation can be easily achieved by rotating the mesh to a suitable pose and using a simple rectangular selection box to roughly isolate the desired object part. This can be done using open-source platforms such as MeshLab [39]. In practice, each mesh can be segmented in under 30 s.

It is also worth noting that our method does not heavily depend on the size or diversity of the template database. As demonstrated in Experiment 4.5, even a small database with fewer than 10 model templates is sufficient to handle a wide variety of novel objects. This strong generalizability significantly reduces the preparatory workload required for database construction in real-world task scenarios, thereby keeping the data collection cost minimal.

3.3. Human instruction interpretation

In human-guided manipulation and human–robot collaboration tasks, accurately interpreting human instructions is essential for generating appropriate robot actions. However, the rich semantic knowledge embedded in contextual instructions is difficult to extract without a powerful inference model. Recent advances in LLMs have made this level of understanding feasible. In the context of TOG, a key challenge is determining *which part of the object the robot should grasp*—the central problem addressed in this work. To associate human instructions with relevant object parts, we propose an LLM-guided object-part-task ontology.

As shown in Fig. 3, the proposed ontology consists of two components: offline and online. The offline ontology predefines object classes with multi-layer part hierarchies. For instance, a *mug* is divided into *handle* and *body*, while the *body* can be further subdivided into *inside* and *outside*. Different object parts correspond to different grasping strategies, which are flexibly selected according to task requirements. The online ontology handles user-provided instructions, which are not previously known. To establish connections between task instructions and relevant functional parts, we leverage the latest LLM, GPT-4o [40], to bridge the online and offline components. The prompt used for this association is originally structured as follows:

Given the following ontology ... A robot is given the following commands ... Question: Which part(s) of the object should the robot grasp?

Despite the strong interpretability of LLMs, we observe that their answers can vary across trials, lacking accuracy and consistency. To address this, we first adopt the prompt optimization method from [41], which uses LLMs themselves as prompt optimizers. Implementation details are provided in Appendix A. While the optimized prompts incorporating task constraints improve answer correctness, they still fall short of human-level reliability. Through a large number of trials, we identify two prompt design principles that significantly enhance LLM performance: (1) Step-by-step reasoning; (2) Using an answer template. In practice, we append a step-by-step reasoning template to the end of each prompt, following this structure:

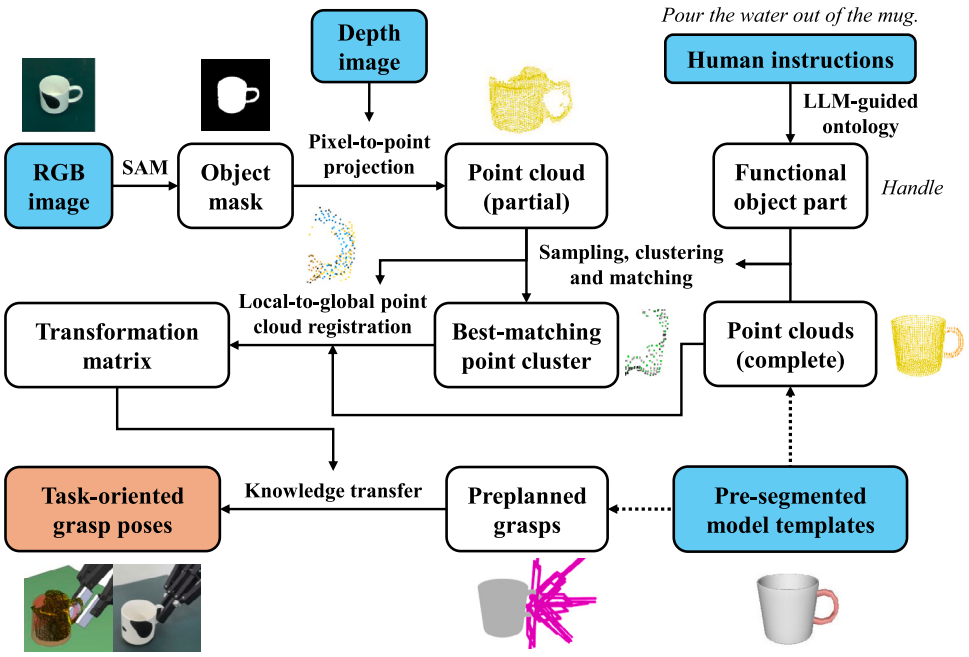


Fig. 2. Proposed TOG system overview. Blue indicates inputs; orange indicates outputs. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

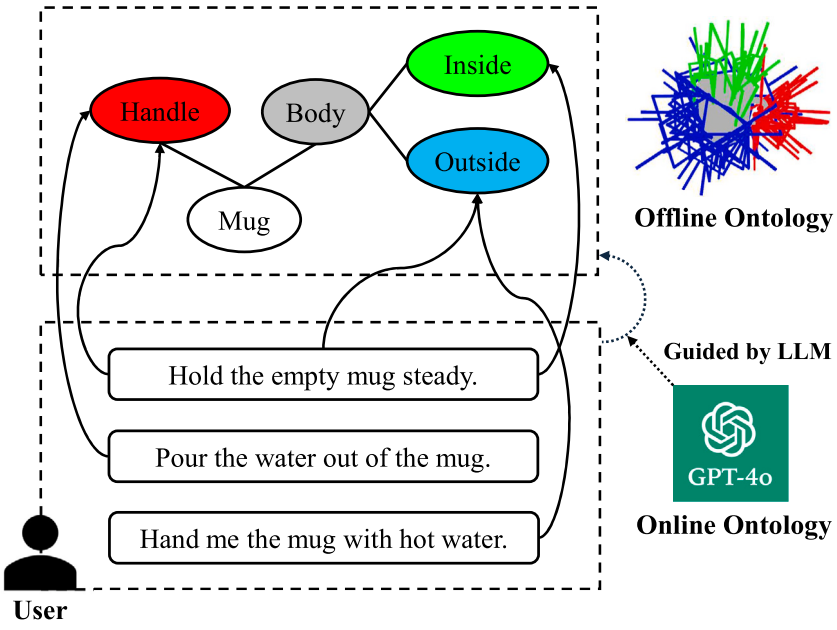


Fig. 3. LLM-guided object-part-task ontology with online and offline components.

The command is ... Step 1: Identify the type of task ... Step 2: Apply task constraints ... Analyzing the object parts ... Best choice for the robot ... ***Conclusion: The robot should grasp ...***

The functional part name can be extracted from the **Conclusion** line. For further evaluation, we apply this structured prompt to the natural language annotations provided in the DROID dataset [42]. We select 20 task instructions that are suitable for TOG scenarios, such as *"Pour the contents of the orange cup into the pot"* and *"Use the sharpie to draw on the board"*. For each target object in each task, we manually define a corresponding offline ontology and assign ground-truth labels for functional part names. To assess repeatability, each instruction is tested 3 times using different agents. Across a total of 60 trials, the LLM responses achieve 100% accuracy, demonstrating the reliability and stability of the proposed ontological reasoning approach.

3.4. Functional part recognition

To map the semantic identification of functional object parts into geometric space for subsequent planning, we need to extract part-level features from visual inputs. A common approach is to train a vision model for object part segmentation. However, we observe that even state-of-the-art segmentation models [43] suffer from viewpoint sensitivity, where segmentation results become inconsistent under varying viewpoints. As shown in Fig. 4, components such as the *mug handle* and the *bottle cap* are not reliably identified when the objects are placed in

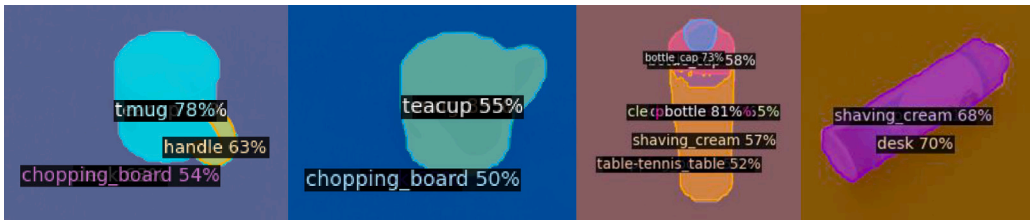


Fig. 4. Sensitivity of model-based part segmentation to viewpoint changes (colors in the figure are randomly assigned during segmentation and do not indicate part identity). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

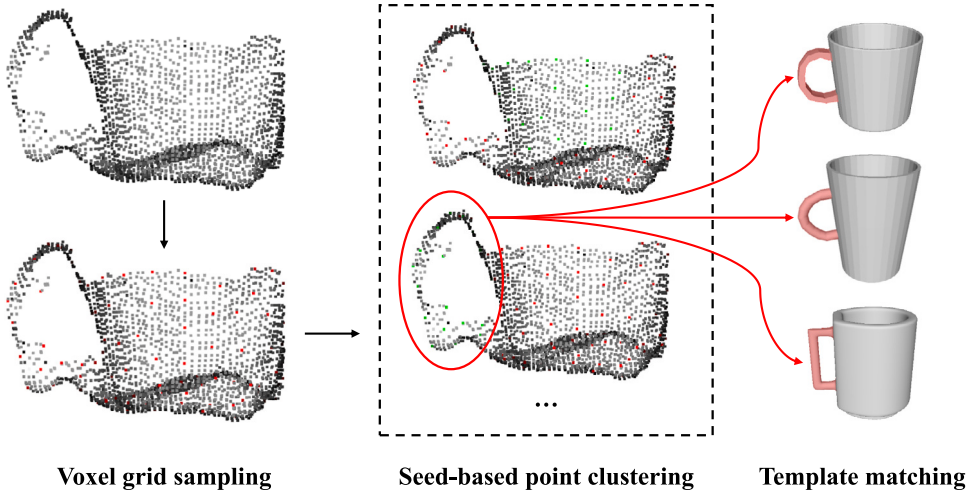


Fig. 5. Template-assisted object part recognition using a three-step strategy.

different poses. To address this limitation, we propose a new strategy based on template-assisted geometric analysis.

Given the RGB input containing the target object, we first apply a powerful class-agnostic segmentation method that is robust to viewpoint variations, SAM [44], to obtain an object mask indicating the object's planar location. Using the corresponding depth input, we perform pixel-to-point projection to map the 2D mask into a 3D point cloud, capturing the object's spatial geometry. However, due to partial observation, the resulting object point cloud often contains large unseen areas, making it difficult to perform reliable part segmentation with incomplete geometric features. To address this, we incorporate pre-segmented model templates associated with the predefined ontology and use them as references to guide part segmentation. This template-guided process consists of three main steps: sampling, clustering and matching, as illustrated in Fig. 5. We begin by converting all model templates from mesh representations to point clouds using a voxel grid filter with an appropriate leaf size (5 mm in our task). Then, in the sampling step, we apply the same voxel filter to downsample the observed object point cloud, ensuring uniform point density. Next, we take each sampled point as a seed and search for its k-nearest neighbors to form a local point cluster. The number of neighbors k is determined using the following equation:

$$k = \frac{N(o_{all})N(m_{part})}{N(m_{all})} \tag{1}$$

where $N(\cdot)$ denotes the number of points in a point cloud. o_{all} and m_{all} represent the entire point clouds of the observed object and the model template, respectively. m_{part} corresponds to the point cloud of the functional part within the template. Finally, the sampled point clusters are matched against the pre-segmented model templates to identify the best-matching cluster, the one that most closely resembles the target functional part.

