

OVAL-Grasp: Open-Vocabulary Affordance Localization for Task Oriented Grasping

Edmond Tong^{*1}, Advaith Balaji^{*1}, Anthony Pipari¹,
Stanley Lewis¹, Zhen Zeng², and Odest Chadwicke Jenkins¹

¹ University of Michigan, Ann Arbor, MI, USA

² J.P. Morgan AI Research

Abstract. To manipulate objects in novel, unstructured environments, robots need task-oriented grasps that target object parts based on the given task. Geometry-based methods often struggle with visually defined parts, occlusions, and unseen objects. We introduce **OVAL-Grasp**, a zero-shot open-vocabulary approach to task-oriented, affordance based grasping that uses large-language models (LLM) and vision-language models (VLM) to allow a robot to grasp objects at the correct part according to a given task. Given an RGB image and a task, OVAL-Grasp identifies parts to grasp or avoid with an LLM, segments them with a VLM, and generates a 2D heatmap of actionable regions on the object. During our evaluations, we found that our method outperformed two task oriented grasping baselines on experiments with 20 household objects with 3 unique tasks for each. OVAL-Grasp successfully identifies and segments the correct object part 95% of the time and grasps the correct actionable area 78.3% of the time in real-world experiments with the Fetch mobile manipulator. Additionally, OVAL-Grasp finds correct object parts under partial occlusions, demonstrating a part selection success rate of 80% in cluttered scenes. We also demonstrate OVAL-Grasp’s efficacy in scenarios that rely on visual features for part selection, and show the benefit of a modular design through our ablation experiments. Our project webpage is available at <https://ekjt.github.io/OVAL-Grasp/>.

Keywords: Task-Oriented Grasping · Affordances

1 Introduction & Related Work

Robots in unstructured environments must identify objects and understand their *affordances*—the actions they enable. Affordance grounding links actions to object geometry. Most current methods are end-to-end, relying on supervised training with fixed affordances, which limits adaptability [2]. Recent work explores one or zero-shot (open vocabulary) affordance localization [9, 10]. However, they still require some fine-tuning, which can be costly and dataset-dependent. We present our method **OVAL-Grasp**, a modular and training free approach to task oriented, affordance-based grasping, that enables a robot to grasp an open set of objects at the correct part according to a task described by language. This

^{*} Equal contribution



Fig. 1. OVAL-Grasp at work on the Fetch mobile manipulator. The robot understands which parts of the object it should grasp and which parts should be avoided to fulfill the given tasks described by language.

paper presents our method, along with an evaluation of its performance compared to two state-of-the-art (SOTA) baselines, ShapeGrasp and GraspGPT [7, 19]. Figure 1 demonstrates OVAL-Grasp in action.

Large Language Models (LLMs) and Vision Language Models (VLMs) have emerged as powerful open-vocabulary tools [4], excelling in robotics applications like planning and navigation [15, 14]. Prior affordance localization approaches used fine-tuned LLMs [13] or text encoders [10]. In contrast, we explore affordance localization using pre-trained foundation models *without* fine-tuning. Task-Oriented Grasping (TOG) selects grasps based on task input and affordances [5]. Deep learning-based TOG methods [20] require supervision, while recent zero-shot approaches [19, 18, 7] aim to generalize but often still need training. In this work, we present the following contributions:

1. **OVAL-Grasp**, a method for zero-shot task-oriented grasping by leveraging LLMs and VLMs.
2. A modular design that enables performance improvements as individual components are enhanced.
3. A method that achieves superior performance over SOTA methods in real robotic experiments.
4. Experiments that demonstrate the capability of our method in novel TOG scenarios such as completing tasks that require both visual and geometric cues.

