Vision-Language Representations for Zero-Shot Robotic Perception

Satvik Sharma
Ken Goldberg, Ed.
Jitendra Malik, Ed.

Electrical Engineering and Computer Sciences
University of California, Berkeley

Technical Report No. UCB/EECS-2024-96
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2024/EECS-2024-96.html

May 10, 2024

Acknowledgement

First, I would like to thank Professor Ken Goldberg for his invaluable mentorship. I want to thank Adam Rashid, Kaushik Shivakumar, Chung Min Kim, Raven Huang, Justin Kerr, Lawrence Chen, and Ryan Hoque for their significant contributions to the work of this thesis. Lastly, I want to thank my family and friends.

Vision-Language Representations for Zero-Shot Robotic Perception

by

Satvik Sharma

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Master of Science

in

Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Ken Goldberg, Chair
Professor Jitendra Malik

Spring 2024

Vision-Language Representations for Zero-Shot Robotic Perception

Satvik Sharma

Research Project

Submitted to the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, in partial satisfaction of the requirements for the degree of **Master of Science, Plan II**.

Approval for the Report and Comprehensive Examination:

Committee:

DocuSigned by:

32B42AC2D1E3498...

Professor Ken Goldberg
Research Advisor

5/7/2024

(Date)

* * * * * * *

DocuSigned by:

Jitendra Malik

18D473A00244427...

Professor Jitendra Malik
Second Reader

5/8/2024

(Date)

Vision-Language Representations for Zero-Shot Robotic Perception

Abstract

Vision-Language Representations for Zero-Shot Robotic Perception

by

Satvik Sharma

Master of Science in Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor Ken Goldberg, Chair

As robotics systems enter the real world, the challenge of creating robotic perception systems that are robust to the real world still remains. The real world contains a visually and semantically diverse set of environments filled with an even more diverse set of objects. We can account for this diversity with large vision-language models (VLMs), which recently have shown promise in capturing semantics at the scale of the real world as they are pretrained on internet-scale data. We want to rely on these VLMs without any additional environment-specific data collection as it can be expensive for many robotic domains. Thus, we seek to integrate VLMs into the robotic perception pipeline to be used out-of-the-box or zero-shot for different tasks. We introduce two methods that utilize VLMs zero-shot for the robotic tasks of occluded object search and grasping, namely Semantic Mechanical Search (SMS) and Language Embedded Radiance Fields for Task-Oriented Grasping (LERF-TOGO) respectively. SMS utilizes LLMs in addition to VLMs to better semantically reason about visually occluded objects when searching. By embedding semantic understanding into the search process, SMS improves efficiency in locating objects across both simulated and real-world environments. On the other hand, LERF-TOGO creates a 3D vision-language field derived from VLMs to execute precise grasps of object parts based on natural language inputs. This method shows high accuracy and adaptability in physical trials, effectively grasping specified parts on a variety of objects. We conclude with the limitations of both of these works and possible future directions.

To my parents and incredible sister

Contents

List of Figures

List of Tables

Acknowledgments

As soon as I came to the Berkeley campus, I wanted to participate in undergraduate research. I am grateful to Professor Blonder for mentoring me in machine learning fundamentals and for giving me my first experience in research.

In my second year, I was able to join the AUTOLab, which became the most crucial part of my undergraduate experience. I am extremely thankful to Professor Goldberg for accepting me into the lab and giving me the opportunity to lead projects, travel to academic conferences, and collaborate with exceptional individuals at the forefront of robot learning. Professor Goldberg has given me invaluable guidance on how to conduct scientific research and effectively present it to an audience. In the AUTOLab, I transitioned from a true beginner to a researcher who could contribute to and eventually lead projects. I had amazing graduate and postdoc mentors at every stage of my research journey without whom I wouldn't have been able to succeed. I want to thank Daniel Brown, Ellen Novoseller, and Ashwin Balakrishna for teaching me the foundations of reinforcement and imitation learning. I am also thankful to Ryan Hoque and Lawrence Chen for our fleet robotics excursion. Additionally, I appreciate Raven Huang, Lawrence, and Ryan for our collaborative work on semantic search. My thanks also go to Justin Kerr and Chung Min Kim for introducing me to the 3D world. Lastly, I am grateful to Simeon Adebola for our shared experiences in gardening, and to Wisdom Agboh for our joint efforts in the multi-object grasping project. In addition to being exceptional academic mentors, more importantly, they were amazing friends who sought to support me and made me excited to come into the lab. In addition to collaborators within the lab, I would also like to thank Antonio Loquercio, Brian Ichter, and Anastasios Angelopoulos for your insightful conversations and valuable advice.

