LASO: Language-guided Affordance Segmentation on 3D Object

Yicong Li[1], Na Zhao[2]*, Junbin Xiao[1], Chun Feng[3], Xiang Wang[3]*†, Tat-seng Chua[1],

[1]National University of Singapore, [2]Singapore University of Technology and Design,
[3]University of Science and Technology of China,

liyicong@u.nus.edu,na_zhao@sutd.edu.sg,junbin@comp.nus.edu.sg
fengchun3364@mail.ustc.edu.cn,xiangwang1223@gmail.com,dcscts@nus.edu.sg

Abstract

Segmenting affordance in 3D data is key for bridging perception and action in robots. Existing efforts mostly focus on the visual side and overlook the affordance knowledge from a semantic aspect. This oversight not only limits their generalization to unseen objects, but more importantly, hinders their synergy with large language models (LLMs) which are excellent task planners that can decompose an overarching command into agent-actionable instructions. With this regard, we propose a novel task, Language-guided Affordance Segmentation on 3D Object (LASO), which challenges a model to segment a 3D object's part relevant to a given affordance question. To facilitate the task, we contribute a dataset comprising 19,751 point-question pairs, covering 8434 object shapes and 870 expert-crafted questions. As a pioneer solution, we further propose PointRefer, which highlights an adaptive fusion module to identify target affordance regions at different scales. To ensure a text-aware segmentation, we adopt a set of affordance queries conditioned on linguistic cues to generate dynamic kernels. These kernels are further used to convolute with point features and generate a segmentation mask. Comprehensive experiments and analyses validate PointRefer's effectiveness. With these efforts, We hope that LASO can steer the direction of 3D affordance, guiding it towards enhanced integration with the evolving capabilities of LLMs. Code and data are available at https://github.com/yl3800/LASO.

1. Introduction

Affordance describes specific regions on an object that enable or facilitate certain human interactions. In this context, the task of 3D affordance segmentation is raised to identify parts of 3D objects that imply a certain function. It has been deemed as a critical step in bridging perception and operation in the physical world for an embodied agent, thus has shown substantial impact on practical applications such as robotic manipulation [10, 14, 31].

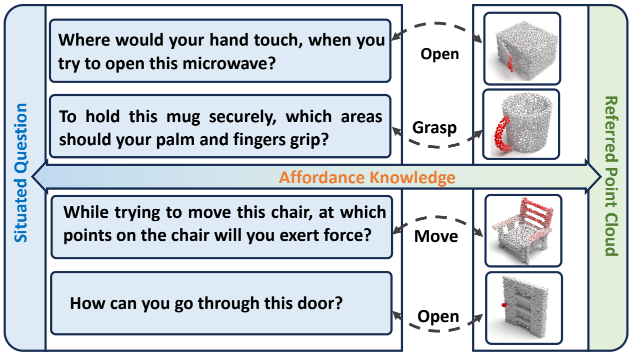

Situated Question
Where would your hand touch, when you try to open this microwave?
Open
To hold this mug securely, which areas should your palm and fingers grip?
Grasp
Affordance Knowledge
While trying to move this chair, at which points on the chair will you exert force?
Move
How can you go through this door?
Open
Referred Point Cloud

Figure 1. Illustration of **LASO**, language-guided affordance segmentation on 3D object. Given an object point cloud and a question, the model is expected to segment the functional part delineated in question.

Recently, the advancement of Large Language Models (LLMs) has enabled their deployment as task planners, breaking down an intricate instruction into a sequence of affordance-aware, robot-executable instructions [12, 35, 36]. This equips intelligent agents with the capability to undertake complex tasks. For example, the overarching command "Pass me the mug in the microwave" can be systematically deconstructed into a series of affordance-aware inquiries: Q1) "How to get to the microwave?", Q2) "How to open the microwave", Q3) "How to grab the mug?", and Q4) "How to return to the user?" Ideally, each query would correspond to a predefined subtask that effectively translates language instructions into actionable steps. While Q1 and Q4 are related to language-conditioned path planning [42] and navigation [37], Q2 and Q3, which involve the 'open microwave' and 'grab the mug' actions, ex-

[*] Corresponding author.
[†] Xiang Wang is also affiliated with Institute of Artificial Intelligence, Institute of Dataspace, Hefei Comprehensive National Science Center.

pose a gap — to the best of our knowledge, currently understanding affordance via language cues remains unexplored.

Leaving language cues untouched, current 3D affordance segmentation focuses primarily on the visual aspect. It associates the specific geo-structure with either explicit affordance categories [7] or implicit affordance information from the 2D showcase [44]. Such visual-only settings make the segmentation models confined to visually predefined affordance types, thus impeding integration with language-based instructions from LLMs. As a result, fulfilling the aforementioned user's request becomes challenging, due to this inability to address Q2 and Q3.

Towards these limitations, we introduce a novel task: Language-guided Affordance Segmentation on 3D Object (LASO), where 3D objects are paired with natural language questions that probe specific affordance parts. As a cornerstone task, LASO can provide an agent with affordance knowledge to answer the aforementioned Q2 and Q3 raised by LLM. For instance, the first two rows in Fig. 1 demonstrate how LASO imparts knowledge that enables an agent to interpret the semantic instructions "open the microwave" and "grasp the mug" in a 3D context, which are sufficient to execute Q2 and Q3, respectively. In LASO, questions are crafted to reflect diverse real-world scenarios, translating various affordance types into a range of semantic contexts. By learning from different contexts, we expect the agent not to memorize predefined affordance types but rather to abstract generalizable affordance knowledge within a broader semantic context. Notably, the diverse scales and shapes of affordance parts pose a significant perceptual challenge for the model. For instance, in the third row of Fig. 1, the target backrest of a chair occupies a large portion and has a significantly different shape compared to the small knob on a door in the last row.

To facilitate the study, we develop the 1st question-affordance dataset comprising 19,751 point cloud-question pairs. The questions, originally crafted by experts, are further diversified using GPT-4 [33] under principles of contextual enrichment, concise phrasing, and structural diversity. These principles enable a deep and varied exploration of object affordances. Based on the dataset, we have developed a strong baseline model named PointRefer that is capable of adapting to varying scales and shapes of object parts, conditioned on the question. It consists of two modules: 1) an Adaptive Fusion Module (AFM) that integrates question semantics to point features at different scales, which caters to varied target regions with better point feature discriminativeness; and 2) a Referred Point Decoder (RPD) that generates a segmentation mask via conditioned affordance queries, which, after refinement, forms dynamic convolution kernels to capture point-wise affordance for mask prediction. In summary, our contributions are as follows:

Table 1. **Efforts on affordance learning**. Part indicates whether the prediction specifies the object part (✓) or the whole object (×).