However, the matching process is non-trivial. As illustrated in Fig. 5, point clouds captured by consumer-grade cameras often suffer from low precision due to sensing noise. To ensure the accuracy, efficiency, and stability of the matching process under such conditions, we introduce a multi-metric similarity evaluation method that considers both local and global geometric similarity. Letting a sampled point cluster be denoted as o_{part}, we first apply Principal Component Analysis (PCA) to evaluate the similarity in point distribution between o_{part} and m_{part}, using the following metric:

$$d_{pca} = \left\| \frac{\sigma_o}{|\sigma_o|} - \frac{\sigma_m}{|\sigma_m|} \right\|_2 \tag{2}$$

where d_{pca} represents the PCA-based distributional difference between o_{part} and m_{part}. σ_o and σ_m denote the 3-dimensional PCA singular value vectors of o_{part} and m_{part}, respectively, normalized by their Euclidean norms. This evaluation allows for a quantitative comparison between point distributions. However, PCA alone is insufficient to fully capture shape similarity. An intuitive example is that a cube and a sphere may yield similar PCA results due to geometric symmetry, despite having fundamentally different shapes.

To address this issue, we introduce a second evaluation metric: Point-to-Point Distance (PPD). As illustrated on the left side of Fig. 6, we use the center point (for m_{part}) and the seed point (for o_{part}) to calculate the PPD. The center point is identified by first computing the center of the bounding box of m_{part}, and then finding its nearest neighbor within m_{part}. Based on this point, we calculate the standard deviation of its distances to all other points in m_{part}, denoted as $D(s_m)$. For the seed point sampled in o_{part}, we similarly compute the standard deviation of its distances to all other points in the local cluster, denoted as $D(s_o)$. The PPD is then evaluated as:

$$d_{ppd} = \left| \frac{D(s_o)}{\max(|s_o - \overline{s_o}|)} - \frac{D(s_m)}{\max(|s_m - \overline{s_m}|)} \right| \tag{3}$$

where $\max(\cdot)$ denotes the maximum deviation from the mean occurring during the point distance calculation. The resulting value, d_{ppd}, reflects

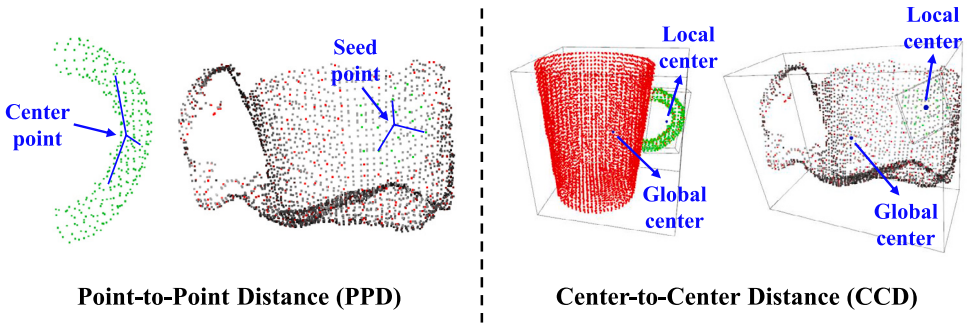


Fig. 6. Two distance metrics for evaluating similarity between point clusters.

the difference in point dispersion between m_{part} and o_{part}, offering an additional cue for distinguishing functional parts with different spatial shapes. However, this metric is non-directional and does not capture the full 3D geometric structure, making it unsuitable for use in isolation.

While the combination of d_{pca} and d_{ppd} performs well for evaluating similarity between point clusters, they focus primarily on local geometry and can become less reliable when visual features are sparse or noisy. To address this limitation, we incorporate a third metric: Center-to-Center Distance (CCD). As illustrated on the right side of Fig. 6, we obtain the bounding box centers of both the entire point clouds m_{all}, o_{all} and the part point clouds m_{part}, o_{part}, denoted as global center and local center, respectively. The CCD is then evaluated as:

$$d_{ccd} = \left| \frac{c_o}{b_o} - \frac{c_m}{b_m} \right| \tag{4}$$

where c_o and c_m represent the distances between the local and global centers within the observed object and the model template, respectively. b_o and b_m denote the maximum possible distances, defined as half the diagonal length of the bounding boxes of o_{all} and m_{all}, respectively. The resulting value, d_{ccd}, reflects the difference in the relative global positions of m_{part} and o_{part} within their respective entire objects, thereby enhancing functional part recognition especially when local visual features are sparse or noisy.

Combining the three metrics, we define the final evaluation function as:

$$d = d_{pca} + d_{ppd} + d_{ccd} \tag{5}$$

where d represents the overall difference between a template functional part and a sampled point cluster within the observed object. Smaller values of d indicate higher similarity. Note that in Eqs. (2)–(4), all metrics have been normalized using appropriate boundary values, allowing them to be combined directly through addition. The final output of the matching process is the point cluster o^*_{part} that achieves the lowest average d across matches with m_{part} from all available templates.

3.5. Task-oriented grasp planning

After recognizing the functional part represented by o^*_{part}, the next step is to plan robust grasps within its region to achieve the goal of TOG. However, due to the sparse and noisy point features in o^*_{part}, directly generating reliable grasps from the point cloud is extremely challenging. To overcome this, we leverage the matched model template again to transfer grasping knowledge from known objects to unknown targets. For each model template, we use a mesh-segmentation-based approach [38] to preplan over 50 grasps on each subdivided part based on the offline ontology. For a given task, only the preplanned grasps associated with the relevant functional part are utilized. To transfer this grasping knowledge from the template to the object, we first need to obtain the transformation matrix between the point clouds o_{all} and m_{all} to determine their relative pose. However, the incomplete point cloud from partial observation poses a challenge for accurate alignment with the complete template point cloud using traditional registration methods like ICP [45]. To address this, we propose a novel strategy called local-to-global registration, which aligns the point clouds step by step, from the functional part to the entire object, thereby enhancing alignment accuracy.

As illustrated in Fig. 7, we begin by matching o^*_{part} with m_{part} using a combined registration algorithm, RANSAC + ICP, implemented in Open3D.[1] This process is repeated iteratively until a sufficient number of point correspondences (over half of the points in o^*_{part}) are identified, indicating that the two point clouds are well-aligned. Next, we apply the resulting transformation matrix, denoted as T_{loc}, to initially align o_{all} with m_{all}. At this stage, although the functional parts are aligned, the entire objects may still show some misalignment due to rotational errors in the local registration. Directly applying ICP at this point often fails to improve the result. To overcome this, we introduce an optimization method that rotates o_{all} around the seed point of o^*_{part} to find an optimal rotation, denoted as T_{opt}, which minimizes the distance between o_{all} and m_{all}. The rotation space is discretized by sampling angles from $-180°$ to $180°$ in $45°$ increments along each of the three axes, resulting in $8^3 = 512$ total rotation attempts. This optimization process does not take long with parallel computing. Once the optimal rotation is identified, we apply the ICP algorithm to further refine the alignment between o_{all} and m_{all}, ensuring precise registration. Assuming the resulting transformation matrix from ICP is T_{icp}, the final transformation from o_{all} to m_{all} is then calculated as $T = T_{loc}T_{opt}T_{icp}$.

While this local-to-global registration is performed between the target object and all relevant model templates, we use only the best-matching template, which achieves the highest fitness score in the final ICP process, as the reference for grasp planning. Assuming the set of preplanned grasps on the best-matching template is G_m, and the camera pose relative to the robot base (the origin of the world coordinate system) is T_0, we can generate a corresponding set of imitative grasps on the target object as $G_o = T_0 T^{-1} G_m$. To ensure executable grasps for real-world tasks, we perform IK computation and collision detection within a simulation environment,[2] filtering out infeasible grasps that are either unreachable or result in collisions. For selecting reliable grasps positioned on the functional part, we prioritize those located within the region of o^*_{part} by generating a cube collision model M_{cube} within the gripper closure area and a point cloud collision model M_{pcd} corresponding to o^*_{part}. When $M_{cube} \cap M_{pcd} \neq \emptyset$, the grasp is considered correctly positioned and selected for execution.

However, we observe that the transferred imitative grasps are not always highly stable due to inherent differences between the model template and the target object. As shown in Fig. 8, when the two point clouds align perfectly, the generated grasps are typically executable without failure. However, when small deviations occur, misalignment between the grasp center and the target object part may introduce

[1] http://open3d.org/docs/release/tutorial/pipelines
[2] https://github.com/wanweiwei07/wrs

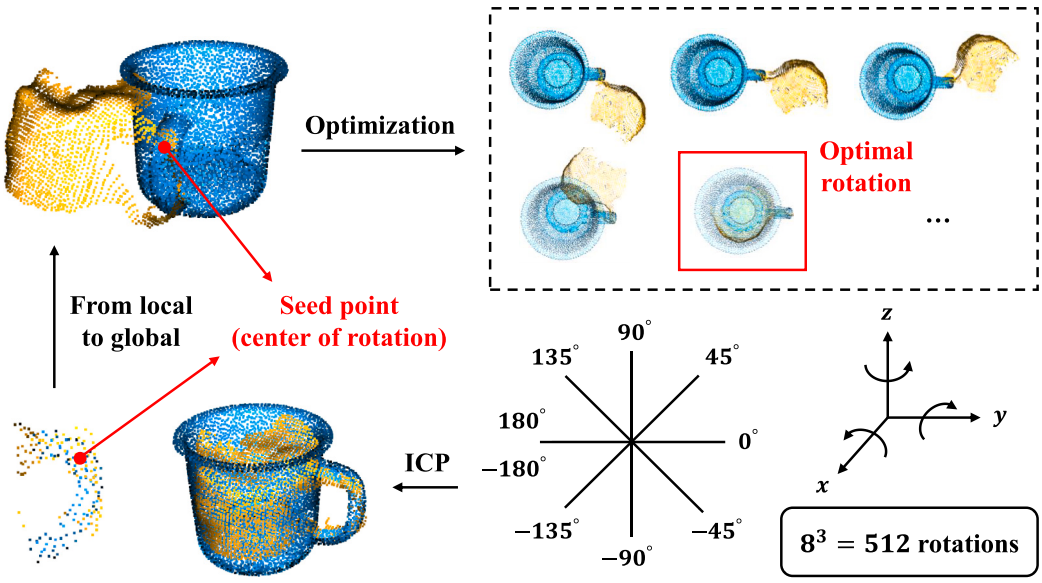


Fig. 7. Local-to-global point cloud registration through rotation optimization.