I would like to thank all the amazing collaborators I've had as my work would not have been possible without you: Adam Rashid, Kaushik Shivakumar, Vainavi Viswanath, Rishi Parikh, Karthik Dharmarajan, Kishore Srinivas, Mallika Parulekar, Gaurav Datta, Zaynah Javed, Jerry Zhu, Tianshuang Qiu, William Wong, Mark Presten, Mark Theis, Shrey Aeron, Ananth Rao, Sandeep Mukherjee, Tomson Qu, Anna Deza. Some of my best friends at Berkeley are from the lab and our friendship made my research experience all the more enjoyable.

I want to emphasize that the work I will present in my thesis is **not** entirely my own. I want to thank all my co-authors for writing different sections of the paper and multiple portions of the code. Specifically for SMS, I want to acknowledge Kaushik's work in running simulation experiments, ideating on how to formalize semantic distributions, and comparing our method to numerous baselines. I also want to acknowledge Raven for her work to set up the physical experiments, her essential ideas for the entire SMS framework, and her perseverance during the CoRL rebuttal. I want to thank Lawrence for his unparalleled literature reviews and his essential contributions to the framing of the SMS paper. For LERF-TOGO, I want to thank Adam for being an amazing partner and acknowledge his work in setting up the robot capture, developing the NeRF regularizations, comparing our method to baselines, and for the endless hours we sat through physical experiments. I also want to thank Justin and Chung Min for their pivotal ideas, for making immaculate figures, and for not only developing our grasp execution pipeline but also streamlining our experiments with Viser. I again emphasize the work I present would **not** have been possible without my amazing collaborators.

I want to thank all my friends at Berkeley who made these last five years truly unforgettable. I want to thank my friends from home for all the memories we made during holiday breaks and group calls. I also want to thank Meghana for always being a phone call away. Lastly, I want to thank my family without whom I would not be at Berkeley. I want to thank my parents for all their sacrifices, my aunts and uncles for their support, and my sister for her unwavering optimism.

Chapter 1

Introduction

The robotics dream is to have a general-purpose robot that can accomplish a list of tasks as well as a human would. However, a prerequisite for this type of robot is a robust perception system that can tackle the diversity of the real-world. Developing this perception system is a challenge. Even when isolated to a single robotic task, the robot has to interact with visually and semantically different environments and objects, many of which are uncommon and thus less likely to have been seen by the system (i.e. long-tail). Recent progress in large vision-language models (VLMs and LLMs) show promise for handling real-world diversity as they are pretrained on internet-scale data which empirically captures the diverse distribution of semantics and more importantly the distribution's tail (i.e. rare instances). A large body of prior work has shown that these models can provide good visual representations [1]–[5], ground language instructions [6]–[12], and serve as planners out of the box [13]–[18]. CLIP [19] is a commonly-used interface to associate vision and language, and many works [20]–[23] use it to build semantic scene representations and show improved performance on object query and navigation tasks. Fine-tuning these models with environment-specific data can be expensive, especially in real-world robotics domains, so the goal should be to use these models zero-shot. Thus, in this thesis, we build on existing work and tackle the question: how do we use VLMs zero-shot to create useful state representations for robotic tasks, specifically occluded object search and grasping?

In Chapter 2, we first go through existing work that has used natural language in robotics. Then, we delve into how natural language can be grounded within 3D state representations, especially for downstream robotic tasks. Lastly, we go through prior work on both robotic tasks: occluded object search (i.e. mechanical search) and task-oriented grasping.