Domain	Studies	Vision	Text	Part
2D	[1, 46]	Image	×	×
	[9, 25, 32]	Image/Video	×	✓
	[24, 29]	Image	✓	×
3D	[7, 30, 30, 43]	Point Cloud	×	✓
	[44]	Point Cloud & Image	×	✓
	Ours	Point Cloud	✓	✓

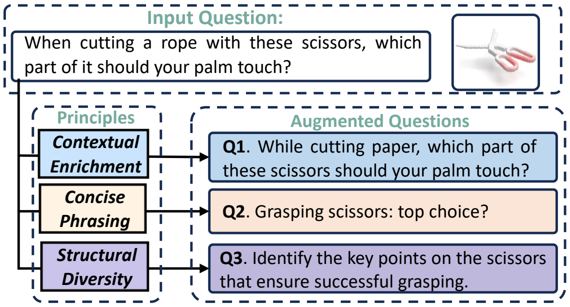
- We propose the task of Language-guided Affordance Segmentation on 3D Objects (LASO), which helps AI agents derive affordance knowledge from textual cues, facilitating a crucial advancement for LLM integration.
- We develop a dataset with 19,751 question-point affordance pairs which are meticulously curated via a collaborative effort of human experts and GPT-4, thereby establishing the 1st benchmark for evaluating LASO.
- We build a strong baseline PointRefer, which employs textual-conditioned affordance queries to isolate afforded segments and devises an adaptive fusion module to enhance the discriminability of point feature.

2. Related Work

Affordance Learning. Originating from the image domain, initial efforts concentrate on detecting objects with affordances [1, 46]. Progressing from there, later studies [24, 29] incorporate linguistic descriptions to augment the process but still neglect the granularity of analysis, generally focusing on object-level affordances. Addressing this oversight, subsequent research [9, 25, 32] shift towards scrutinizing specific affordance parts, establishing a new norm for precision in the field.

With the rise of embodied AI, the scope of affordance learning expands into the 3D realm. 3D AffordanceNet [7] introduces the first benchmark dataset for learning affordance from object point cloud geometry. Building on this, Yang et al. [44] propose a setting for learning 3D affordance parts guided by image demonstrations. Nonetheless, these approaches predominantly rely on a visual-only approach, tightly coupling geometric features with affordance labels and overlooking the semantic dimension. This limitation hinders the integration with Large Language Models (LLMs) that could otherwise substantiate affordance segmentation to real-world deployment. Moreover, the common practice of predicting affordance types as an auxiliary task [7, 44] may dilute the primary challenge of understanding affordance knowledge. In contrast, our work sidesteps auxiliary predictions, advocating for direct learning from the linguistic context, which aligns more closely with the innate capabilities of LLMs and their semantic richness.

Text-Point Cloud Cross-Modal Learning. The integration of vision and natural language has recently sparked increased interest [16–21, 41]. Such a tendency naturally extends to 3D vision due to its pivotal role in embodied AI. To



When cutting a rope with these scissors, which part of it should your palm touch?

Q1. While cutting paper, which part of these scissors should your palm touch?

Q2. Grasping scissors: top choice?

Q3. Identify the key points on the scissors that ensure successful grasping.

Figure 2. Illustration of question augmentation principles.

advance the 3D-language tasks, datasets like ScanRefer [5] and ReferIt3D [2] have been instrumental in benchmarking the grounding of natural language to 3D objects. Complementing this, Azuma et al. [3] introduce ScanQA for 3D question-answering, while Ma et al. [28] propose SQA3D for situated reasoning within 3D environments. In terms of models, a variety of efforts [3, 13, 45] have emerged from these benchmarks, with notable examples like 3D-SPS [26] and BUTD-DETR [13] employing cross-attention mechanisms and linguistic cues for object discovery. Others, such as 3DVG [45], and ViL3DRel [6], focus on 3D grounding by incorporating spatial relation cues, showcasing the evolving landscape of 3D-VL models. Unlike the existing work that focuses on grounding and QA that perform scene-level reasoning, we tackle a dense prediction task at object *part* level that bridges reasoning to operation. To the best of our knowledge, we are *the first* to introduce the 3D affordance problem as a language-guided segmentation task.

3. Dataset

To underpin our proposed task, we have meticulously compiled a question-point affordance dataset with 19,751 paired samples, featuring 8,434 distinct object shapes across 23 classes. Accompanying these are 870 expertly crafted questions, covering 17 affordance categories.

3.1. Annotation Details

Point Cloud. Our dataset leverages the point cloud and affordance annotations from the 3D-AffordanceNet [7]. Specifically, the affordance annotation in [7] is multi-class, with each point on an object potentially supporting multiple affordance types. In our case, each question is tailored to a specific affordance type, while each sample can associate with several questions. Consequently, the same object can exhibit diverse affordance segmentation, depending on the specific question posed.

Question. To create the questions, we draw 58 object-affordance combinations from [7], and craft 15 unique questions for each of the 58 combinations, resulting in 870 tailored questions. Specifically, five manual questions per combination are initially composed to represent different scenarios. Subsequently, we employ GPT-4 [33] to diversify the dataset with 10 additional questions for each combination, adhering to three key principles (as shown in Fig. 2): 1) *Contextual Enrichment.* Some questions were expanded to include additional context or details, fostering a more precise connection to the specified affordance of the target object. For instance, Q1 in Fig. 2 is reformed by a functional situation “cutting paper”. 2) *Concise Phrasing.* In other cases, questions were distilled to their essence, making them succinct yet still meaningful. Q2 in Fig. 2 exemplifies this approach. 3) *Structural Diversity.* We employed a range of sentence structures, from questions to statements, to introduce natural linguistic variations and prevent model bias toward any specific phrasing or sentence length. A diversified example can be found in the Q3 of Fig. 2. All questions generated by GPT-4 undergo manual verification after this augmentation process.