2 Problem Definition

In order for the robot to perform a successful zero-shot task oriented grasp given an RGB-D image $I \in \mathbb{R}^{H \times W \times C}$ as input, the robot must generate an end-effector pose $g \in SE(3)$ that lies on a part of an object x that facilitates the execution of a task t described by natural language. We assume that the robot has an egocentric RGB-D camera as well as a manipulator arm with a parallel jaw gripper capable of navigating to the grasp target g . We also assume access to a Large Language Model L , a vision-language part segmentation model V , and a grasp proposal model G . TOG methods can be evaluated by measuring their part identification and grasp success rate [7, 18, 19]. A task oriented grasp is

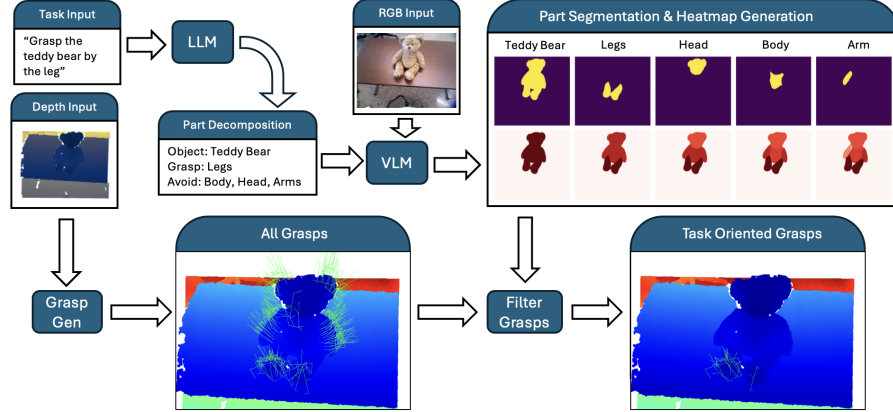


Fig. 2. System overview. The robot generates a task-oriented grasps by using an LLM to identify grasp-relevant object parts, a VLM to segment them, and a constructed heatmap to filter grasp candidates to produce a set of grasp that fulfill the given task.

considered successful if the proposed grasp location fulfills the given task based on commonsense reasoning.

3 OVAL-Grasp: Affordance-Prompting

OVAL-Grasp processes an RGB image and a task description to localize affordances. As shown in Figure 2, the language model L uses the task t to identify the object x , and decompose it into the desirable parts (to grasp), and the undesirable parts (to avoid). The part segmentation model V then segments these parts, which we use to generate a heatmap based on each part’s confidence score. The grasp generator G proposes candidate grasps for the entire object, which are subsequently scored using the heatmap.

Part Decomposition: We prompt L to identify and decompose the relevant object x into desirable and undesirable parts, where parts that facilitate task t are desirable, and vice-versa. We use the GPT-4o API [12].

Part Localization: We prompt V with the list of identified parts, producing a binary mask and confidence score for each part. We use PartGLEE [6] for its SOTA open-vocabulary part segmentation capabilities.

Heatmap Generation: OVAL-Grasp filters grasp candidates by composing a heatmap from relevant part segments and confidence scores. We generate a heatmap with the same height and width as the input image to score potential grasps. The heatmap H is initialized to zero, and the whole object segmentation is added with a positive value. The desirable part segments are added to the heatmap and undesirable segments are subtracted from the heatmap and are both scaled by their respective confidence scores. The values in the heatmap H were scaled to the range $[0, 255]$ and then smoothed using a Gaussian blur with a 3×3 kernel to remove any noise from adding segments.

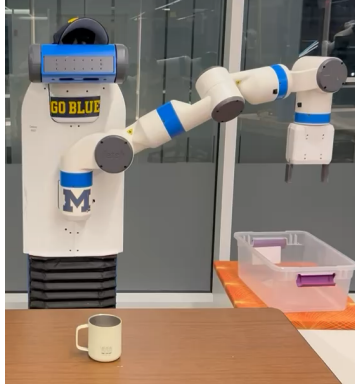


Fig. 3. Our experimental setup used the Fetch robot (left) and household and YCB objects (right) to evaluate OVAL-Grasp and baseline methods.