In Chapter 3, we discuss Semantic Mechanical Search (SMS) [24], which uses VLMs zero-shot to create a semantic occupancy distribution that can be used to better search for occluded objects. Moving objects to find a fully-occluded target object, known as *mechanical search*, is a challenging problem in robotics. As objects are often organized semantically, we conjecture that semantic information about object relationships can facilitate mechanical search and reduce search time. VLMs and LLMs have shown promise in generalizing to uncommon objects and previously unseen real-world environments. SMS conducts scene understanding and generates a semantic occupancy distribution explicitly using LLMs. Compared to methods that rely on visual similarities offered by CLIP embeddings, SMS leverages the deep reasoning capabilities of LLMs. Unlike prior work that uses VLMs and LLMs as end-to-end planners, which may not integrate well with specialized geometric planners, SMS can serve as a plug-in semantic module for downstream manipulation or navigation policies. For mechanical search in closed-world settings such as shelves, we compare with a geometric-based planner and show that SMS improves mechanical search performance by 24% across the pharmacy, kitchen, and office domains in simulation and 47.1% in physical experiments. For open-world real environments, SMS can produce better semantic distributions compared to CLIP-based methods, with the potential to be integrated with more downstream search policies.

In Chapter 4, we discuss Language Embedded Radiance Fields for Zero-Shot Task-Oriented Grasping (LERF-TOGO)[25], which uses VLMs zero-shot to create a 3D representation used for task-oriented grasping. Grasping objects by a specific subpart is often crucial for safety and for executing downstream tasks. LERF-TOGO outputs a grasp distribution over an object given a natural language query. To accomplish this, we first construct a LERF of the scene, which distills CLIP embeddings into a multi-scale 3D language field queryable with text. However, LERF has no sense of object boundaries, so its relevancy outputs often return incomplete activations over an object which are insufficient for grasping. LERF-TOGO mitigates this lack of spatial grouping by extracting a 3D object mask via DINO features and then conditionally querying LERF on this mask to obtain a semantic distribution over the object to rank grasps from an off-the-shelf grasp planner. We evaluate LERF-TOGO's ability to grasp task-oriented object parts on 31 physical objects, and find it selects grasps on the correct part in 81% of trials and grasps successfully in 69%.

In Chapter 5, we conclude with the limitations of both algorithms, a discussion of future work, and parting words for my time at Berkeley.

Chapter 2

Related Works

2.1 Natural Language in Robotics

Grounding natural language instructions is a widely-studied problem in robot navigation [26]–[37], human-robot interaction [38], [39], and is increasingly studied in the manipulation literature [40]–[43]. While classical methods commonly rely on semantic parsing and factor graphs [31], [35], [38], end-to-end learning and leveraging pretrained models are now the most popular paradigms thanks to advances in deep learning and LLMs. Examples include language-conditioned imitation learning [8], [10]–[12], [44], [45], language-conditioned reinforcement learning [6], [46], [47], and online correction of robot policies through language feedback [48], [49]. In particular, pretrained image encoders and open-vocabulary object detectors have enabled generalization to novel object queries at test time [5], [16], [17]. LERF-TOGO and Semantic Mechanical Search (SMS) also take in novel object targets specified using natural language. However, in SMS, since the target objects are not visible in the scene, the robot instead needs to detect and localize other objects and reason about their relationships. This is particularly challenging in an open-world environment when the set of possible objects is unknown, making object detectors significantly less accurate. SMS shares similarity with HOLM [50], which uses an LLM to hallucinate nearby objects in partially observable scenes based on semantics computed from affinity scores. However, it relies on an object list and only considers camera adjustment actions in simulation. We relax this assumption of accessing object lists [20], [50], [51], propose a pipeline for generating object labels without access to any object lists, and generate semantic distributions for open-world environments.

Many studies have also used LLMs as a planner by letting them break down tasks through step-by-step reasoning [14]–[17], [52], [53] or directly write code [13], [18]. While these end-to-end planning paradigms benefit from the deep reasoning abilities of LLMs, it's not straightforward to incorporate additional non-language information and integration with domain-specific policies. The latter is particularly valuable when the task is more complex and a flexible generalist LLM can benefit from specialized searching and planning algorithms developed by the robotics community. In SMS, we propose decoupling semantic reasoning and geometric planning; rather than directly output primitive instructions from image observations, SMS uses LLMs' semantic reasoning from its feature space into a semantic distribution that specialized planning and manipulation policies can use.