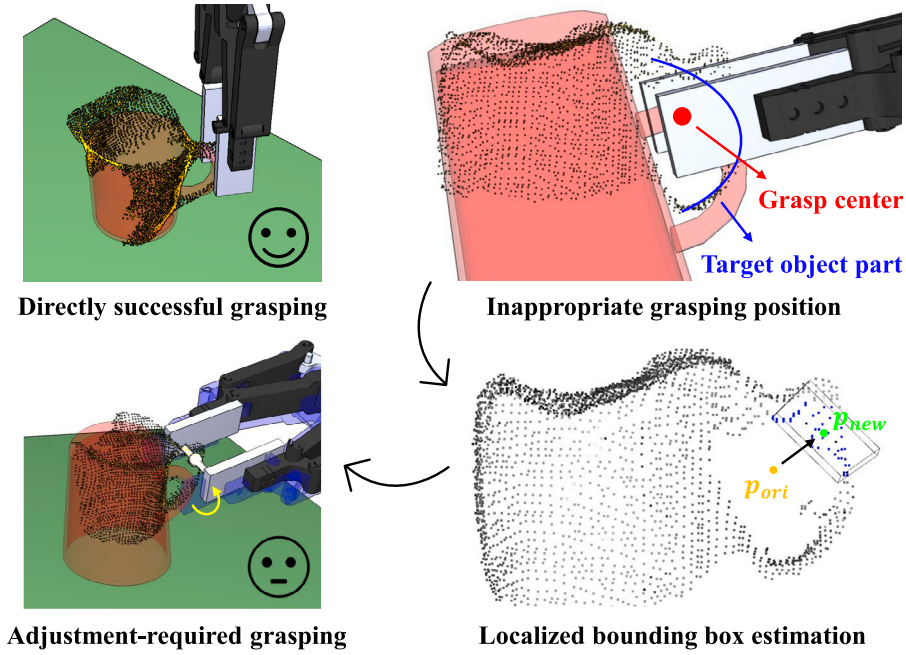


Fig. 8. Adjustment of suboptimal grasps via localized bounding box estimation.

potential risks, such as: (1) Collision between the grasp pose and other parts of the object; (2) Suboptimal contact between the gripper and the object, which may reduce grasp stability. To detect such conditions, we generate a stick collision model M_{stick} connecting the two ends of the gripper. When $M_{stick} \cap M_{pcd} = \emptyset$, the grasp center is considered misaligned with the target object part and requires adjustment. Assuming the original grasp center is p_{ori}, we search for its nearest neighbor point in o^*_{part}, denoted as p_{neg}. We then search for the k-nearest neighbors of p_{neg} and estimate a localized bounding box (LBB) based on these points. Here, the value of k is approximately set to half the number of points in o^*_{part}. The center of the LBB is adopted as the new grasp center, denoted as p_{new}. Finally, the original grasp pose is translated along the vector $\overrightarrow{p_{ori}p_{new}}$ to let the grasp center align with the target object part, thereby improving stability.

4. Experiments

4.1. Experimental setup

To validate the effectiveness of our proposed method, we conduct the following TOG experiments: (1) a benchmark study on grasping various types of unseen objects under different human instructions; (2) an ablation study to evaluate the local-to-global registration and the stability-aware grasp adjustment; and (3) an investigation into the generalization to novel-category objects that are not part of the existing ontology. All experiments are performed in the real world, using a UR5e robot arm equipped with a Robotiq 2F-140 adaptive gripper and a wrist-mounted RealSense D435 RGB-D camera. The experimental objects are placed on a fixed platform within the robot's reachable workspace, and are observed by the wrist camera from a diagonal

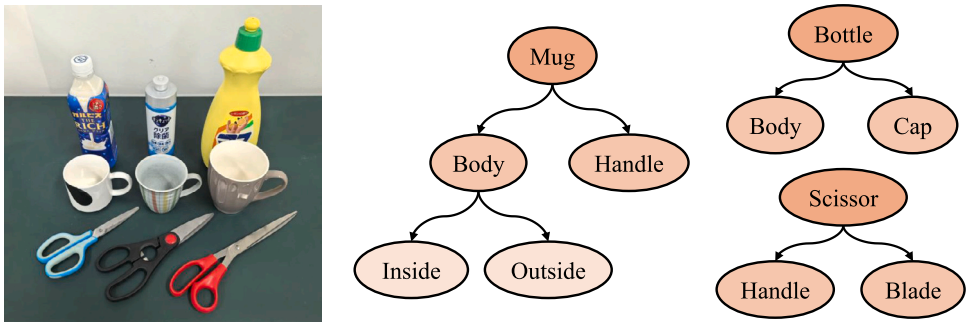


Fig. 9. Test objects used in TOG experiments with predefined ontological knowledge.

Table 1
Evaluation of PRA via a baseline study against a VLM-based approach.

Object Class	Mug	Bottle	Scissor	Average PRA
Functional part	Handle	Cap	Handle	
VLPart	12/15	11/15	6/15	29/45 (64.4%)
Our method	12/15	13/15	15/15	**40/45 (88.9%)**

downward view. All computations are carried out on a computer equipped with a Ryzen 7 5800H CPU and a GeForce RTX 3060 GPU.

4.2. Task-oriented grasping of unseen objects

For the first TOG experiment, we select three types of divisible objects with functional parts: *mug*, *bottle*, and *scissor*. Each category includes three previously unseen items with varying shapes and sizes, and their predefined ontological knowledge is stored in a database, as shown in Fig. 9. Meanwhile, we incorporate a set of model templates extracted from open-source libraries such as GrabCAD[3] into the database. Each template is pre-segmented according to the predefined ontology, and is accompanied by downsampled point clouds and preplanned grasping knowledge for each functional part.

4.2.1. Evaluation of part recognition accuracy

We begin by evaluating our geometry-based part recognition method through a comparison with a state-of-the-art VLM-based approach, VLPart [43]. We select three salient functional parts: *mug handle*, *bottle cap*, and *scissor handle*, as target regions to facilitate a clear assessment of recognition performance. Each object is placed in 5 randomized poses with their functional parts visible to the camera. For the VLPart trials, we capture RGB images and apply a pre-trained segmentation model based on Cascaded Swin Transformers, using a custom vocabulary defined by the name of the target functional part. For our method, we capture object point clouds and employ the proposed template matching approach to identify the target functional part as a point cluster. To ensure a fair evaluation, we map this point cluster to a mask region in the RGB image and represent the functional part with a minimal 2D bounding box, in the same manner as VLPart. Ground truth labels are manually assigned by fitting bounding boxes to the exact target regions. A recognition is considered accurate if the Intersection over Union (IoU) exceeds 0.5. The final part recognition accuracy (PRA) is calculated as:

$$\text{PRA} = \frac{\text{Number of correctly recognized parts}}{\text{Number of total recognition attempts}}$$

The experimental results are presented in Table 1. Our method significantly outperforms VLPart in recognizing various functional parts across different object categories. Notably, in the case of the *scissor handle*, VLPart exhibits frequent failures with the red scissor featuring an irregular handle, whereas our method maintains consistent performance. This performance gap is primarily attributed to the viewpoint sensitivity of model-based approaches, as discussed in Section 3.4. In contrast, our method mitigates this issue by leveraging 3D geometric features, which remain relatively invariant under changes in viewpoint. Fig. 11 illustrates the robustness of our method in recognizing various functional object parts from different viewpoints.

Additionally, we investigate the impact of model template quantity on part recognition performance. For each object class, we prepare 5 model templates that vary in shape and size. During evaluation, we progressively increase the number of templates used for similarity matching from a single template up to all five. As in the previous experiment, we conduct 5 trials per object under each condition and record the number of correct recognitions. As shown in Fig. 10, both the *mug cap* and *bottle cap* reach optimal performance when the number of templates increases to 3, with no further improvement observed beyond that point. This suggests that using multiple templates helps average out matching errors and enhance recognition accuracy; however, excessive template inclusion yields no additional benefit. On the other hand, object parts with distinctive geometric features, such as the *scissor handle*, demonstrate high recognition accuracy even with a minimal number of templates. Based on these findings, we standardize the use of 3 model templates per object class in our TOG framework.

4.2.2. Evaluation of grasp selection accuracy

Next, we evaluate our grasp selection method against ShapeGrasp [11], a recent task-guided approach that also utilizes LLM reasoning but employs a decomposition graph for functional part recognition, and GraspGPT [35], a recent learning-based method that relies on neural network inference. Due to GraspGPT's inability to generate real-time grasps and ShapeGrasp's restriction to top-down grasping, this experiment focuses specifically on selecting appropriate grasping positions based on human instructions, rather than on grasp stability. To achieve this, we first employ our similarity-based planning method to pre-generate a set of feasible grasp candidates densely distributed across all object parts, and then apply different baselines for grasp pose selection. It should be noted that other grasp planning approaches can serve as alternatives; however, we find our method to be the most efficient in generating part-level dense grasps. Additionally, the grasp generation process is separated from functional part reasoning and recognition, ensuring no bias in performance comparisons. For all baselines, the inputs consist of object point clouds, pre-generated grasp candidates, and a natural language instruction. The output is a single optimal grasp selected either through neural inference (GraspGPT), graph searching (ShapeGrasp), or ontological reasoning (our method). For both ShapeGrasp and our method, the selected grasp is the first found candidate located within the identified functional part region. For evaluation, we design the following task instructions as language inputs, each targeting a specific functional part of the selected objects:

(1) *Pour the water out of the mug.* (Mug → Handle)

(2) *Hold the coffee-filled mug steady.* (Mug → Body → Outside)

[3] https://grabcad.com/library

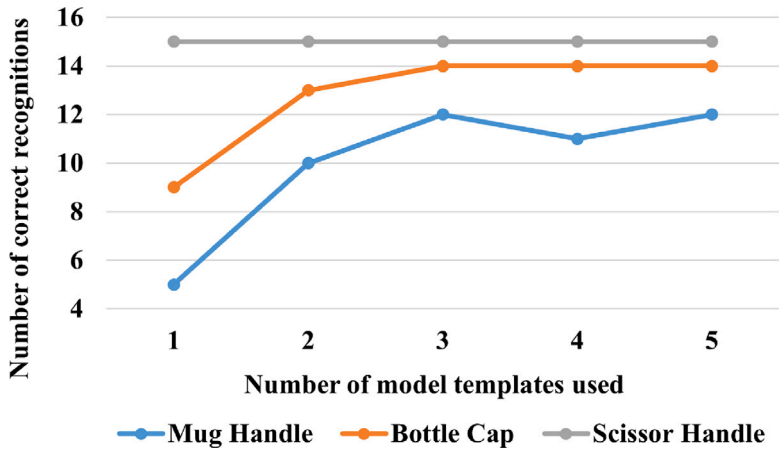


Fig. 10. Exploring the impact of template quantity on part recognition performance.