Grasp Generation and Filtering: Based on the previous modules’ outputs, OVAL-Grasp scores candidate grasps using the task-specific heatmap, selecting the highest-scoring pose for execution. We used ContactGraspNet [17] to generate the candidate grasps g_c , where the number of candidates ranges from 0 to N depending on the scene. Each candidate grasp is scored using H . We use two scores, the contact score ensures the grasp targets the correct region, while the z-axis score discourages approach angles that could obstruct task-relevant areas. The contact score $S_c(g_{c_i})$ is computed by reprojecting the gripper’s contact point (x_c, y_c) , provided by ContactGraspNet, into the heatmap H . From the object point cloud P , and the line z_g created by extending the z-axis of the grasp, we find point $p \in P$ that is closet to z_g . The z-axis score $S_z(g_{c_i})$ is calculated by reprojecting p to H , (x_p, y_p) .

$$S_c(g_{c_i}) = H(x_c, y_c) \quad S_z(g_{c_i}) = H(x_p, y_p) \quad S(g_{c_i}) = S_c(g_{c_i}) + S_z(g_{c_i}) \quad (1)$$

The grasps are then sorted by their total score $S(g_{c_i})$ which is calculated as shown in Equation 1. The highest score grasp is selected as the grasp target g .

4 Experiments & Results

We evaluate OVAL-Grasp in four aspects: (1) part identification and grasp success in table-top settings, (2) robustness in cluttered scenes, (3) vision-based reasoning for task-oriented grasping, and (4) an ablation study isolating the impact of different language and segmentation models.

4.1 Experiment Setup

We tested our method and a set of baseline task-oriented grasping algorithms using a Fetch and Freight Research Edition robot to grasp objects placed on a table. We compare our model against the baselines GraspGPT [19] and ShapeGrasp [7]. These models were chosen based on their state-of-the-art performance on task oriented grasping benchmarks. RGB-D images were captured using the

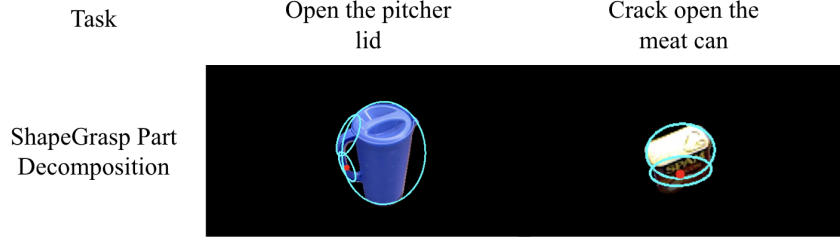


Fig. 4. Examples of ShapeGrasp failures. When dealing with more convex geometries and parts that are flush with the object, ShapeGrasp fails to identify the part.

onboard head camera. For each method, we evaluate grasp success through multiple trials. We control for object configuration by resetting the objects’ locations consistently between each trial and across all the methods tested. It is important to note that both GraspGPT and ShapeGrasp are given the ground truth segmentation mask of the relevant object to allow them to produce a predicted task oriented grasp; our method produces a task oriented grasp directly without needing a ground truth object mask. To evaluate the models, we tested them on 20 objects, each with three tasks requiring the robot to grasp the object in three different locations while positioned in front of them. The object set includes samples from the first and second baseline methods, objects from the YCB dataset [1], and additional objects not demonstrated in either baseline. Figure 3 shows the experimental setup and the objects tested. Each method was evaluated on two metrics—Part Selection Success and Grasp Success—similar to the metrics in [7]. Part Selection Success is the proportion of trials in which the system correctly labels the task-relevant part and (by commonsense visual inspection) generates a mask that covers 100% of the part’s visible region in the RGB image. Grasp Success is the proportion of trials in which the robot then executes a stable grasp on the identified part and lifts it off the table.