Table 2. Dataset Statistics

Task	Train			Val			Test		
	Shape	Qst	Inst	Shape	Qst	Inst	Shape	Qst	Inst
Seen	6883	638	16120	516	58	1215	1035	174	2416
Unseen	6160	458	11558						

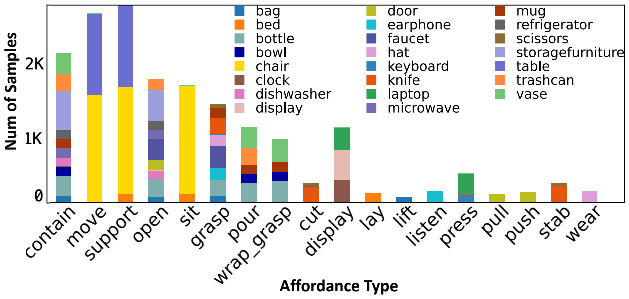


Figure 3. Landscape of our dataset.

It’s worth mentioning that the affordance type was only a tool for crafting the questions and pinpointing the correct segmentation areas. We refrain from using explicit affordance labels during both training and testing. This deliberate choice ensures that the learning process is driven by the semantic content of the questions, rather than by direct affordance label association.

3.2. Statistics and Setting

Dataset Settings. We show the statistical landscape of our dataset in Fig. 3. Specifically, it contains 17 affordance types and 23 object categories, composing 58 unique object-affordance combinations.

In the spirit of the work by [44], we offer two distinct dataset settings: **Seen**: The default configuration, where the training and testing phases share similar distributions of object classes and affordance types. **Unseen**: This configuration is purpose-built to test the model’s ability to generalize affordance knowledge. Certain affordance types, when paired with specific object classes, are omitted from the training set but included in the test set. For instance, while the model may learn to grasp bags and mugs during training, it is expected to apply the concept of 'grasp' to earphones, which is an affordance-object pairing not encountered during training. For a comprehensive account of the unseen setting, please refer to Appendix Sec. 7.1.

Training and Evaluation Protocals. Notably, since questions are created based on a combination of object class and affordance type, one object class can have many shape instances. Thus, the design of our dataset accounts for the multiplicity of shapes within each object class, leading to a non-bijective pairing between shape instances and corresponding questions. To ensure comprehensive learning, during training, each shape instance is matched with a randomly selected question that aligns with its affordance type for every iteration. This random pairing exposes the model to a variety of semantic contexts, bolstering its ability to generalize. For the validation and test sets, question pairings are fixed to minimize variability and ensure consistent inference. These questions are unique to the evaluation phase and are not revealed during the training process to maintain the integrity of the evaluation.

4. Method

LASO Task Definition. Given a question Q_{raw} and an object point cloud $\mathbf{P}_{raw} \in \mathbb{R}^{N\times 3}$ with N points, the goal of LASO is to predict a binary mask of $\mathbf{M} \in \mathbb{R}^{N}$ that segments the functional part related to the question.

Framework Overview. Due to the varying scale of target affordance region, LASO naturally challenges models adaptiveness at different scales. To address this, we devise a strong baseline, named PointRefer. As shown in Fig. 4, the object point cloud $\mathbf{P}_{raw}$ is first processed by a 3D backbone, which typically consists of the multi-stage encoding and decoding process. To adapt PointRefer to the point feature at multiple scales, we introduce an Adaptive Fusion Module (AFM) to the backbone's decoding process. It performs multi-scale cross-modal fusion by injecting text clues to point features at a different decoding stage, which progressively refine the point feature map in a top-down manner. Then, to predict the segmentation mask, we introduce a Referred Point Decoder (RPD) that leverages a set of learnable queries conditioned on the input question, termed *affordance queries*, as the input of the transformer decoder. These affordance queries are obligated to look at the referred points only and generate question-aware dynamic kernels. The final segmentation mask is obtained by convoluting these dynamic kernels with the AFM-enhanced point feature.

Feature Encoding. For the question, we prepend a "[CLS]" token to the question sequence before feeding them to a language model to capture the global language context. The encoded feature of the question is denoted by $\mathbf{X} \in \mathbb{R}^{L\times d}$, where L is the number of tokens and d is the feature dimension. For the point cloud, PointRefer can adopt any off-the-shelf 3D segmentation backbone. [1] to transform the $\mathbf{P}_{raw}$ into the feature space. Note that a 3D backbone for segmentation typically consists of several encoding and decoding stages, and we take the point-wise feature map after the last decoding stage as the output of the backbone.

4.1. Adaptive Fusion Module

AFM is designed to enhance the point-wise features by incorporating question information. As illustrated in Fig. 4, we integrate AFM into different decoder layers to enhance its adaptiveness to target regions of various scales and shapes. Given the conventional downsampling operators in 3D backbones (*e.g.* PointNet++ [34]), the number of points varies across different encoder/decoder layers. Our AFM is designed to seamlessly adapt to the varying number of points.

For simplification, we omit the notation for different decoder layers and use $\mathbf{P} \in \mathbb{R}^{T\times d}$ to denote the point feature at a certain decoding layer, where T represents the number of points at that layer. As visualized in Fig. 4, AFM adopts a bottleneck architecture with three key steps: Grouping, Mixing, and Ungrouping. In the grouping step, the point features are grouped by text tokens, these grouped tokens are then thoroughly mixed via an MLP-Mixer [38]. Subsequently, the ungrouping step encapsulates the mixed features into individual point features, completing the AFM procedure by embedding valuable textual information into the point feature.

To facilitate an efficient grouping process for our AFM, we employ a lightweight cross-attention module by omitting the transformation of both query and value. It takes the question feature $\mathbf{X}$ as query and the point feature $\mathbf{P}$ as key and value, then output grouped token $\mathbf{G} \in \mathbb{R}^{L\times d}$:

$$\mathbf{G} = \text{Attention}(\mathbf{X}, \mathbf{W}_1\,\mathbf{P}, \mathbf{P}) + \mathbf{X}, \tag{1}$$

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}}\right)\mathbf{V}, \tag{2}$$

where $\mathbf{W}_1$ is a linear transformation.