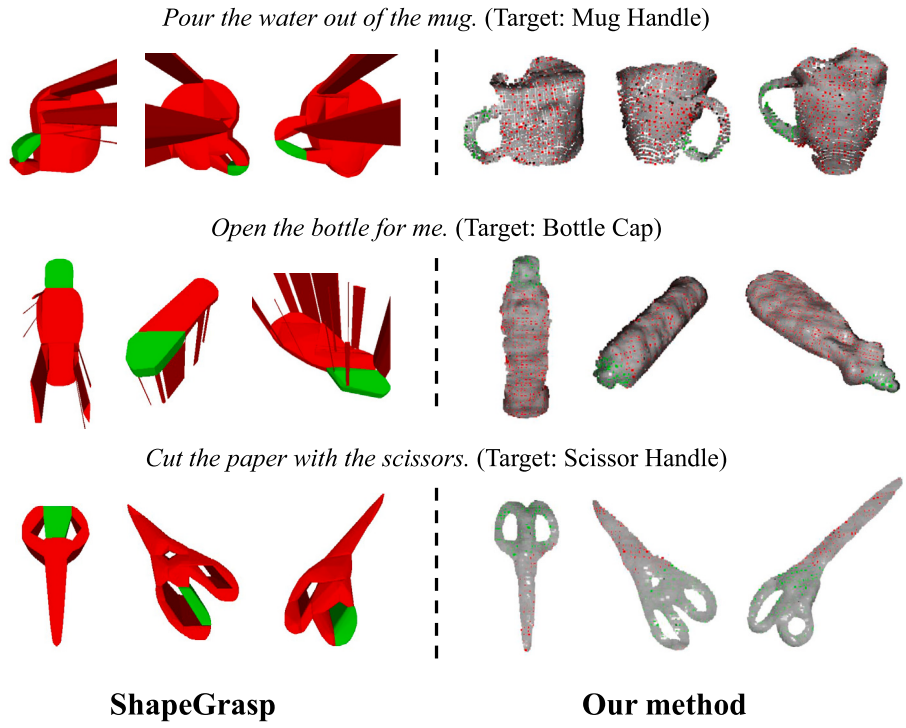


Fig. 11. Task-oriented functional part recognition: ShapeGrasp vs. Our method.

(3) *Shake the bottle before I drink it.* (Bottle → Body)
(4) *Open the bottle for me.* (Bottle → Cap)
(5) *Cut the paper with the scissors.* (Scissor → Handle)
(6) *Hand the scissors to me.* (Scissor → Blade)

Instructions 1, 3, and 5 correspond to robot manipulation tasks, while Instructions 2, 4, and 6 involve human–robot interaction. Each instruction is tested on 5 object point clouds captured from different viewpoints during the recognition experiments. A selection is considered accurate if the chosen grasp lies within the region of the target functional part. The final grasp selection accuracy (GSA) is calculated as:

$$\text{GSA} = \frac{\text{Number of correctly selected grasps}}{\text{Number of total selection attempts}}$$

The experimental results are presented in Table 2. Our method significantly outperforms both GraspGPT and ShapeGrasp in language-guided grasp selection. Notably, we observe substantial performance gaps between the methods for Instructions 4 and 6. For GraspGPT, we attribute this gap to two main reasons: (1) The *bottle* category is absent from the training dataset, and the *cap*, being a small and distinctive part of the *bottle*, is likely overlooked during grasp selection. (2) The *scissor* requires careful handling in a handover task, whereas GraspGPT does not incorporate safety constraints, such as *the robot should handle dangerous parts instead of the human*. Example failures are shown in Fig. 12. For ShapeGrasp, we observe significant deformation on the backside of the objects (the area not visible to the camera) when reconstructing 3D meshes from 2D masks, as shown in Fig. 11. This deformation introduces a risk of misalignment between the predicted grasp position and the actual part position. Additionally, the mesh decomposition in ShapeGrasp is highly sensitive to parameter variations, which sometimes leads to the recognized functional part being either smaller than the actual region (e.g., *mug handle* and *scissor handle*) or larger than the actual region (e.g., *bottle cap*). In contrast, our method requires no parameter tuning and achieves highly reliable grasp position selection by fully leveraging the interpretability of LLMs through prompt optimization and ontological reasoning. The sequential strategy – from task interpretation to part recognition, and finally

Table 2
Evaluation of GSA against two state-of-the-art task-guided approaches.

Object Class	Mug		Bottle		Scissor		Average GSA
Instruction No.	1	2	3	4	5	6	
GraspGPT	10/15	9/15	14/15	3/15	12/15	5/15	53/90 (58.9%)
ShapeGrasp	12/15	10/15	15/15	8/15	13/15	8/15	66/90 (73.3%)
Our method	11/15	14/15	15/15	11/15	15/15	13/15	**79/90 (87.8%)**

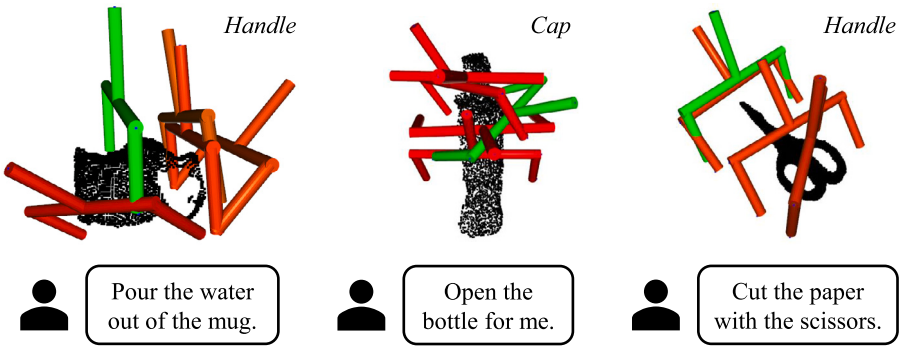


Fig. 12. Failure cases of GraspGPT in language-guided grasp selection. Selected grasps are highlighted in green, while the ground truth labels are shown in the top-right corner. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 3
Evaluation of GSR via a baseline study against grasp detection benchmarks.

Object Class	Mug		Bottle		Scissor		Average GSR
Target part	Handle	Body (Outside)	Cap	Body	Handle	Blade	
GraspNet	3/15	10/15	3/15	12/15	0/15	2/15	30/90 (33.3%)
HGGD	3/15	12/15	5/15	14/15	0/15	0/15	34/90 (37.8%)
Our method	12/15	15/15	12/15	14/15	13/15	10/15	**76/90 (84.4%)**

to grasp selection – also demonstrates better stability compared to methods that integrate all features into a single embedding space.

4.2.3. Evaluation of grasp success rate

Finally, we evaluate the reliability of part-aware grasp generation by comparing our method with GraspNet [46], a grasp detection benchmark trained on large-scale datasets, and HGGD [47], a state-of-the-art grasp planning framework that outperforms GraspNet in dynamic and challenging scenarios. For both GraspNet and HGGD, we capture RGB-D images of the scene and use their pre-trained baseline models to generate grasp candidates with associated quality scores. Since these grasp detection methods are not capable of interpreting human instructions or recognizing functional object parts, this experiment focuses solely on grasp generation performance, assuming the target object parts are predefined. Given a specific object part, we apply our template-assisted approach to identify the best-matching point cluster o^*_{part} within the observed point cloud. For all three methods, we select only grasp poses located within the region of o^*_{part} for execution. In terms of grasp selection strategy, GraspNet and HGGD rank candidates by their predicted quality scores and select the one with the highest score, while our method selects the first candidate that successfully passes the stability-aware adjustment process. A grasp attempt is considered successful if the object is grasped at the designated functional part and lifted without being dropped. Failure cases include the absence of an executable grasp, an incorrect grasp location, or the object dropping during lifting. Each functional part of each object is subjected to 5 grasp attempts, with the object placed in different positions and orientations. In cases of part recognition failure, the trial is discarded and repeated with a new object pose. The final grasp success rate (GSR) is calculated as:

$$\text{GSR} = \frac{\text{Number of successful grasps}}{\text{Number of total grasping attempts}}$$

The experimental results are presented in Table 3. Our method significantly outperforms both GraspNet and HGGD in grasping specific parts across various objects. We observe that GraspNet struggles to generate grasp candidates for small object parts, such as the *bottle cap*, due to its tendency to favor larger surface areas that are more likely to ensure grasp robustness, as shown in Fig. 13. While HGGD demonstrates higher stability when grasping mugs and bottles, it fails to generate feasible grasps on either part of thin-shaped objects such as the *scissor* in all trials, highlighting a key limitation of task-agnostic grasp detection approaches when applied to TOG problems. In contrast, our method consistently achieves high GSR across diverse object parts by leveraging similarity-based grasp planning combined with a localized positional adjustment process to optimize grasp performance.

4.3. Ablation study

In the second TOG experiment, we further verify our local-to-global point cloud registration and stability-aware grasp adjustment through two baseline comparisons: *direct registration* and *no grasp adjustment*. In the *direct registration* baseline, we employ a conventional point cloud registration pipeline using RANSAC followed by ICP to align the partial object point cloud with the complete template. In the *no grasp adjustment* baseline, we execute the first generated grasp that is both IK-solvable and collision-free without considering the position of its grasp center. To clearly illustrate differences in grasp quality, we reuse the mugs from previous experiments, fill them with water, and designate their handles as the grasping targets. Each mug undergoes five grasp planning attempts under each baseline condition. Two types of failure are considered: (1) *unsuccessful planning*, defined as either the absence of a generated grasp or a grasp targeting an incorrect object part, and (2) *unstable grasping*, identified by water spilling from the mug during

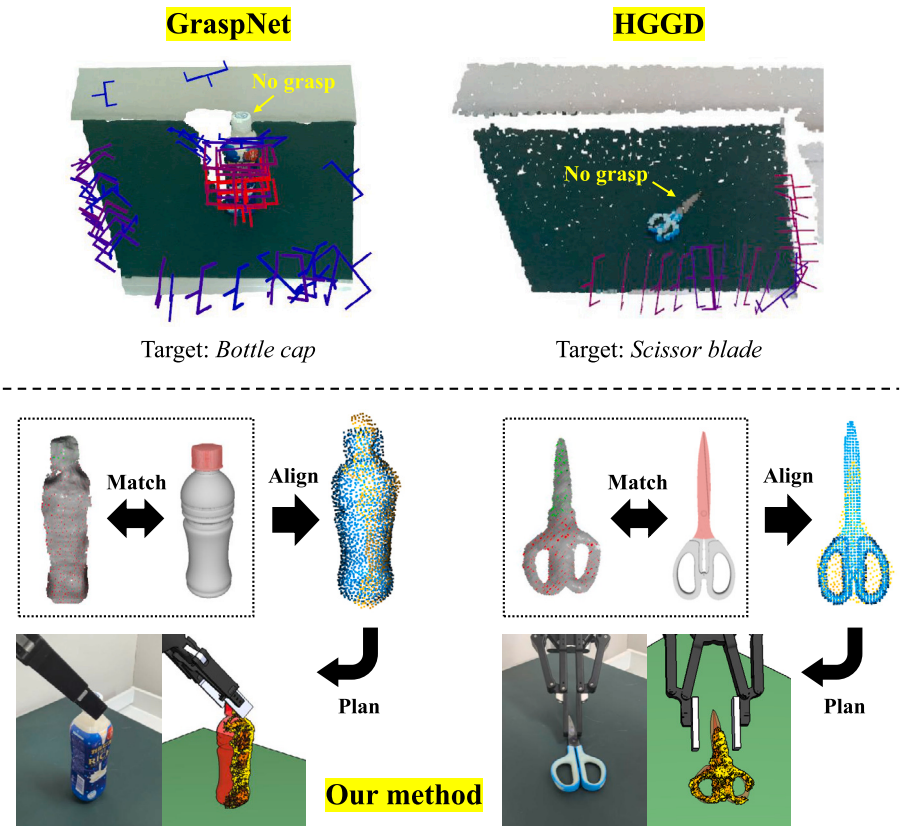


Fig. 13. Comparison of TOG performance between benchmarks and our method.