4.2 Grasp Success Experiment

The results of the Grasp Success Experiment are shown in Table 1. We find that OVAL-Grasp outperforms the other two baselines, achieving 35% better performance than GraspGPT and 21.7% better than ShapeGrasp in part selection, despite not having access to the ground truth object mask. GraspGPT struggles to generalize to unique and novel objects or tasks, frequently scoring all proposed grasps nearly equal. ShapeGrasp performs better but has difficulty decomposing objects with simple convex geometries, contiguous parts or parts flush with the object’s body. This is exemplified in Figure 4, where ShapeGrasp is unable to find the pitcher lid, and pull tab of the can. The few part-selection errors for OVAL-Grasp, accounting for 5% of failed trials, occur when V fails to generate a valid segment or when L hallucinates an incorrect decomposition, whereas the 21.7% of failed grasps result from G producing only marginal or no grasp candidates.

Table 1. Task-oriented grasping experiment on 20 objects with 3 tasks per object.

Method	Segment Source	Part Selection Success	Grasp Success	Time
GraspGPT [19]	Ground truth	60.0%	56.7%	128.67s
ShapeGrasp [7]	Ground truth	73.3%	66.7%	22.21s
OVAL-Grasp	PartGLEE[6]	95.0%	78.3%	19.86s

Our grasping success rate is higher than both baselines. ShapeGrasp’s grasping strategy relies on geometry and heuristics around the part’s centroid, which sometimes leads to grasp failures. GraspGPT’s grasps also have a lower success rate due to poor semantic understanding. In contrast, our method leverages an off-the-shelf grasp generator that proposes a variety of grasps, which are then filtered through an affordance-based heatmap, resulting in more effective overall grasping.

4.3 Runtime Evaluation

To assess runtime efficiency, we measured the average wall-clock time of each method on a system with an RTX 3090 GPU and an Intel i9-11900KF CPU. GraspGPT is an order of magnitude slower than both ShapeGrasp and our method due to repeated LLM queries. In contrast, ShapeGrasp and our approach require only a single query. However, none of the methods currently achieve real-time performance, as LLM inference introduces significant latency.

4.4 Visual Semantics

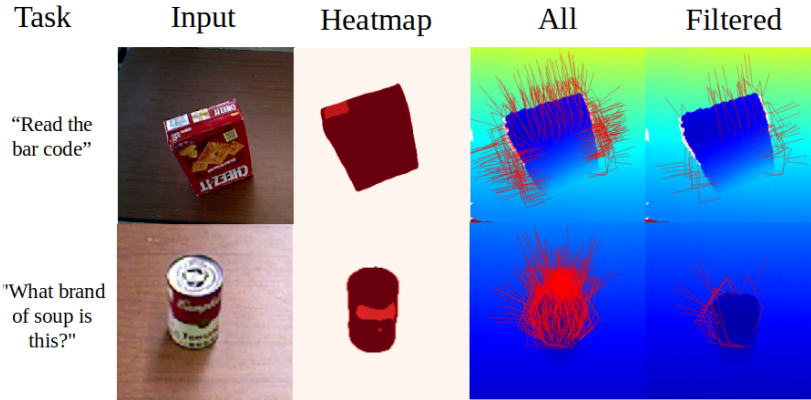


Fig. 5. OVAL-Grasp identifies object parts not linked to the object’s geometry. Scores are assigned to the barcode and soup can label segments in the heatmap and grasps that obstruct them are filtered out.