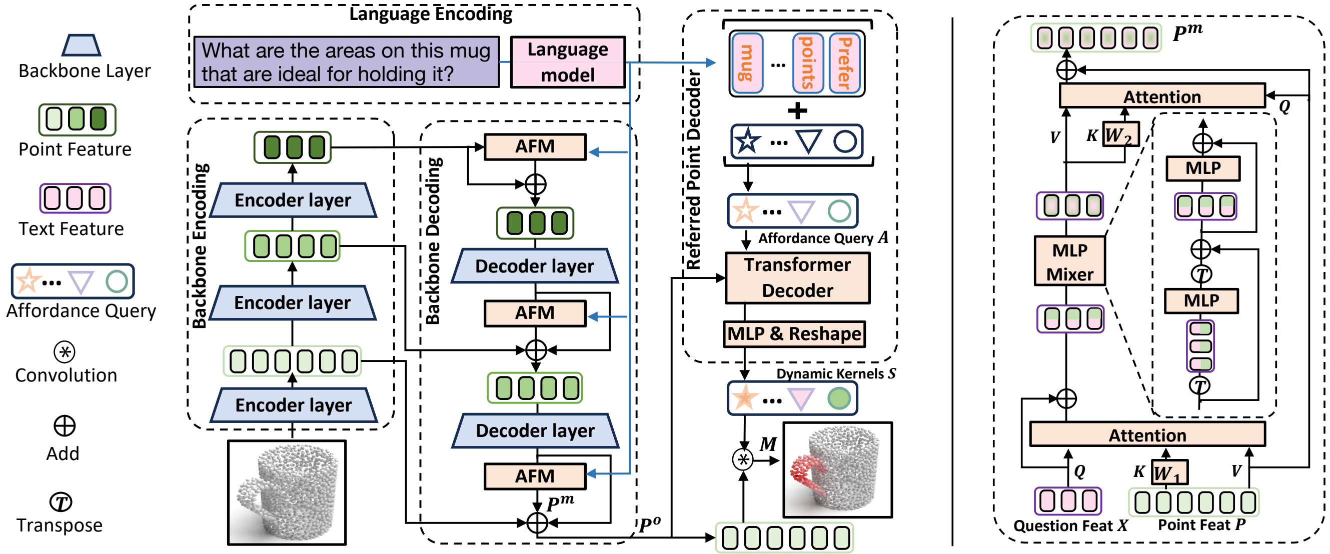
After attaining the grouped feature, we update them using an MLP-Mixer [38], which employs two consecutive multilayer perceptrons (MLPs), producing fused feature $\mathbf{F}$ by combining with two consecutive MLPs.

$$\mathbf{G}' = \mathbf{G} + \text{MLP}_1(\mathbf{G}^T)^T, \tag{3}$$

$$\mathbf{F} = \mathbf{G}' + \text{MLP}_2(\mathbf{G}'), \tag{4}$$

where, MLP_1 and MLP_2 are used to mix group-wise and channel-wise information, respectively, and $T\,[\cdot]$ denotes the transpose operation. Finally, we return the fused feature $\mathbf{F}$ to the point-space via the *ungrouping* process and

[1] The impact of backbone choices can be observed in Tab. 5.



(a) Overview of PointRefer.

(b) Adaptive Fusion Module.

Figure 4. **The overall framework of our proposed PointRefer** (a), which contains an **adaptive fusion modal** (b) that can be injected in different stages of the backbone decoder. As a result, it takes the point-wise feature generated at the last decoder layer as the final output of the 3D backbone. Then, the Referred Point Decoder takes in a set of question-conditioned affordance queries and feeds it to a transformer decoder together with the fused point feature, where the output is then reformed to a set of dynamic kernels. Finally, these dynamic kernels are used as convolution kernels to filter out the segmentation mask from the fused point feature.

acquire the fused point feature $\mathbf{P}^m$. We implement the ungrouping process as a similar lightweight attention module as the grouping process. However, in this case, it takes $\mathbf{P}$ as query and $\mathbf{F}$ as key and value:

$$\mathbf{P}^m = \text{Attention}(\mathbf{P}, \mathbf{W}_2\,\mathbf{F}, \mathbf{F}) + \mathbf{P}, \tag{5}$$

where $\mathbf{W}_2$ is a linear transformation. Lastly, the final output $\mathbf{P}^o$ is formed by adding residual connections to $\mathbf{P}^m$. Notably, we take $\mathbf{P}^o$ after the last decoding stage as the output of the 3D backbone.

4.2. Referred Point Decoder

Inspired by the success of learnable query-based methods in object detection [4] and segmentation [40], we introduce *learnable affordance queries* conditioned on the corresponding question of an object to decode the segmentation mask for the object. Specifically, we impose linguistic restrictions on all affordance queries, narrowing the model's focus to the relevant object parts and thereby facilitating convergence. These affordance queries, after being processed by a transformer decoder, are repurposed as dynamic kernels to deduce the segmentation mask from the point features, which are the final output of the 3D backbone.

Question-Conditioned Affordance Query In the transformer decoder, it is well known that content and position embeddings are responsible for encoding instance-specific and spatial information, respectively. In PointRefer, we feed these two parts with the text feature and learnable query parameters, respectively, so that all the queries are shaped by the nuances of the linguistic input. Specifically, the affordance queries $\mathbf{A}$ are formed by adding question embedding $\mathbf{X}$ with learnable embeddings. Thus, the resulting $\mathbf{A} \in \mathbb{R}^{L\times d}$ are the same length of the question.

Reasoning and Mask Prediction. After forming the affordance query, we feed it to a transformer decoder [39] together with the fused point feature $\mathbf{P}_o$ from the 3D backbone. In this manner, all the queries will use the question context as guidance and target to aggregate the feature of the referred points from the object, resulting in $\mathbf{A}'$:

$$\mathbf{A}' = \text{Transformer-Decoder}(\mathbf{A}; \mathbf{P}_o). \tag{6}$$

Subsequently, we perform dynamic convolution by applying a two-layer MLP on top of the transfer decoder to produce L dynamic kernels $\Omega = \{\omega_i\}_{i=1}^{L}$, where each kernel ω_i is reshaped to a 1×1 convolutional kernel with the channel number of d.

$$\Omega = \text{MLP}(\mathbf{A}'). \tag{7}$$

Since the dynamic kernels have captured the question- information, we use them as convolution filters on the feature maps for mask decoding. Specifically, we use each dynamic kernel in Ω to convolute the point-wise feature map $\mathbf{P}_o$, which returns L point masks $\mathbf{S} = \{\mathbf{s}_i\}_{i=1}^{L}$.

$$\mathbf{S}_i = \{\mathbf{P}_o * \omega_i\}_{i=1}^{L}\,. \tag{8}$$

Then, we apply mean pooling over all masks in $\mathbf{S}$ followed by a sigmoid activation to acquire the final segmentation mask $\mathbf{M} \in \mathbb{R}^N$:

$$\mathbf{M} = \sigma(\text{Max-Pool}((\mathbf{S}))), \tag{9}$$

where $\sigma(\cdot)$ denote the sigmoid function.

Intuitively, each dynamic kernel is expected to capture one aspect of the referred object part according to conditioned the question token. By ensembling all generated point masks, the final segmentation mask can integrate different response regions, thus making the result more robust.