2.2 Grounding Language with 3D representations

With the advent of large pretrained language and vision-language models, several works have explored building 3D map representations to guide robot navigation. VL-Maps [54] and OpenScene [51] build a 3D language embedding from pretrained open-vocab detectors [55], [56], which can be used to navigate to target queries. CLIP-Fields [57], ConceptFusion [58], and NLMaps-SayCan [59] take more of a region-proposal based approach, querying CLIP on the outputs of some region proposal methods and fusing them into 3D point clouds for downstream navigation tasks. Region-based zero-shot methods retain more language understanding than fine-tuned features but run the risk of missing objects by insufficiently masking input images. Semantic Abstraction [60] avoids this by extracting relevancy from vision-language models using [61] and uses these for composing multiple language queries with spatial relationships. Language has also been studied in the context of robot manipulation. [43] use the MAttNet [62] vision-language model for object rearrangement, and CLIPort [63] uses language understanding from CLIP to train a language-conditioned pick and place module from demonstrations. PerAct [64] uses language-conditioned demonstrations with a 3D scene transformer to learn diverse tasks, MOO [5] uses the outputs from OWL-ViT to condition a manipulation policy for grasping objects, and large-scale demonstration datasets like RT-1 [45] train on massive language-conditioned demonstration trajectories. In contrast to many other language-conditioned approaches, LERF-TOGO uses internet-scale vision models purely zero-shot and does not require fine-tuning on demonstrations or robot exploration.

2.3 Mechanical Search

Mechanical search [65], [66] refers to a broad class of robotics problems on searching for occluded and out-of-view objects via manipulation and navigation. In the former case, bin [65] and shelf environments [67]–[71] are widely studied, where intelligent estimation and manipulation planning based on possible locations of the hidden target object significantly affects the search efficiency. Many prior work uses only geometric priors [65], [72]–[75]. A number of authors have also explored using semantic context object information [76]. Kollar and Roy [77] obtain co-occurrence statistics from web-based ontologies and Wong et al. [78] extend the approaches to occluded target objects. [66] propose a hierarchical model to integrate semantic and geometric information and learn in simulation. However, they manually craft semantic categories, which are also sparse and can not accurately and scalably reflect real-world distributions. Instead, in SMS, we harness large pretrained models to extract open-vocabulary semantic information zero-shot.

There are many types of navigation tasks, such as point goals [79]–[81], image goals [82], [83], and object goals [84], [85]. Finding out-of-view objects is an object goal navigation task, and the problem is also known as active visual search [86], [87]. Classical geometry-based methods typically first build a map [88], [89] and then perform planning [90], [91]. Learning-based methods typically use reinforcement learning trained in simulation [80], [84], [92]–[98], through YouTube videos [85], or by querying the Internet [99] to learn semantics and efficient exploration strategies. Recently, many works have explored using LLMs and VLMs out-of-the-box for semantic scene understanding [100] and zero-shot object navigation, which this work belongs to. The most common strategy is to use CLIP features [19] obtained from pretrained open-vocab detectors [55], [56] as in VL-Maps [20] and OpenScene [51] or from region proposals models as in CLIP-Fields [57], ConceptFusion [58], and NLMaps-SayCan [59] and fuse them into 3D point clouds or implicit representations [101]. The constructed representations can then be used for open-vocabulary target queries to locate the object and perform navigation. [21] propose a family of methods to adapt CLIP and open-vocabulary models to localize target objects. Through a systematic comparison, they find OWL-ViT detector [102] works best, followed by patchifying images to obtain separate CLIP embeddings and compute similarity with text embeddings. In SMS, instead of using the similarity of CLIP embeddings to construct relevancy maps [103], we use the LLM feature space to explicitly reason about the object's semantic relationships.