OVAL-Grasp is also able to reason about visual semantics. Some tasks require visual understanding beyond geometry, such as scanning a barcode or reading a label on a can or bottle, a common task in a retail or domestic environment. These parts are defined by visual features (e.g., colors and patterns) and cannot

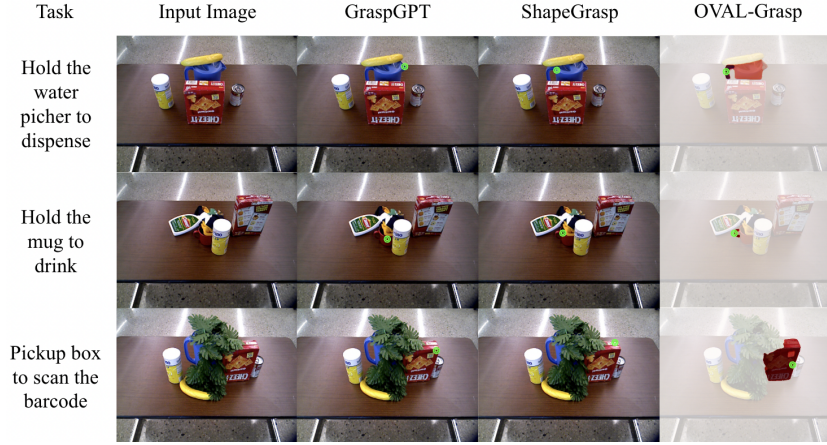


Fig. 6. Examples comparing OVAL-Grasp and baselines on part identification and grasp proposal in clutter. Proposed grasp point is highlighted in green. OVAL-Grasp successfully identifies the grasp point even with partially occluded parts.

Table 2. Task-oriented grasping experiment on 15 cluttered scenes

Method	Segment Source	Part Selection Success
GraspGPT [19]	Ground truth	26.7%
ShapeGrasp [7]	Ground truth	46.7%
OVAL-Grasp	PartGLEE[6]	80.0%

be captured purely by the object’s geometry. Unlike other baselines that rely solely on geometry, our system uses visual RGB input to segment parts, enabling more context-aware grasping. Figure 5 illustrates an example of this scenario, where our approach is able to correctly identify and avoid grasping the barcode and can label, both of which do not exist as part of the geometry.

4.5 Cluttered Scenes

Next, we set out to test each of the considered algorithms in cluttered scenarios that are common in everyday human environments. To do this, we evaluated our method and baseline approaches in cluttered scenes containing multiple objects with varying degrees of occlusion and overlap. Each method was tested on a similar set of tasks and objects as in the main experiment to ensure a fair comparison. While cluttered scenes fall outside the original scope of the other two methods, we included them to assess how robust our model would be in real-world settings. For this evaluation, we focused only on Part Selection Success, not on the success of the final grasp execution. The results are reported in Table 2 and are visualized in Figure 6.

OVAL-Grasp performed robustly in cluttered scenes with occluded objects, because it does not rely solely on geometric reasoning. In contrast, geometry-dependent methods like GraspGPT and ShapeGrasp struggled when the target object was partially hidden or fragmented. These methods rely on clear object

visibility and intact shape priors: GraspGPT, trained on complete point clouds, failed to generalize to fragmented inputs, while ShapeGrasp required canonical parts to be visible for accurate part identification. Both methods were prone to misidentifying object parts in clutter or when disjointed configurations made a single object appear as multiple parts. OVAL-Grasp, however, remained effective under occlusion and appearance shifts, successfully identifying task-relevant grasps even when parts were only partially visible or distinguishable by visual features alone.

4.6 Ablation on Modular Design

To evaluate the contribution of each component, we conducted an ablation study by replacing the part segmentation and language models. Using a subset of objects from the main experiments, we assessed performance based on part selection results.