Objectives. Unlike [44], which relies on an auxiliary affordance label for prediction, our model seeks to forge a direct link between language context and object affordance. Thus, we solely employ Dice loss and Binary Cross-Entropy (BCE) loss to guide the segmentation mask prediction, bypassing the need for additional affordance labels.

$$\mathcal{L} = \mathcal{L}_{BCE} + \mathcal{L}_{Dice} \tag{10}$$

5. Experiment

Evaluation Metrics. For a thorough assessment, we benchmark our dataset against leading studies in 3D affordance learning [7, 44], employing four evaluation metrics to ensure robust analysis. These include Area Under the Curve (AUC), Mean Intersection Over Union (mIoU), Similarity (SIM), and Mean Absolute Error (MAE). Appendix Sec. 7.2 provides detailed explanations for the evaluation protocols.

Implementation Details. PointRefer employs PointNet++ as the default 3D backbone to align with the standards established in [7, 44]. For text encoding, we utilize a pre-trained RoBERTa model [23] to process linguistic inputs. The feature dimension d is set to 512. During training, we use the Adam optimizer with a learning rate set to 1e-4.

Baseline Models. Since there is no prior work using paired question-point cloud data to segment object affordance. For a thorough comparison of our method, we adopt two 3D cross-modal works (3D-SPS [26] and IAGNet[44]) and two referred image segmentation works (ReferTrans [15] and RelA [22]) to LASO task. For the referred image segmentation models, we substitute their image backbones with a 3D counterpart, retaining their fusion mechanisms and mask decoding strategies. For the language-guided grounding model, 3D-SPS [26], we omit its bounding box prediction module while leveraging its inherent point selection scheme for our segmentation setting. The most related work is IAGNet [44], an affordance detection method that takes paired image-point cloud as input. To adapt IAGNet to our needs, we simply replace its image backbone with a language model, preserving the remainder of its architecture.

Next, we show the effectiveness of our PointRefer by answering the following questions:

- **Q1:** How is PointRefer compared to other baselines on the proposed dataset?
- **Q2:** How effective are the proposed components?
- **Q3:** What learning pattern does the PointRefer capture?

Table 3. **Main Results**. The overall results of all comparative methods, the best results are in bold. Seen and Unseen are two partitions of the dataset. AUC and aIOU are shown in percentage.

	Method	mIoU↑	AUC↑	SIM↑	MAE↓
Seen	ReferTrans [15]	13.7	79.8	0.497	0.124
	ReLA [22]	15.2	78.9	0.532	0.118
	3D-SPS [26]	11.4	76.2	0.433	0.138
	IAGNet [44]	17.8	82.3	0.561	0.109
	PointRefer	**20.8**	**87.3**	**0.629**	**0.093**
Unseen	ReferTrans [15]	10.2	69.1	0.432	0.145
	ReLA [22]	10.7	69.7	0.429	0.144
	3D-SPS [26]	7.9	68.8	0.402	0.158
	IAGNet [44]	12.9	77.8	0.443	0.129
	PointRefer	**14.6**	**80.2**	**0.507**	**0.119**

5.1. Main Result (Q1)

In Table 3, PointRefer demonstrates superior performance across all evaluation metrics compared to the baseline methods. Our observations are as follows:

PointRefer vs. Other Models: IAGNet secures the second-best performance in both seen and unseen settings for all metrics. This is likely due to its specific design for object affordance segmentation, despite differences in input modality, making it a strong fit for our task with minimal adaptation. In contrast, 3D-SPS underperforms, potentially due to its progressive point selection mechanism. This approach involves a non-differentiable selection process, which may complicate the optimization within our task framework. Image-based methods, while showing decent performance, appear to be constrained by the modality gap. Originally trained on vast datasets of image-text pairs, where text is often a brief label, these methods now face fully formed, context-rich questions with a more limited training corpus.

Seen vs. Unseen Performance: A notable performance drop from seen to unseen settings is observed across all baselines. This discrepancy underscores the difficulty in learning affordance knowledge that generalizes well, corroborating the necessity of our task design. Current methods struggle to bridge this gap in knowledge transfer, suggesting an avenue for future research: utilizing the extensive commonsense knowledge embedded in LLMs could potentially mitigate this challenge.

5.2. In-Depth Study (Q2)

Ablation Study. We show ablative results in Tab. 4. To verify that PointRefer prediction is based on the question, we conduct a blind test by training without questioning information, where the distinctive performance gap indicates that PointRefer has comprehended the question and can make a prediction based on it. Then, we study the effectiveness of PointRefer by removing both AFM and PRD ("w/o AFM & PRD"), which induces a severe performance

Table 4. Ablative Results.

Variants	mIoU↑	AUC↑	SIM↑	MAE↓
Blind	10.8	78.2	0.506	0.125
w/o AFM & PRD	17.7	82.1	0.558	0.110
w/o AFM	19.8	83.9	0.592	0.104
w/o multi-scale	20.4	87.1	0.628	0.095
w/o PRD	18.7	85.6	0.619	0.098
w/o condition	19.2	86.9	0.590	0.100
w/o dynamic filter	19.5	87.1	0.623	0.096
PointRefer	**20.8**	**87.3**	**0.629**	**0.093**

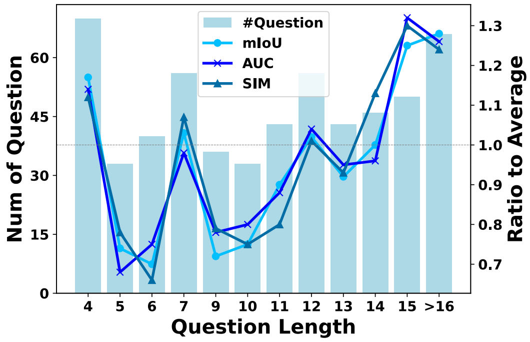


Figure 5. The bar chart shows the training sample distribution over different question lengths, and the line chart shows the performance of PointRefer grouped by question length.

decline on every metric, thus, validating the overall effectiveness of our design. Then, we also verify the functionality of AFM via the performance drop when it is removed ("w/o AFM). As a comparison, a relatively slighter declination is witnessed when we only apply AFM at the single scale ("w/o multi-scale"), which indicates the benefit of multi-scale cross-modal fusion. Then, we conduct some detailed breakdown tests to study PRD, we notice that when the affordance queries are not conditioned on the question ("w/o condition") the performance drops drastically, indicating that our affordance queries help to capture semantic clues in 3D space. Finally, we validate the superiority of the dynamic filtering strategy, because a declination is also observed when the dynamic filter is erased ("w/o dynamic filter "), and this variant adopts IAGNet's [44] mask prediction strategy as an alternative.