2.4 Task-Oriented Grasping

Task oriented grasping studies how to grasp objects by specific parts based on a use case. It has been studied by probabilistically modeling human grasps [104], extracting geometric features from labeled object parts [105], training on part-affordance datasets in simulation [106], or transferring category-specific part grasps to new instances [107]. Recent works [108], [109] train object-part grasp networks by leveraging object part and manipulation affordance datasets for a range of household objects. [110] use videos of humans interacting with objects to guide grasps towards the same part. Decomposing objects into parts has also long been studied as a co-segmentation task in vision [111], [112]. Recent approaches use pretrained vision features to discover common parts within sets of objects [113]. This technique has been applied at scale to segment parts of objects based on a canonical object [114] or detect object affordances from example images of human usage [115]. Though effective, it assumes access to a canonical image of each object and pre-existing part labels or demonstrations, which are restrictive in real-world applications. In contrast, LERF-TOGO uses off-the-shelf vision-language models trained at scale, so it captures long-tail object and parts more easily without the use of affordance datasets.

Chapter 3

Semantic Mechanical Search

3.1 Background

Mechanical search, where a robot manipulates objects and/or navigates to find a fully occluded target object [65], [66], is a challenging robotics problem. Prior work has shown success in revealing the desired object by manipulating the occluding objects [72]–[74], obtaining new observations after rotating the camera [50], or navigating to new locations [20], [21]. However, generalization to unseen environments remains challenging due to the numerous long-tail objects present in the real world.

Environments are often organized semantically, for example, toothpaste is often stored in a home bathroom near toothbrushes. Here, we explore how LLMs can provide such semantic relationships to facilitate mechanical search. Prior work has shown that VLMs and LLMs can capture these semantic relationships relatively well, commonly using CLIP to interface between language and image observations. These existing approaches that use CLIP to create state representations, while informative, the dot product of CLIP text and image embeddings lacks deep reasoning capabilities and sometimes behaves as a bag-of-words [101]. As such, CLIP is most useful for localizing objects that are already visible somewhere in the scene or map [5], [21], a property that many open-vocabulary object detectors build on [116]–[118]. When the target object is fully occluded, CLIP alone may not provide enough clues about potential target object locations.

LLMs demonstrate advanced reasoning and planning capabilities [119]. Many prior works [17], [20], [52] use VLMs and LLMs as end-to-end planners for both perception and planning. While such paradigms benefit from the semantic reasoning abilities of LLMs, they do not handle additional information that cannot be easily expressed through language and may not integrate well with other domain-specific policies. For example, for mechanical search on shelves, the geometric properties of objects provide valuable cues for identifying potential target object positions, and various algorithms have been proposed for handling uncertainty and planning ahead [72]–[74]. Likewise, for object navigation, prior research has explored exploration and navigation strategies that are independent of semantic understanding [120]–[123]. As such, decoupling semantic reasoning and geometric planning may allow flexible integration with task-specific modules for various downstream settings.

We propose SMS, which generates an explicit intermediate representation, a *semantic occupancy distribution*, as a plug-in semantic module for existing mechanical search algorithms. This distinguishes it from prior work where VLMs and LLMs serve as end-to-end planners doing both semantic reasoning and action planning. With the goal of adding semantic reasoning to existing search policies, we study two questions: (1) Can a semantic distribution facilitate mechanical search? (2) What is the best way to generate this semantic distribution? For the second question, we hypothesize that translating image features into language features (with VLMs) first and then extracting semantic distributions from only language features (with LLMs) can outperform VLM-only methods that most current works use. We show that, rather than burdening VLMs (e.g. CLIP) with both object detection and reasoning, decoupling these two tasks leads to better results as the LLM language feature space is better at capturing semantic relations. For the first question, we show SMS can be easily integrated with a geometric shelf searching algorithm [74] to improve performance for closed-world environments such as pharmacy shelves with known object lists. In closed-world settings where object lists are available, SMS uses an open vocabulary object detection model [124] refined with Optical Character Recognition (OCR) to identify objects. In open-world settings where object lists are unavailable, SMS combines segmentation [125] and image captioning [126] to generate object mask descriptions.