Table 4
Ablation study on proposed methods using water-filled mugs.

Indicator	PR ↑	SR ↑
Direct registration	33.3% (5/15)	60.0% (3/5)
No grasp adjustment	80.0% (12/15)	50.0% (6/12)
Full method	**86.7% (13/15)**	**76.9% (10/13)**

execution. For performance evaluation, we define the following two metrics:

$$\text{PR (Planning Rate)} = \frac{\text{Number of successfully planned grasps}}{\text{Number of total grasp planning attempts}}$$

$$\text{SR (Stabilization Rate)} = \frac{\text{Number of stable grasps}}{\text{Number of executed grasps}}$$

In the case of *unsuccessful planning*, the trial is terminated without executing any grasp and excluded from the calculation of SR. The experimental results are presented in Table 4. Our full method significantly outperforms both baseline approaches across both evaluation metrics. The notable performance gap between the *direct registration* baseline and our full method highlights the effectiveness of the proposed local-to-global point cloud registration in achieving precise matching between observed objects and model templates, which is an essential prerequisite for the reliable transfer of grasping knowledge and robust grasp generation. Additionally, the substantial improvement in SR observed between the *no grasp adjustment* baseline and our full method underscores the importance of the stability-aware grasp adjustment in enhancing the quality of the executed grasps. A visual comparison showing the superiority of our full method is provided in Fig. 14.

4.4. Robustness to occlusion, noise, and scale variation

In the third TOG experiment, we evaluate the robustness of our proposed framework under various real-world uncertainties, focusing on three unstructured conditions: (I) partial occlusion of the target object, (II) sensor-induced noise in the point cloud, and (III) scaling of model templates to varying sizes. The impacts of these conditions are shown in Fig. 15. Similar to Experiment 4.2.1, we select the *mug handle*, *bottle cap*, and *scissor handle* as representative functional parts for clear performance assessment.

For Condition I, we use a sauce bottle to block part of the target object within the camera view, resulting in a point cloud with substantial missing regions. Despite the sparsity and incompleteness of the visual input, our framework reliably identifies the functional parts, as shown in Fig. 15 (left). This resilience is largely attributed to the use of multi-metric similarity evaluation, which enables robust recognition under partial observation. However, it is worth noting that recognition failures in extreme cases – such as when the target part is fully occluded – are beyond the scope of this study.

For Condition II, we employ the camera's built-in filtering capabilities to simulate varying levels of sensor noise. Specifically, we apply the spatial filter provided by the RealSense SDK to smooth the original object point cloud. This filtering process significantly alters the point distribution, leading to noticeable variations in the object's geometry. Nevertheless, we observe that such variations have minimal impact on the accuracy of functional part recognition when employing the proposed sampling-based strategy, demonstrating the robustness of our method to sensor-induced noise.

For Condition III, we scale the model templates to both smaller and larger sizes to evaluate their matching and grasp planning performance on the same target object. Although changes in model size degrade the quality of point cloud matching (as shown in 15, right), our framework continues to generate feasible grasps on the functional parts. This

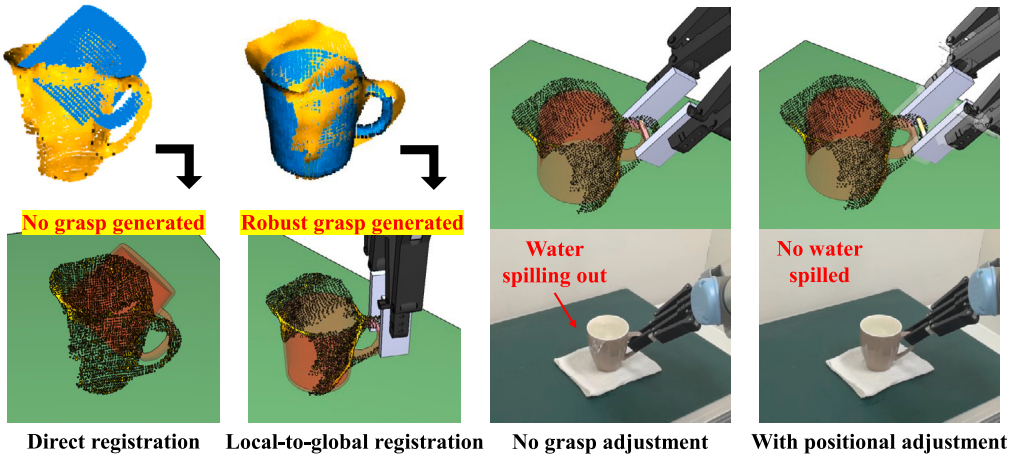


Fig. 14. Performance discrepancy between the two baselines and our full method.

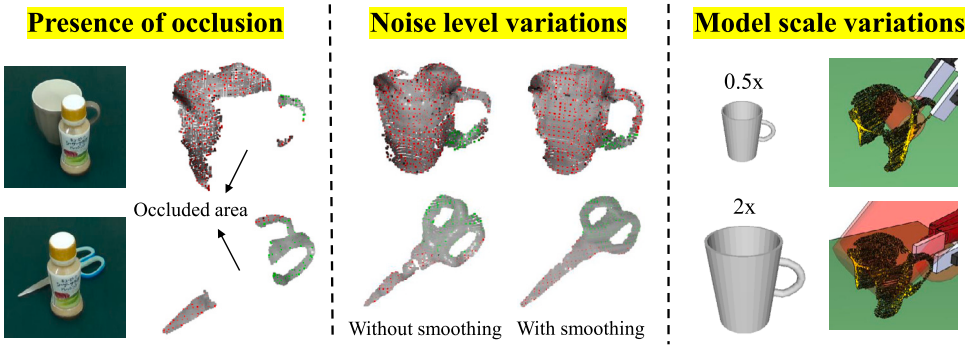


Fig. 15. Robust part recognition and grasp planning under unstructured conditions.

robustness is achieved through a combination of local-to-global registration and stability-aware positional adjustment, which ensures that grasps transferred from the model templates are consistently positioned in appropriate and stable regions.

For further quantitative evaluation, we conduct 15 trials (3 different objects × 5 grasp attempts per object) for each selected object part under each unstructured condition, and compare the performance against that under the original condition. To provide a comprehensive assessment, we introduce a new evaluation metric, PGSR (Plan and Grasp Success Rate), defined as:

$$\text{PGSR} = \frac{\text{Number of accurately planned and successfully executed grasps}}{\text{Number of total planning and grasping attempts}}$$

where only grasps planned on the accurate functional part and that stably lift the object are considered as successful. The results presented in Table 5 demonstrate the exceptional robustness of our proposed method under various real-world uncertainties. Each component of our framework – matching, registration, and adjustment – works in concert to form a reliable and effective solution for task-oriented object grasping.

However, it is important to acknowledge that the robustness of our method is limited under certain levels of interference. More challenging sensing conditions – such as intense illumination, object transparency, or highly cluttered scenes – that cause significant changes in visual features are beyond the scope of this study. Combining our approach with point cloud refinement or shape reconstruction techniques, which can recover object geometry from noisy or distorted features, may offer a potential solution for these rigorous conditions.

Table 5
Quantitative Evaluation of Performance Robustness Under Varying Conditions.

Object Class Functional part	Mug Handle	Bottle Cap	Scissor Handle	Average PGSR
Original condition	**11/15**	**12/15**	**13/15**	**36/45 (80.0%)**
With occlusion	9/15	10/15	13/15	32/45 (71.1%)
With noise	10/15	12/15	13/15	35/45 (77.8%)
With scaling	9/15	12/15	12/15	33/45 (73.3%)

4.5. Generalization to novel-category objects

Finally, in addition to delivering high performance on objects included in the ontology, our method demonstrates strong generalizability to novel-category objects, extending beyond the existing knowledge base. This is accomplished by leveraging the scalable inferential capabilities of LLMs, which can be activated by appending the following instruction to the prompt:

If the target object is not listed in the ontology, find its closest object in the ontology and use its part information.

The answer template is also modified accordingly, as described in Appendix B. Through this prompt refinement, a novel target object can be mapped to a similar known instance within the ontology, allowing the reuse of both ontological knowledge and corresponding model templates for subsequent recognition, matching, and planning. Fig. 16 illustrates two representative examples involving a *juice box* and a pair of *pliers*. Despite lacking prior knowledge of these objects, the system successfully maps them to the *bottle* and *scissor* categories in the ontology, respectively. Their subclasses and associated templates

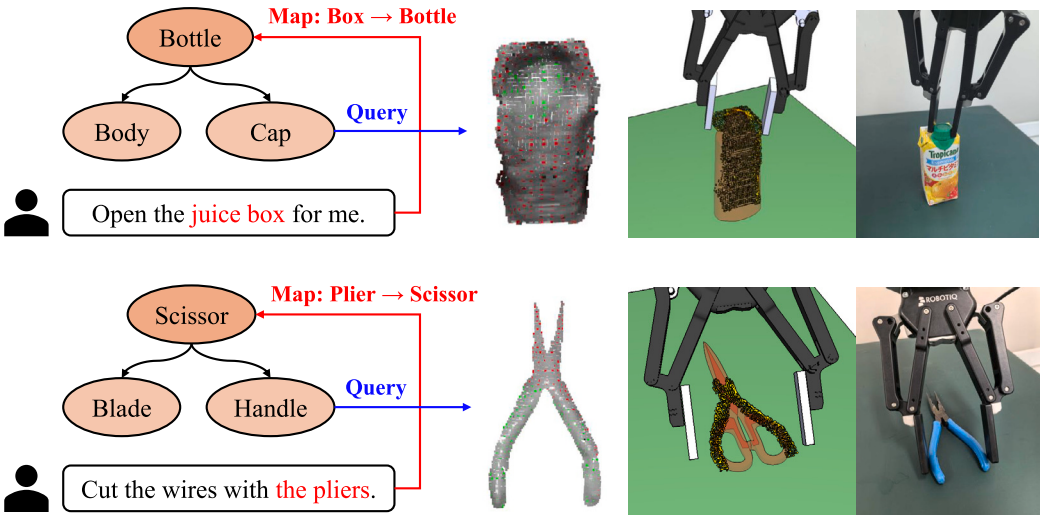


Fig. 16. Generalizing TOG to novel categories via ontological knowledge extension.