Effect of Question Length. In Fig. 5, we show the performance of PointRefer on different question lengths. Specifically, The bar plot reveals the distribution of question length in the training set. Meanwhile, the line chart illustrates PointRefer's performance on different question lengths. Here, the performances are normalizing against the average number under that metric, *e.g.* $\frac{\text{IoU for each question length}}{\text{mIoU}}$. Notably, PointRefer excels with questions that are more prevalent in the training set and with those that are lengthier—exceeding 14 words—owing to the richer contextual information they provide, which is instrumental in discerning the object's affordance.

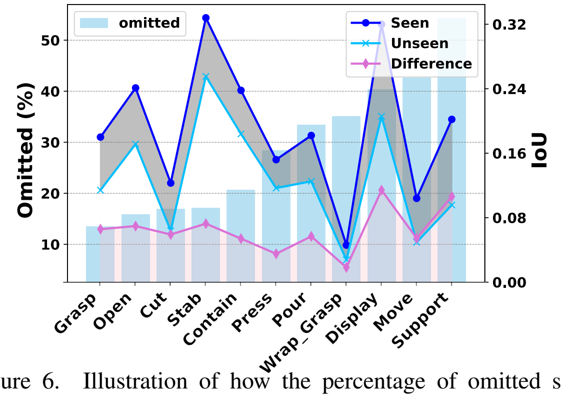


Figure 6. Illustration of how the percentage of omitted samples affect transferability of affordance knowledge (indicated by the difference between seen and unseen). For each affordance type, the bar shows a number of omitted samples from seen, *i.e.* $\frac{\#\text{Seen - \#Unseen}}{\#\text{Seen}}$(%), and the lines show the testing IoU of models trained on Seen and Unseen. Difference (IoU): Seen - Unseen.

Table 5. **Study of Backbones.** PN:PointNet++; PM:PointMLP.

	3D	LM	mIoU↑	AUC↑	SIM↑	MAE↓
Seen	PM [27]	Bert [8]	18.2	85.5	0.618	0.099
		Deberta [11]	19.3	86.5	0.621	0.096
		Roberta [23]	19.6	86.4	62.3	0.097
	PN [34]	Bert [8]	20.1	86.4	0.615	0.101
		Deberta [11]	20.4	87.0	0.624	0.095
		Roberta [23]	**20.8**	**87.3**	**0.629**	**0.093**
Unseen	PM [27]	Bert [8]	11.5	76.3	0.43	0.128
		Deberta [11]	12.3	76.8	0.435	0.133
		Roberta [23]	12.1	76.8	0.434	0.136
	PN [34]	Bert [8]	14.1	78.1	0.47	0.122
		Deberta [11]	14.5	78.9	0.50	**0.114**
		Roberta [23]	**14.6**	**80.2**	**0.507**	0.119

Seen vs. Unseen. Recall that to create the training set for the unseen setting, for an affordance type, we omit its combination with certain objects from seen. For example, the seen partition has "grasp-mug", and "grasp-bag" in its training, we create the unseen training set by removing the "grasp-mug" from the seen training set. In this case, the model is expected to learn the generalizable affordance knowledge of "grasp" and transfer it to an unseen object during testing. Note that seen and unseen setting shares the same validation and testing set. To investigate how PointRefer captures such transferability for each affordance type, The bar in Fig. 6 shows the percentage of samples that are omitted from seen under each affordance type. Then, the lines show the performance of the model trained with seen and unseen partitions for each affordance type, and their difference—the indicator of transferability of learned affordance knowledge. In general, we notice that the curve of seen and unseen follows a similar pattern, which indicates that PointRefer can capture a similar affordance knowledge even trained with far less object class. However, we also notice that the difference between seen and unseen enlarges as the omitted percentage grows, regardless of the absolution IoU. This is expected since the removed object can lead to under-diversified affordance representation.

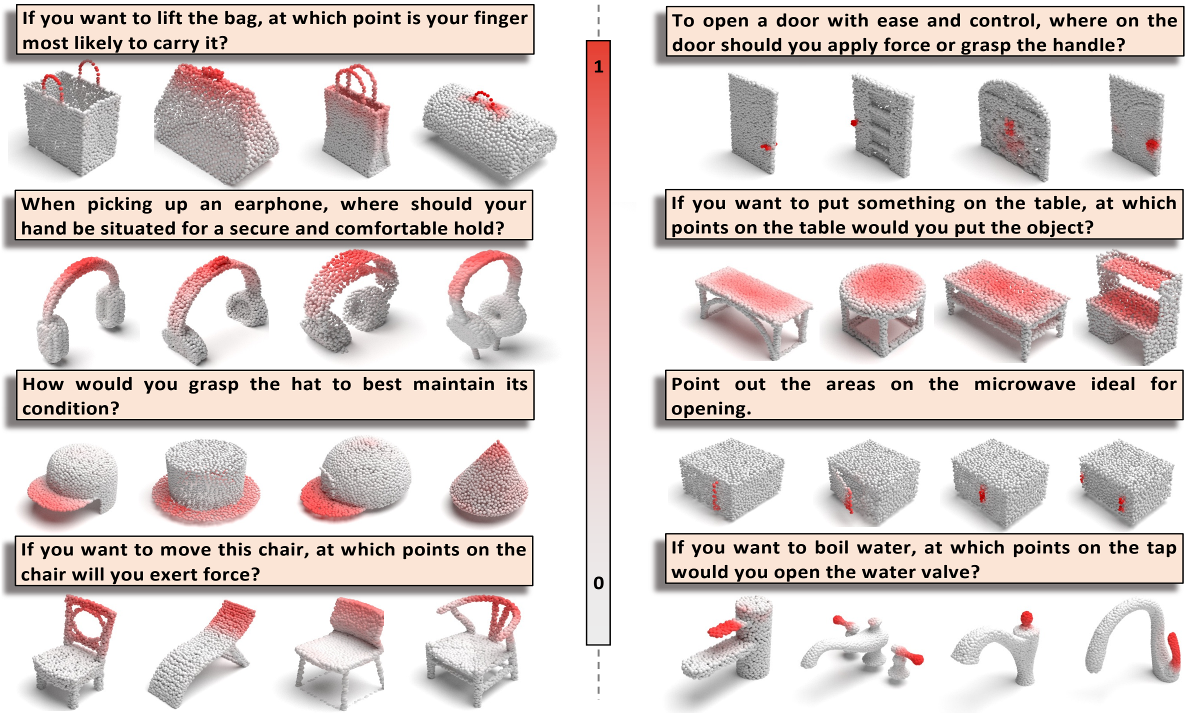

If you want to lift the bag, at which point is your finger most likely to carry it?
When picking up an earphone, where should your hand be situated for a secure and comfortable hold?
How would you grasp the hat to best maintain its condition?
If you want to move this chair, at which points on the chair will you exert force?
1
0
To open a door with ease and control, where on the door should you apply force or grasp the handle?
If you want to put something on the table, at which points on the table would you put the object?
Point out the areas on the microwave ideal for opening.
If you want to boil water, at which points on the tap would you open the water valve?