3.2 Preliminaries

LAX-RAY

LAX-RAY[72] is a mechanical search policy for shelf environments. LAX-RAY utilizes geometric information by considering object geometries and camera perspective (e.g., tall target objects cannot be occluded by short objects and objects in the center of an image occlude more areas) to facilitate the search. It consists of a perception module and a greedy action selection module. The perception module takes the depth observation and predicts the geometric/spatial occupancy distribution to encode the geometric information. LAX-RAY learns this module on a simulation dataset, with the ground-truth occupancy distribution calculated using Minkowski sum. A greedy action selection module called Distribution Area Reduction (DAR) selects robot actions to greedily reduce the overlap between objects and the distribution. Another search policy, Distribution Entropy Reduction (DER), selects the action that would reduce the entropy of the distribution the most after taking the action.

Occupancy Distribution

An occupancy distribution indicates the probability of each pixel in the image containing the target object's amodal segmentation mask [65]. Prior works [65], [72]–[74] have utilized geometric information to generate spatial occupancy distributions by considering object geometries and camera perspective (e.g., tall target objects cannot be occluded by short objects and objects in the center of an image occlude more areas) to facilitate the search. [72] propose the LAX-RAY system, which uses a neural network to predict the spatial occupancy distribution. SMS generates the occupancy distribution using semantic information, which can then be combined with the LAX-RAY spatial distribution for downstream search as now it can be used in larger environments where the geometric information is more subtle.

3.3 Problem Statement

We consider a partially observable environment that contains a target $\mathcal{O}_T$ and N other objects $\{\mathcal{O}_1, ..., \mathcal{O}_N\}$. We assume the scenes in the environment are semantically organized, meaning that the starting state of the environment is sampled proportionally to their approximate likelihood of occurrence in the real world. With this assumption of semantically organized scenes, the target object location probability is proportional to object pair affinities. States $s_t \in \mathcal{S}$ consist of the full geometries, poses, and names of the objects in the scene at timestep t, and observations $y_t \in \mathcal{Y} = \mathcal{R}^{H \times W \times 3}$ are RGB images from a robot-mounted RGB camera at timestep t. Given the name of the target object and the observation y_t, the goal is to generate a useful dense occupancy distribution that encodes semantic affinities (with respect to the target object).

3.4 Algorithm

We propose SMS, a framework using VLMs and LLMs to create a dense semantic distribution between a scene and the target object to be used for downstream tasks. Fig. 3.1 visualizes the pipeline. SMS first uses VLMs to perform scene understanding by creating mask-label pairs to densely describe all image portions. It then uses an LLM to generate affinity scores between the labels and the target object. We spatially ground these affinities using the labels' corresponding masks. In this way, we densely represent the affinities between a target object and all parts of a scene using an LLM. SMS can be applied to two common situations: 1) a closed world where all objects in the scene are a subset of a known list and 2) an open world where some objects in the scene are previously unseen.

Scene Understanding

The goal of scene understanding is to generate mask-label pairs that characterize the scene.

Object Detection + OCR

When an object list is available, we use an open vocabulary object detection model, specifically ViLD [124], to obtain object segmentation masks and labels from an RGB image. We