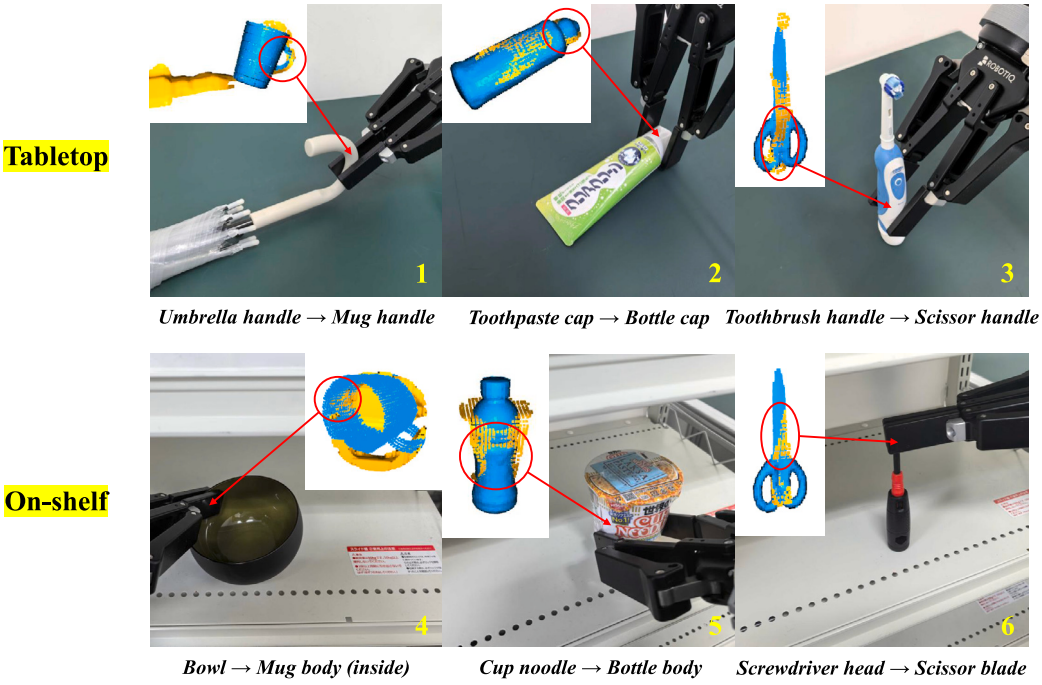


Fig. 17. Validation on diverse novel objects across tabletop and on-shelf scenarios.

are then employed to guide functional part recognition and similarity-based grasp planning. From this perspective, our approach exhibits a tree-like growth pattern: each newly introduced instance contributes to the exponential expansion of existing knowledge. Compared to conventional model-based methods that encode diverse object features into a single network, our framework offers superior reliability and robustness in handling novel objects through structural and scalable ontological reasoning.

For further quantitative evaluation, we perform TOG experiments on a diverse set of novel objects across two real-world settings: *tabletop* and *on-shelf*. As illustrated in Fig. 17, our method effectively maps unknown target functional parts to predefined object parts within the ontology, enabling the generation of task-oriented grasps for a wide variety of unseen objects that differ significantly from existing templates. Notably, the matching results shown in the top corners highlight the effectiveness of the local-to-global registration strategy in achieving accurate alignment between functional parts, even when the overall object geometry varies considerably.

Table 6 presents the overall experimental results. The object numbers correspond to those shown in Fig. 17. Each object is tested in 5 randomized positions and orientations, with the following task instructions used as inputs:

(1) *Bring me the umbrella.* (Target: umbrella handle)
(2) *Get the toothpaste ready for use.* (Target: toothpaste cap)
(3) *Pick up the toothbrush.* (Target: toothbrush handle)
(4) *Take the bowl off the shelf.* (Target: bowl)
(5) *Fetch the cup noodles for me to eat.* (Target: cup noodle)
(6) *Hand me the screwdriver so I can use it.* (Target: screwdriver head)

The results of PRA and GSR clearly demonstrate the effectiveness of our proposed method in handling a wide variety of novel objects across different operation locations. By focusing on the similarity between local functional parts rather than the global object geometry, our

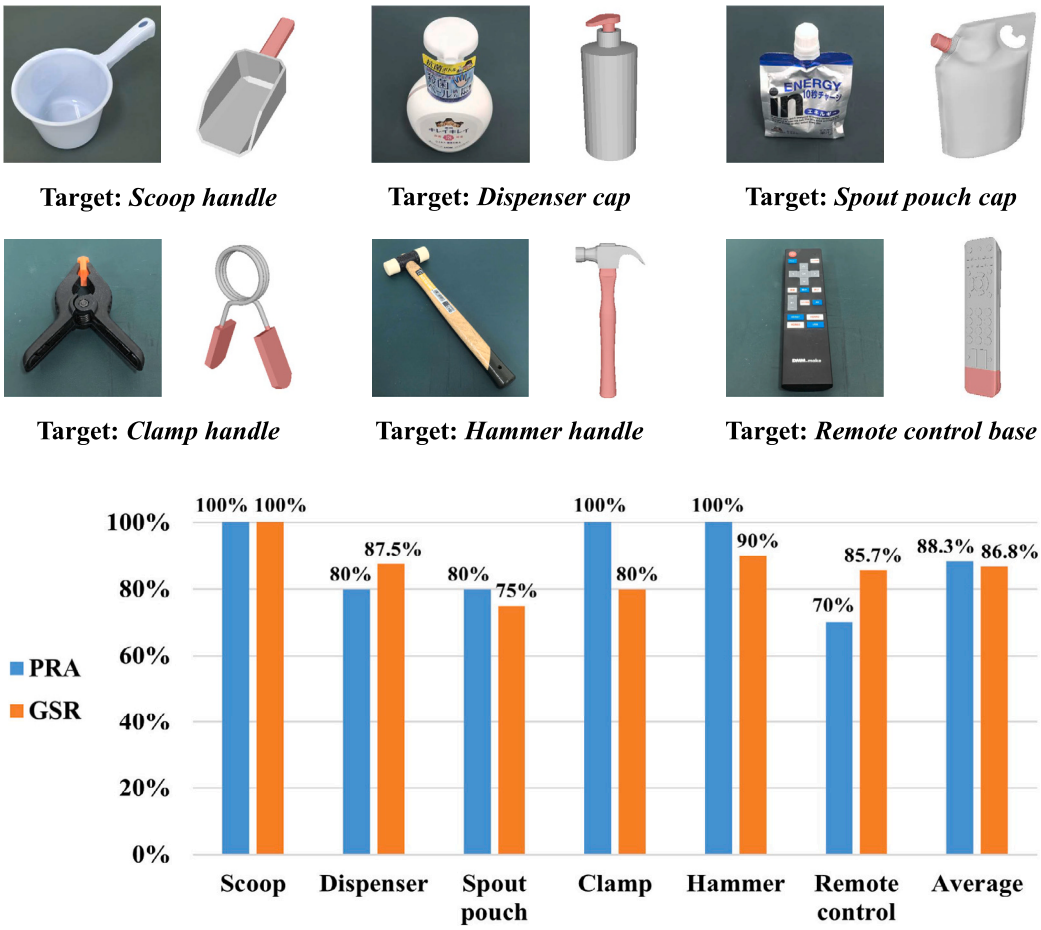


Fig. 18. Extended evaluation across a broader set of object categories and tasks.

Table 6
Quantitative evaluation of generalization performance on diverse novel objects.

Object No.	1	2	3	4	5	6	Average
PRA	4/5	5/5	4/5	5/5	5/5	4/5	27/30 (90.0%)
GSR	4/4	4/5	4/4	3/5	5/5	3/4	23/27 (85.2%)

approach achieves extensive generalization capacity despite a limited number of existing templates.

5. Discussion

5.1. Strengths and limitations

The experiments above demonstrate the superiority of our proposed method in leveraging a small number of templates to efficiently handle a wide range of novel objects. This strong generalizability primarily stems from emphasizing *local part similarity* rather than *global object similarity*. While previous studies have explored part registration for grasp transfer from demonstrated to unseen objects [48–50], their approaches typically require transforming part features into implicit representations for neural network training, followed by retrieving grasp poses from an embedding space. Such training pipelines reduce the interpretability of grasp planning outcomes – particularly in failure cases – thereby hindering model refinement and generalization. In contrast, our method directly aligns object parts using explicit geometric features through a local-to-global registration strategy. This design ensures traceability of grasp transfer errors and facilitates the optimal selection of model templates with minimal transfer error, leading to greater performance stability across diverse object categories.

To further evaluate the strengths and limitations of our proposed method, we expand the initial experimental set from three object categories to a broader set, following existing TOG benchmarks [27, 28,32]. The upper part of Fig. 18 illustrates the selected objects, their corresponding model templates, and the targeted functional parts. As in the previous experiments, we prepared three templates per object (only one is shown in the figure), all within the same category but varying in geometry. Each object was tested 10 times for both part recognition and grasp planning, using the following task instructions as inputs:

(1) *Use the scoop to get some water for me.* (Target: Scoop handle)
(2) *Press the dispenser to release soap.* (Target: Dispenser cap)
(3) *Unscrew the spout pouch so I can drink.* (Target: Spout pouch cap)
(4) *Open the clamp to secure the books.* (Target: Clamp handle)
(5) *Pass me the hammer.* (Target: Hammer handle)
(6) *Hold the remote control steady.* (Target: Remote control base)

The results in the lower part of Fig. 18 demonstrate the consistent superiority of our method across diverse object categories. Exploiting the local similarity of functional parts provides sufficient accuracy and stability when matching unseen target objects to semantically similar templates, even when their global geometries differ. However, failures occur when the target part is small with sparse features (e.g., the spout pouch cap) or has a geometrically similar counterpart on the symmetrical side (e.g., the remote control base). In such cases, the part's geometric features are less distinctive relative to the entire object, and similar regions on the surface can significantly interfere with

part recognition. This represents a potential limitation of the current approach when dealing with specific object types.