Figure 7. Case-Study of PointRefer's segmentation. Each showcase comes with one question and four shapes, showing the generalization ability of the affordance knowledge. The segmented affordance part is highlighted in red.

Choice of Backbone. To Study the influence of the backbone, we investigate the Performance of PointRefer with some alternative backbone. We notice from Tab. 5 that, for both seen and unseen settings, using PointNet++ as a 3D backbone generally provides much better results than PointMLP, thus, we set PointNet++ as our default 3D backbone. As for the language model, the performance of Roberta prevails Bert and Deberta in the seen setting. Whereas in the unseen setting, the use of Deberta and Roberta brings no significant difference. Given that, PointRefer uses Roberta as the default language model.

5.3. Qualitative Results

Case Study. As shown in Fig. 7, our model demonstrates the capacity to accurately comprehend object affordance given the question. It is noteworthy that even when dealing with small affordance components, such as the microwave door handle, our model still exhibits decent ability.

Failure Analysis. As shown in Fig. 8, we identify two types of failure cases. The first case is attributed to non-comprehensive text. The overly brief text makes PointRefer struggle to encapsulate the necessary affordance knowledge, thus, failing to segment the relevant part. Besides, we also encounter an over-segmentation issue, which occurs when multiple independent affordance parts are listed as targets. In this case, our model attempts to ”connect” disparate segments, inadvertently leading to the segmentation of irrelevant middle parts.

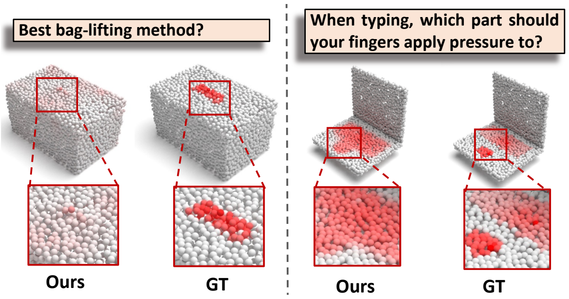

Best bag-lifting method?
When typing, which part should your fingers apply pressure to?
Ours
GT
Ours
GT

Figure 8. Failure cases due to non-comprehended text (left) and over-segmentation (right).

6. Conclusion

In this paper, we have introduced the pioneering task of Language-guided Affordance Segmentation on 3D Objects (LASO), which establishes a vital connection between AI agents and Large Language Models (LLMs) via textual cues. Our contributions include the development of an extensive dataset consisting of 19,751 question-point pairs and the creation of PointRefer, a baseline model that lays the groundwork for the LASO task. We envisage that LASO will steer the trajectory of the 3D affordance field towards better integration with the advancements in LLMs.

7. Acknowledgments

This work is supported by the National Natural Science Foundation of China (92270114), the Agency for Science, Technology and Research (A*STAR) under MTC Programmatic Funds (Grant No.M23L7b0021), and NExT center.

References

[1] Do, etc. Affordancenet: An end-to-end deep learning approach for object affordance detection. In *ICRA*, 2018. 2

[2] Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas J. Guibas. Referit3d: Neural listeners for fine-grained 3d object identification in real-world scenes. In *ECCV*, pages 422–440. Springer, 2020. 3

[3] Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. Scanqa: 3d question answering for spatial scene understanding. In *CVPR*, pages 19107–19117. IEEE, 2022. 3

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, pages 213–229. Springer, 2020. 5

[5] Dave Zhenyu Chen, Angel X. Chang, and Matthias Nießner. Scanrefer: 3d object localization in RGB-D scans using natural language. In *ECCV*, pages 202–221. Springer, 2020. 3

[6] Shizhe Chen, Pierre-Louis Guhur, Makarand Tapaswi, Cordelia Schmid, and Ivan Laptev. Language conditioned spatial relation reasoning for 3d object grounding. In *NeurIPS*, 2022. 3

[7] Shengheng Deng, Xun Xu, Chaozheng Wu, Ke Chen, and Kui Jia. 3d affordancenet: A benchmark for visual object affordance understanding. In *CVPR*, pages 1778–1787. Computer Vision Foundation / IEEE, 2021. 2, 3, 6

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *ACL*, pages 4171–4186. Association for Computational Linguistics, 2019. 7

[9] Kuan Fang, Te-Lin Wu, Daniel Yang, Silvio Savarese, and Joseph J. Lim. Demo2vec: Reasoning object affordances from online videos. In *CVPR*. Computer Vision Foundation / IEEE Computer Society, 2018. 2

[10] Yiran Geng, Boshi An, Haoran Geng, Yuanpei Chen, Yaodong Yang, and Hao Dong. Rlafford: End-to-end affordance learning for robotic manipulation. In *ICRA*, pages 5880–5886. IEEE, 2023. 1

[11] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: decoding-enhanced bert with disentangled attention. In *ICLR*. OpenReview.net, 2021. 7

[12] Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, Pierre Sermanet, Tomas Jackson, Noah Brown, Linda Luu, Sergey Levine, Karol Hausman, and Brian Ichter. Inner monologue: Embodied reasoning through planning with language models. In *CoRL*, pages 1769–1782. PMLR, 2022. 1

[13] Ayush Jain, Nikolaos Gkanatsios, Ishita Mediratta, and Katerina Fragkiadaki. Bottom up top down detection transformers for language grounding in images and point clouds. In *ECCV*, pages 417–433. Springer, 2022. 3

[14] Safoura Rezapour Lakani, Antonio Jose Rodríguez-Sánchez, and Justus H. Piater. Towards affordance detection for robot manipulation using affordance for parts and parts for affordance. pages 1155–1172, 2019. 1

[15] Muchen Li and Leonid Sigal. Referring transformer: A one-step approach to multi-task visual grounding. In *NeurIPS*, pages 19652–19664, 2021. 6, 1, 2

[16] Yicong Li, Xun Yang, Xindi Shang, and Tat-Seng Chua. Interventional video relation detection. In *ACM MM*, pages 4091–4099, 2021. 2

[17] Yicong Li, Xiang Wang, Junbin Xiao, and Tat-Seng Chua. Equivariant and invariant grounding for video question answering. In *ACM MM*, pages 4714–4722, 2022.