Table 3. Ablation study demonstrating OVAL-Grasp’s performance with different part segmentation models and LLMs

Method	Segment Source	Language Model	Part Selection Success
OVAL-Grasp	PartGLEE	Deepseek-R1 (7B)	33.3%
	PartGLEE	GPT-3.5 Turbo	58.3%
	VLPart	GPT-4o	78.3%
	PartGLEE	GPT-4o	95.0%

Table 3 contains the results of our ablation experiments. We tested 3 language models—GPT-4o [12], GPT-3.5 Turbo [11], and DeepSeek-R1 [3] with the best-performing part segmentation model, PartGLEE [6]. We also tested 2 segmentation models, PartGLEE and VLPart [16], with the best performing language model GPT-4o. On the language model side, GPT-3.5 Turbo struggled with part-level reasoning, defaulting to object-level descriptions and showing higher rates of hallucination and prompt sensitivity. DeepSeek-R1 performed worst, likely due to its smaller model size (7B) and incorrect reasoning. These issues align with evidence that chain-of-thought reasoning can impair visual understanding when language struggles to capture spatial or part-level concepts [8]. On the segmentation side, VLPart underperformed compared to PartGLEE due to lower part detection accuracy. PartGLEE’s hierarchical modeling of object parts contributed to more reliable affordance localization. These results suggest that continued improvements in LLM reasoning and VLM segmentation will directly enhance system performance.

Despite its strong performance, OVAL-Grasp remains subject to hallucinations and other limitations of large pretrained models. Most failures stem from incorrect part decomposition by the LLM, either hallucinating non-existent parts or misclassifying graspable and ungraspable regions. Figure 7 shows two failure cases in cluttered scenes: one over-segments a mug due to high clutter, and the other incorrectly segments multiple handles from overlapping objects. In clutter,

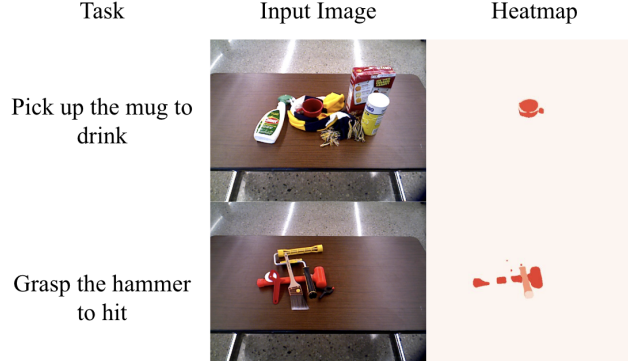


Fig. 7. Examples of part identification failures in clutter

the segmentation model becomes the main source of error. While LLM hallucinations still occur, occlusion often causes the segmentation model to misidentify or over-segment parts, even when given the correct part name. As our ablation studies show, these failures should decrease as individual components improve.

5 Conclusion

In this paper, we introduced OVAL-Grasp, a zero-shot task-oriented grasping framework that uses large language and vision-language models for open-vocabulary affordance localization. OVAL-Grasp outperforms GraspGPT and ShapeGrasp in part selection and grasp success, especially in cluttered scenes and visual part identification scenarios. By leveraging VLMs, it enables context-aware grasping beyond geometric cues. Its modular design ensures scalability as foundation models improve, advancing zero-shot grasping in robotics. The main limitations of our method stem from OVAL-Grasp’s open-loop reasoning method. It lacks feedback mechanisms to detect and recover from failed or incorrect grasps, preventing re-identification or regrasping with new information. The system also supports only single-step grasps and cannot plan multi-stage manipulations or recovery actions. Its reliance on pre-trained models reduces robustness, especially for niche or underrepresented object parts. Future work will integrate closed-loop feedback—using tactile, force, or visual signals to detect failures and active strategies like object reorientation, camera repositioning, or probing to improve part segmentation and affordance localization in unstructured environments.

5.1 Acknowledgement

This work was supported in part by a Qualcomm Innovation Fellowship, Amazon, and Ford Motor Company. We extend our gratitude to our sponsors.