5.2. Guidelines for database construction

In addition to the registration strategy, constructing an appropriate template database is a critical step in achieving optimal performance. While our ultimate goal is to generalize the method to handle open-ended objects and tasks, several key considerations must be addressed: (I) What types of templates should be included in the database? (II) How many instances should be collected for each object category? (III) What intra-category variation should be considered during data collection? Based on our experimental results, we summarize the following principles for selecting model templates:

(I) In the context of robotic grasping, typical functional parts can be categorized as *Graspable Parts* (e.g., handle, body), *Active Parts* (e.g., blade, head), *Support Parts* (e.g., bottom, base), and *Container Parts* (e.g., cap, lid). For each part type, one representative object category can be included in the database and generalized to other categories with similar functional parts. For example, incorporating a *mug* template with the functional part *handle* enables the method to generalize to *pots*, *kettles*, and *umbrellas* that share similar handles. However, a single category may not fully represent all variations—for instance, the *scissor handle* and *hammer handle* differ significantly from a *mug handle* in geometry. Therefore, template selection should consider high-level category associations (e.g., *handle* appearing in both *containers* and *tools*) to ensure comprehensive coverage of object types.

(II) Regarding the number of templates per object category, our TOG experiments (Fig. 10) show that three templates with different shapes and sizes are generally sufficient for good performance. Nevertheless, the optimal number should be adaptively determined based on category-specific requirements, which calls for a systematic derivation of the relationship between template quantity and object characteristics. We leave this derivation to future work for a more precise determination of category-aware template quantities.

(III) Within each object category, the selected templates should exhibit diversity in shape, size, and texture to capture a broad range of real-world instances. To minimize database redundancy, templates with overly similar properties should be avoided. While our method demonstrates robustness to object variations, real-world tests confirm that a more diverse template set increases the likelihood of finding relevant references, typically resulting in higher part-matching accuracy and improved grasp stability.

Based on these principles, a compact yet highly effective template database capable of supporting a wide range of objects and tasks can be constructed.

5.3. Accuracy-runtime trade-off

We further investigate the precise impact of a growing template database on TOG performance and computational cost, aiming to provide a valid reference for real-world deployment. To this end, we reuse the objects shown in Fig. 18 and perform the same trials while varying the number of templates used for each object. The results of PRA, GSR, and runtime for the matching process are recorded in Table 7. Consistent with the results in Fig. 10, both recognition and grasp accuracy improve as the number of templates increases from 1 to 3, but remain virtually unchanged thereafter. Although the best performance is observed with 10 ∼ 50 templates, the significant increase in runtime renders the marginal performance gains negligible. In conclusion, the number of templates per object category should be set around 5 in the initial stage, with further adjustments made based on intra-category variations.

Table 7
Quantitative trade-off analysis: Recognition, grasping, and computation.

Number of Templates	1	3	5	10	50
PRA ↑	65.0%	88.3%	90.0%	90.0%	**91.7%**
GSR ↑	61.5%	86.8%	87.0%	**88.9%**	87.3%
Runtime ↓	**0.35s**	0.63s	0.89s	1.49s	6.72s

6. Conclusion

In this paper, we present a novel strategy for task-oriented grasping of unseen objects. Our approach introduces an object-part-task ontology consisting of online and offline components, associated through guidance from LLMs. By leveraging optimized user prompts, the LLM accurately interprets intuitive human instructions and infers corresponding functional object parts based on ontological knowledge. Additionally, we propose a novel part recognition method that utilizes pre-existing model templates to identify target object parts as point clusters. Based on this representation, we employ a local-to-global point cloud registration framework, followed by a stability-aware grasp adjustment process, to transfer grasping knowledge from the best-matching model template to the unseen target object, enabling robust task-oriented grasp generation. The stable and efficient integration of task reasoning (language), object-part recognition (vision), and grasp detection (action) constitutes the main novelty of our work. To validate this, we first conduct real-world experiments on similar-category objects, demonstrating its superiority over state-of-the-art approaches. Furthermore, we highlight its scalability to a wide range of novel-category objects using a target-agnostic, pre-generated database composed of only a small set of model templates.

Despite its strengths, the proposed method has several limitations: (1) it cannot handle completely novel objects that lack semantically or geometrically similar references in the ontology; (2) its matching accuracy degrades when the viewpoint provides insufficient features of the target part or when the part is not geometrically distinctive; and (3) it does not currently account for constraints imposed by subsequent manipulation tasks beyond grasp execution. Addressing these limitations will be the focus of our future work.

CRediT authorship contribution statement

Hao Chen: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Takuya Kiyokawa:** Writing – review & editing, Supervision, Resources, Methodology. **Weiwei Wan:** Supervision, Software, Resources. **Kensuke Harada:** Writing – review & editing, Supervision, Project administration, Methodology, Funding acquisition.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author(s) used ChatGPT in order to improve the readability and language of the manuscript. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the published article.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Hao Chen reports financial support was provided by New Energy and Industrial Technology Development Organization (NEDO). If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This research is subsidized by New Energy and Industrial Technology Development Organization (NEDO) under a project JPNP20016. This paper is one of the achievements of joint research with and is jointly owned copyrighted material of ROBOT Industrial Basic Technology Collaborative Innovation Partnership (ROBOCIP).

Appendix A. LLM prompt optimization

To optimize LLM prompts for accurate instruction interpretation, we adopt the framework proposed in [41], which consists of three components: (1) *Executor*, the LLM that processes the given prompt and generates a corresponding response; (2) *Evaluator*, a human supervisor who assesses the LLM's output and determines the next step—either terminating the optimization if the result is satisfactory, or providing feedback for improvement if not; and (3) *Improver*, where the LLM is reused to refine the prompt based on human feedback. The improved prompt is then fed back into the *Evaluator*, initiating a new iteration of the process (see Fig. A.19). The key advantage of this framework is that humans only need to act as advisors, offering suggestions rather than revising the prompts themselves. The LLM (essentially a more efficient "engineer"), through repeated iterations and human supervision, ultimately generates an optimal prompt. Below, we illustrate an example of prompt optimization for instructing a TOG task:

Create Agent 1

User: *Given the following ontology:*

Mug → Handle, Mug → Body → Inside, Mug → Body → Outside

Scissor → Handle, Scissor → Blade

Considering these two commands given to a robot:

1. "Hand me the mug with hot water in it".

2. "Pick up the scissor and pass it to me.

Question: Which part(s) of each object should the robot grasp? Please only answer the part names.

Agent: *1. Mug → Handle 2. Scissor → Handle*

(Ground truth) 1. Mug → Body → Outside 2. Scissor → Blade

User: *Both the answers are wrong because in a human–robot interaction task, we want the robot to handle dangerous parts instead of the human. Meanwhile, the robot and human should take different parts to ensure sufficient operating space. How can we improve the prompt to obtain correct answers?*

Agent: *You can improve the question by explicitly stating the criteria for grasping ... (Unimportant statements)*

Revised Prompt: "Given the following ontology:

Mug → Handle, Mug → Body → Inside, Mug → Body → Outside

Scissor → Handle, Scissor → Blade

Consider these two commands given to a robot:

1. Hand me the mug with hot water in it

2. Pick up the scissor and pass it to me

Which part(s) of the objects should the robot grasp, ensuring that (a) the robot handles dangerous parts instead of the human, and (b) the robot and human grasp different parts for sufficient operating space?"

This revision ensures that the intended grasping strategy is clear, leading to correct answers ... (Unimportant statements)

Create Agent 2

User: *(The revised prompt from Agent 1)*

Agent: *1. Mug → Handle 2. Scissor → Blade*

(Ground truth) 1. Mug → Body → Outside 2. Scissor → Blade

User: *The first answer is still wrong because in a human–robot interaction task, we want the robot to handle dangerous parts instead of the human. Meanwhile, the robot and human should take different parts to ensure sufficient operating space. How can we improve the prompt to obtain correct answers?*

Agent: *To ensure correct answers, we should explicitly state the constraints more clearly ... (Unimportant statements)*

Revised Prompt: "Given the following ontology:

Mug → Handle, Mug → Body → Inside, Mug → Body → Outside

Scissor → Handle, Scissor → Blade

A robot is given the following commands:

1. Hand me the mug with hot water in it

2. Pick up the scissor and pass it to me

Constraints:

1. The robot should grasp parts that are either difficult to manipulate or potentially dangerous, while the human should grasp the safer, easier-to-handle parts.

2. The robot and human should each grasp a different part of the object to ensure enough operating space.

Question: Which part(s) of each object should the robot grasp? Answer with only the part names".

This version explicitly defines the constraints, making it clear that ... (Unimportant statements)

Create Agent 3

User: *(The revised prompt from Agent 2)*

Agent: *1. Mug → Body → Outside 2. Scissor → Blade*

(Ground truth) 1. Mug → Body → Outside 2. Scissor → Blade

Prompt optimization completed

Appendix B. Modified answer template

To fully leverage the scalability of LLMs for novel-category objects, we design the following step-by-step reasoning process as the answer template:

The given command is "Empty the bowl into the sink".

Step 1: Identify the Type of Task

The robot is instructed to pick up the bowl and pour its contents into the sink.

There is no human involved in handling the bowl.

Therefore, this is a manipulation task.

Step 2: Find the Closest Object in the Ontology

The bowl is not in the ontology. Let's look for the most structurally similar object:

The bowl, like the mug, is a container with a concave inside and round outer body.

The key difference: a mug typically has a handle, whereas a bowl usually does not.

Despite this, the bowl's structure (body with inside and outside) closely aligns with the mug's body parts.

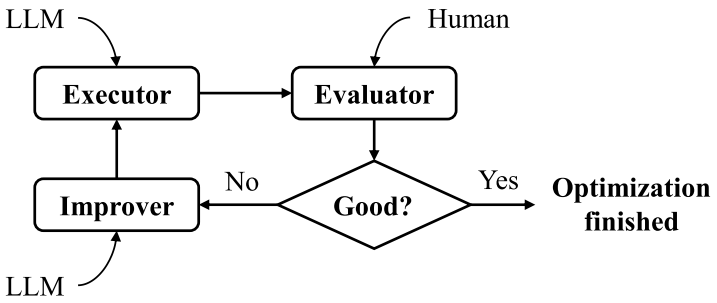


Fig. A.19. Framework for reusing LLMs in iterative prompt optimization.

So, we map: Bowl ≈ Mug (without handle)

Relevant parts:

Body → inside

Body → outside

Step 3: Apply Task Constraints

The robot must grasp the bowl in a way that allows it to tilt and pour the contents into the sink.

This requires a secure and controlled grip—likely from the outside of the body to allow for rotation.

Analyzing the Object Parts

Body (inside): Contains the contents—not suitable for gripping.

Body (outside): Provides surface area for a stable grip and control when tilting.

Best Choice for the Robot

The robot should grasp the body (outside) of the bowl to securely hold, tilt, and empty its contents into the sink.