[18] Yicong Li, Xiang Wang, Junbin Xiao, Wei Ji, and Tat-Seng Chua. Invariant grounding for video question answering. In *CVPR*, pages 2928–2937, 2022.

[19] Yicong Li, Xiang Wang, Junbin Xiao, Wei Ji, and Tat-Seng Chua. Transformer-empowered invariant grounding for video question answering. *TPAMI*, 2023.

[20] Yicong Li, Junbin Xiao, Chun Feng, Xiang Wang, and Tat-Seng Chua. Discovering spatio-temporal rationales for video question answering. In *ICCV*, pages 13869–13878, 2023.

[21] Yicong Li, Xun Yang, An Zhang, Chun Feng, Xiang Wang, and Tat-Seng Chua. Redundancy-aware transformer for video question answering. In *ACM MM*, pages 3172–3180, 2023. 2

[22] Chang Liu, Henghui Ding, and Xudong Jiang. GRES: generalized referring expression segmentation. In *CVPR*, pages 23592–23601. IEEE, 2023. 6, 1, 2

[23] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019. 6, 7

[24] Liangsheng Lu, Wei Zhai, Hongchen Luo, Yu Kang, and Yang Cao. Phrase-based affordance detection via cyclic bilateral interaction. *IEEE Trans. Artif. Intell.*, 4(5):1186–1198, 2023. 2

[25] Hongchen Luo, Wei Zhai, Jing Zhang, Yang Cao, and Dacheng Tao. Learning affordance grounding from exocentric images. In *CVPR*, pages 2242–2251. IEEE, 2022. 2

[26] Junyu Luo, Jiahui Fu, Xianghao Kong, Chen Gao, Haibing Ren, Hao Shen, Huaxia Xia, and Si Liu. 3d-sps: Single-stage 3d visual grounding via referred point progressive selection. In *CVPR*, pages 16433–16442. IEEE, 2022. 3, 6, 1

[27] Xu Ma, Can Qin, Haoxuan You, Haoxi Ran, and Yun Fu. Rethinking network design and local geometry in point cloud: A simple residual MLP framework. In *ICLR*. OpenReview.net, 2022. 7

[28] Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. SQA3D: situated question answering in 3d scenes. In *ICLR*. OpenReview.net, 2023. 3

[29] Jinpeng Mi, Hongzhuo Liang, Nikolaos Katsakis, Song Tang, Qingdu Li, Changshui Zhang, and Jianwei Zhang. Intention-related natural language grounding via object affordance detection and intention semantic extraction. *Frontiers Neurorobotics*, 14:26, 2020. 2

[30] Kaichun Mo, Yuzhe Qin, Fanbo Xiang, Hao Su, and Leonidas J. Guibas. O2o-afford: Annotation-free large-scale object-object affordance learning. In *CoRL*, pages 1666–1677. PMLR, 2021. 2

[31] Bogdan Moldovan, Plinio Moreno, Martijn van Otterlo, José Santos-Victor, and Luc De Raedt. Learning relational affordance models for robots in multi-object manipulation tasks. In *2012 IEEE International Conference on Robotics and Automation*, pages 4373–4378, 2012. 1

[32] Tushar Nagarajan, Christoph Feichtenhofer, and Kristen Grauman. Grounded human-object interaction hotspots from video. In *ICCV*, pages 8687–8696. IEEE, 2019. 2

[33] OpenAI. Gpt-4 technical report, 2023. 2, 3

[34] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, pages 5099–5108, 2017. 4, 7

[35] Ishika Singh, Valts Blukis, Arsalan Mousavian, Ankit Goyal, Danfei Xu, Jonathan Tremblay, Dieter Fox, Jesse Thomason, and Animesh Garg. Progprompt: Generating situated robot task plans using large language models. In *ICRA*, pages 11523–11530. IEEE, 2023. 1

[36] Chan Hee Song, Jiaman Wu, Clayton Washington, Brian M. Sadler, Wei-Lun Chao, and Yu Su. Llm-planner: Few-shot grounded planning for embodied agents with large language models. *CoRR*, abs/2212.04088, 2022. 1

[37] NN Sriram, Tirth Maniar, Jayaganesh Kalyanasundaram, Vineet Gandhi, Brojeshwar Bhowmick, and K Madhava Krishna. Talk to the vehicle: Language conditioned autonomous navigation of self driving cars. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5284–5290. IEEE, 2019. 1

[38] Ilya O. Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. Mlp-mixer: An all-mlp architecture for vision. In *NeurIPS*, pages 24261–24272, 2021. 4

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017. 5

[40] Jiannan Wu, Yi Jiang, Peize Sun, Zehuan Yuan, and Ping Luo. Language as queries for referring video object segmentation. In *CVPR*, pages 4964–4974. IEEE, 2022. 5

[41] Junbin Xiao, Angela Yao, Yicong Li, and Tat Seng Chua. Can i trust your answer? visually grounded video question answering. *arXiv preprint arXiv:2309.01327*, 2023. 2

[42] Amber Xie, Youngwoon Lee, Pieter Abbeel, and Stephen James. Language-conditioned path planning. In *7th Annual Conference on Robot Learning*, 2023. 1

[43] Chao Xu, Yixin Chen, He Wang, Song-Chun Zhu, Yixin Zhu, and Siyuan Huang. Partafford: Part-level affordance discovery from 3d objects. *CoRR*, abs/2202.13519, 2022. 2

[44] Yuhang Yang, Wei Zhai, Hongchen Luo, Yang Cao, Jiebo Luo, and Zheng-Jun Zha. Grounding 3d object affordance from 2d interactions in images. *CoRR*, abs/2303.10437, 2023. 2, 3, 6, 7, 1

[45] Lichen Zhao, Daigang Cai, Lu Sheng, and Dong Xu. 3dvg-transformer: Relation modeling for visual grounding on point clouds. In *ICCV*, pages 2908–2917. IEEE, 2021. 3

[46] Xue Zhao, Yang Cao, and Yu Kang. Object affordance detection with relationship-aware network. *Neural Comput. Appl.*, 32(18):14321–14333, 2020. 2