References

1. Calli, B., Singh, A., Walsman, A., Srinivasa, S., Abbeel, P., Dollar, A.M.: The ycb object and model set: Towards common benchmarks for manipulation research. In:

- 2015 International Conference on Advanced Robotics (ICAR), pp. 510–517 (2015). DOI 10.1109/ICAR.2015.7251504
2. Chen, D., Kong, D., Li, J., Wang, S., Yin, B.: A survey of visual affordance recognition based on deep learning. *IEEE Transactions on Big Data* **9**(6), 1458–1476 (2023). DOI 10.1109/TBDATA.2023.3291558
3. DeepSeek-AI: Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning (2025). URL <https://arxiv.org/abs/2501.12948>
4. Driess, D., Xia, F., Sajjadi, M.S.M., Lynch, C., et al., A.C.: Palm-e: An embodied multimodal language model (2023). URL <https://arxiv.org/abs/2303.03378>
5. Kleeberger, K., Bormann, R., Kraus, W., Huber, M.F.: A survey on learning-based robotic grasping. *Current Robotics Reports* **1**, 239–249 (2020)
6. Li, J., Wu, J., Zhao, W., Bai, S., Bai, X.: Partglee: A foundation model for recognizing and parsing any objects. *arXiv preprint arXiv:2407.16696* (2024)
7. Li, S., Bhagat, S., Campbell, J., Xie, Y., Kim, W., Sycara, K., Stepputtis, S.: Shapegrasp: Zero-shot task-oriented grasping with large language models through geometric decomposition. In: 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 10,527–10,534. IEEE (2024)
8. Liu, R., Geng, J., Wu, A.J., Sucholutsky, I., Lombrozo, T., Griffiths, T.L.: Mind your step (by step): Chain-of-thought can reduce performance on tasks where thinking makes humans worse (2024). URL <https://arxiv.org/abs/2410.21333>
9. Luo, H., Zhai, W., Zhang, J., Cao, Y., Tao, D.: Learning affordance grounding from exocentric images. In: *CVPR* (2022)
10. Nguyen, T., Vu, M.N., Vuong, A., Nguyen, D., Vo, T., Le, N., Nguyen, A.: Open-vocabulary affordance detection in 3d point clouds. In: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 5692–5698 (2023). DOI 10.1109/IROS55552.2023.10341553
11. OpenAI: Gpt-3.5 turbo (2023). URL <https://platform.openai.com/docs/models/gpt-3-5>. Accessed via OpenAI API
12. OpenAI: ChatGPT (GPT-4o) (2025). URL <https://chat.openai.com>
13. Qian, S., Chen, W., Bai, M., Zhou, X., Tu, Z., Li, L.E.: Affordancellm: Grounding affordance from vision language models. *arXiv preprint arXiv:2401.06341* (2024)
14. Shah, D., Osinski, B., Ichter, B., Levine, S.: Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action (2022)
15. Song, C.H., Wu, J., Washington, C., Sadler, B.M., Chao, W.L., Su, Y.: Llm-planner: Few-shot grounded planning for embodied agents with large language models. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2023)
16. Sun, P., Chen, S., Zhu, C., Xiao, F., Luo, P., Xie, S., Yan, Z.: Going denser with open-vocabulary part segmentation (2023)
17. Sundermeyer, M., Mousavian, A., Triebel, R., Fox, D.: Contact-graspnet: Efficient 6-dof grasp generation in cluttered scenes. 2021 IEEE International Conference on Robotics and Automation (ICRA) (2021)
18. Tang, C., Huang, D., Dong, W., Xu, R., Zhang, H.: Foundationgrasp: Generalizable task-oriented grasping with foundation models. *IEEE Transactions on Automation Science and Engineering* (2025)
19. Tang, C., Huang, D., Ge, W., Liu, W., Zhang, H.: Graspgpt: Leveraging semantic knowledge from a large language model for task-oriented grasping. *IEEE Robotics and Automation Letters* **8**(11), 7551–7558 (2023)
20. Tang, C., Huang, D., Meng, L., Liu, W., Zhang, H.: Task-oriented grasp prediction with visual-language inputs. In: 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4881–4888. IEEE (2023)