***Conclusion**: The Robot Should Grasp the Body (Outside)*

This enables the robot to effectively manipulate the bowl for pouring while maintaining control and avoiding contact with the contents.

Data availability

The authors do not have permission to share data.

References

[1] S. Deng, X. Xu, C. Wu, K. Chen, K. Jia, 3D AffordanceNet: A benchmark for visual object affordance understanding, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2021.

[2] Y. Li, N. Zhao, J. Xiao, C. Feng, X. Wang, T.-s. Chua, LASO: Language-guided affordance segmentation on 3D object, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2024, pp. 14251–14260.

[3] H. Dang, P.K. Allen, Semantic grasping: Planning robotic grasps functionally suitable for an object manipulation task, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2012, pp. 1311–1317.

[4] D. Song, K. Huebner, V. Kyrki, D. Kragic, Learning task constraints for robot grasping using graphical models, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2010, pp. 1579–1585.

[5] R. Detry, J. Papon, L. Matthies, Task-oriented grasping with semantic and geometric scene understanding, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2017, pp. 3266–3273.

[6] F.-J. Chu, R. Xu, P.A. Vela, Learning affordance segmentation for real-world robotic manipulation via synthetic images, IEEE Robot. Autom. Lett. 4 (2) (2019) 1140–1147.

[7] A. Murali, W. Liu, K. Marino, S. Chernova, A. Gupta, Same object, different grasps: Data and semantic knowledge for task-oriented grasping, in: The Conference on Robot Learning, CoRL, 2020.

[8] C. Tang, D. Huang, L. Meng, W. Liu, H. Zhang, Task-oriented grasp prediction with visual-language inputs, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2023, pp. 4881–4888.

[9] Y.-L. Wei, J.-J. Jiang, C. Xing, X.-T. Tan, X.-M. Wu, H. Li, M. Cutkosky, W.-S. Zheng, Grasp as you say: Language-guided dexterous grasp generation, NeurIPS, in: Advances in Neural Information Processing Systems, vol. 37, 2024, pp. 46881–46907.

[10] J. Zhang, W. Xu, Z. Yu, P. Xie, T. Tang, C. Lu, DexTOG: Learning task-oriented dexterous grasp with language condition, IEEE Robot. Autom. Lett. 10 (2) (2025) 995–1002.

[11] S. Li, S. Bhagat, J. Campbell, Y. Xie, W. Kim, K. Sycara, S. Stepputtis, ShapeGrasp: Zero-shot task-oriented grasping with large language models through geometric decomposition, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2024, pp. 10527–10534.

[12] H. Chen, T. Kiyokawa, W. Wan, K. Harada, Category-association based similarity matching for novel object pick-and-place task, IEEE Robot. Autom. Lett. 7 (2) (2022) 2961–2968.

[13] A. ten Pas, M. Gualtieri, K. Saenko, R. Platt, Grasp pose detection in point clouds, Int. J. Robot. Res. 36 (13–14) (2017) 1455–1473.

[14] J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, K. Goldberg, Learning ambidextrous robot grasping policies, Sci. Robot. 4 (26) (2019) eaau4984.

[15] M. Breyer, J.J. Chung, L. Ott, S. Roland, N. Juan, Volumetric grasping network: Real-time 6 DOF grasp detection in clutter, in: The Conference on Robot Learning, CoRL, 2020.

[16] Z. Jiang, Y. Zhu, M. Svetlik, K. Fang, Y. Zhu, Synergies between affordance and geometry: 6-DoF grasp detection via implicit representations, in: Robotics: Science and System, RSS, 2021.

[17] H.-S. Fang, C. Wang, H. Fang, M. Gou, J. Liu, H. Yan, W. Liu, Y. Xie, C. Lu, AnyGrasp: Robust and efficient grasp perception in spatial and temporal domains, IEEE Trans. Robot. 39 (5) (2023) 3929–3945.

[18] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, S. Levine, QT-opt: Scalable deep reinforcement learning for vision-based robotic manipulation, in: The Conference on Robot Learning, CoRL, 2018.

[19] P. Mandikal, K. Grauman, Learning dexterous grasping with object-centric visual affordances, in: IEEE International Conference on Robotics and Automation, ICRA, 2021, pp. 6169–6176.

[20] L. Wang, Y. Xiang, W. Yang, A. Mousavian, D. Fox, Goal-auxiliary actor-critic for 6D robotic grasping with point clouds, in: The Conference on Robot Learning, CoRL, 2021.

[21] M. Adjigble, N. Marturi, V. Ortenzi, V. Rajasekaran, P. Corke, R. Stolkin, Model-free and learning-free grasping by local contact moment matching, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2018, pp. 2933–2940.

[22] Y. Wu, W. Liu, Z. Liu, G.S. Chirikjian, Learning-free grasping of unknown objects using hidden superquadrics, in: Robotics: Science and System, RSS, 2023.

[23] C. Goldfeder, M. Ciocarlie, H. Dang, P.K. Allen, The columbia grasp database, in: IEEE International Conference on Robotics and Automation, ICRA, 2009, pp. 1710–1716.

[24] A. Mitrevski, P.G. Plöger, G. Lakemeyer, Ontology-assisted generalisation of robot action execution knowledge, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, 2021, pp. 6763–6770.

[25] W. Liu, A. Daruna, S. Chernova, CAGE: Context-aware grasping engine, in: IEEE International Conference on Robotics and Automation, ICRA, 2020, pp. 2550–2556.

[26] V. Holomjova, A.J. Starkey, B. Yun, P. Meißner, One-shot learning for task-oriented grasping, IEEE Robot. Autom. Lett. 8 (12) (2023) 8232–8238.

[27] Y. Ju, K. Hu, G. Zhang, G. Zhang, M. Jiang, H. Xu, Robo-abc: Affordance generalization beyond categories via semantic correspondence for robot manipulation, in: The European Conference on Computer Vision, ECCV, 2025, pp. 222–239.

[28] A. Rashid, S. Sharma, C.M. Kim, J. Kerr, L.Y. Chen, A. Kanazawa, K. Goldberg, Language embedded radiance fields for zero-shot task-oriented grasping, in: The Conference on Robot Learning, CoRL, 2023.

[29] A. Radford, J.W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, I. Sutskever, Learning transferable visual models from natural language supervision, in: The International Conference on Machine Learning, ICML, 2021.

[30] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, A. Joulin, Emerging properties in self-supervised vision transformers, in: IEEE/CVF International Conference on Computer Vision, ICCV, 2021, pp. 9630–9640.
[31] M. Ji, R.-Z. Qiu, X. Zou, X. Wang, GraspSplats: Efficient manipulation with 3D feature splatting, in: The Conference on Robot Learning, CoRL, 2024.
[32] C. Tang, D. Huang, W. Dong, R. Xu, H. Zhang, FoundationGrasp: Generalizable task-oriented grasping with foundation models, IEEE Trans. Autom. Sci. Eng. 22 (2025) 12418–12435.
[33] Z. Wang, G. Tian, Task-oriented robot cognitive manipulation planning using affordance segmentation and logic reasoning, IEEE Trans. Neural Netw. Learn. Syst. 35 (9) (2024) 12172–12185.
[34] C. Li, G. Tian, M. Zhang, A semantic knowledge-based method for home service robot to grasp an object, Knowl.-Based Syst. 297 (2024) 111947.
[35] C. Tang, D. Huang, W. Ge, W. Liu, H. Zhang, GraspGPT: Leveraging semantic knowledge from a large language model for task-oriented grasping, IEEE Robot. Autom. Lett. 8 (11) (2023) 7551–7558.
[36] A.Z. Ren, B. Govil, T.-Y. Yang, K. Narasimhan, A. Majumdar, Leveraging language for accelerated learning of tool manipulation, in: The Conference on Robot Learning, CoRL, 2022.
[37] A.D. Vuong, M.N. Vu, H. Le, B. Huang, H.T.T. Binh, T. Vo, A. Kugi, A. Nguyen, Grasp-anything: Large-scale grasp dataset from foundation models, in: IEEE International Conference on Robotics and Automation, ICRA, 2024, pp. 14030–14037.
[38] W. Wan, K. Harada, F. Kanehiro, Planning grasps with suction cups and parallel grippers using superimposed segmentation of object meshes, IEEE Trans. Robot. 37 (2021) 166–184.
[39] P. Cignoni, M. Callieri, M. Corsini, M. Dellepiane, F. Ganovelli, R. Scopigno, MeshLab: an open-source mesh processing tool, in: Eurographics Italian Chapter Conference, 2008, pp. 129–136.
[40] OpenAI, GPT-4: Openai's generative pre-trained transformer, version 4, 2024, https://openai.com/index/gpt-4o.
[41] C. Yang, X. Wang, Y. Lu, H. Liu, Q.V. Le, D. Zhou, X. Chen, Large language models as optimizers, in: The International Conference on Learning Representations, ICLR, 2024.
[42] A. Khazatsky, K. Pertsch, et al., DROID: A large-scale in-the-wild robot manipulation dataset, in: Robotics: Science and System, RSS, 2024.
[43] P. Sun, S. Chen, C. Zhu, F. Xiao, P. Luo, S. Xie, Z. Yan, Going denser with open-vocabulary part segmentation, in: IEEE/CVF International Conference on Computer Vision, ICCV, 2023, pp. 15407–15419.
[44] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A.C. Berg, W.-Y. Lo, P. Dollár, R. Girshick, Segment anything, in: IEEE/CVF International Conference on Computer Vision, ICCV, 2023, pp. 4015–4026.
[45] P. Li, R. Wang, Y. Wang, W. Tao, Evaluation of the ICP algorithm in 3D point cloud registration, IEEE Access 8 (2020) 68030–68048.
[46] H.-S. Fang, C. Wang, M. Gou, C. Lu, GraspNet-1billion: A large-scale benchmark for general object grasping, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2020, pp. 11444–11453.
[47] S. Chen, W. Tang, P. Xie, W. Yang, G. Wang, Efficient heatmap-guided 6-dof grasp detection in cluttered scenes, IEEE Robot. Autom. Lett. 8 (8) (2023) 4895–4902.
[48] A. Tekden, M.P. Deisenroth, Y. Bekiroglu, Grasp transfer based on self-aligning implicit representations of local surfaces, IEEE Robot. Autom. Lett. 8 (10) (2023) 6315–6322.
[49] R. Wu, T. Zhu, X. Lin, Y. Sun, Cross-category functional grasp transfer, IEEE Robot. Autom. Lett. 9 (11) (2024) 10652–10659.
[50] A. Simeonov, Y. Du, A. Tagliasacchi, J.B. Tenenbaum, A. Rodriguez, P. Agrawal, V. Sitzmann, Neural descriptor fields: SE(3)-equivariant object representations for manipulation, in: IEEE International Conference on Robotics and Automation, ICRA, 2022, pp. 6394–6400.