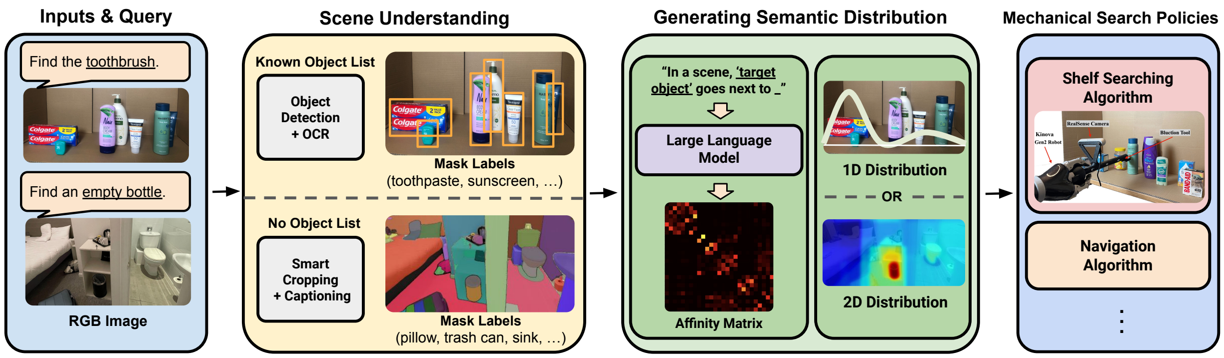

Figure 3.1: **Overview of (SMS).** SMS accepts as input a scene image and a desired target object. It applies an object detection, or segmentation algorithm combined with captioning as necessary when object lists are unavailable. SMS then uses an LLM to compute affinities between detected objects to the target object, and it uses these affinities to output a semantic occupancy distribution which can be used for downstream mechanical search policies.

also find using Keras Optical Character Recognition (OCR) [127] can improve the quality of the ViLD object detection.

Crop Generation + Image Captioning

When an object list is not available, many open vocabulary detectors such as ViLD cannot be used. We instead create image crops and use a VLM for crop captioning, specifically BLIP-2 [126], to convert object crops to their text descriptions. We ask for the dominant objects in each crop for less noisy captions. We generate crops that are both object-centric (using Segment-Anything (SAM) [125]) for better object boundaries in the semantic distribution and general multiscale, overlapping crops that help encode large-scale semantic information.

Creating the Semantic Distribution

We consider two ways to use a language model to generate affinity scores for the semantic distribution. **(1) SMS-LLM:** We iterate over all the mask-label pairs and query the LLM with a specific prompt: "*I see the following in a room:* {*label*}*. This is likely to be the closest object to* {*target object*}". This prompt directly represents the probability of the target object given we see the label. Since object labels are contained within the prompt, we do not need to normalize to account for the prior. A similar prompt with the label and the target object switched would also provide affinity scores between objects but would then have to be normalized to account for that object's prior. The affinity score for the target object with each label is the completion probability for the tokens that represent the target object. We generate a semantic distribution from these affinity scores and detected objects. The semantic distribution models the probability of the target object occupying each location, which we approximate to be proportional to the affinity score between the target and the object closest to that location. To account for noise, we apply spatial smoothing using a Gaussian kernel with std σ. **(2) SMS-E:** An alternative method we explore is to use a language embedding model (e.g. OpenAI Embedding Model [128]) to get embeddings for all labels and the target object, and obtain an affinity score between each label and the target object through the dot product between these vectors.

When there are no object lists, the Crop Generation + Image Captioning pipeline described in Section 3.4 can contain many incorrect or hallucinated labels, making the distribution noisy. To mitigate this, we use CLIP to verify the captions and not for any semantic reasoning. Specifically, we compute the CLIP dot products between the image crops and the generated labels and weight the affinity scores by these relevance scores. To produce the final semantic distribution, each pixel receives the average of the weighted affinity scores of all the masks it belongs to. We find that averaging across multiple overlapping masks also helps reduce noises in the absence of object lists.

Combining with Mechanical Search Policies

We consider both closed-world and open-world environments.

Closed-World Environments

We consider semantically organized shelves with objects from a known list. For mechanical search on shelves, the robot needs to manipulate objects in the shelves to reveal the occluded target object using pushing and pick-and-place actions. The goal is to minimize the number of actions taken to reveal the target object. Thus, we use SMS as a plug-in semantic module for an existing search algorithm, LAX-RAY [72], by multiplying the semantic occupancy distribution with a learned spatial distribution that LAX-RAY generates based on geometry.

We then use the DAR policy [72] to perform mechanical search. Since the search in cluttered environments requires manipulating other objects, once the search begins, the shelf may become semantically disorganized. As such, at each step in a rollout, SMS computes the semantic distribution using the object locations where each object was first discovered.

Open-World Environments

We consider large room spaces, with semantic diversity (rooms of an office, home, aisles in a grocery store, etc.). We do not perform any manipulation in this setting and explore a downstream heuristic navigation policy that terminates when the object is within view. Given a starting position, the policy moves a fixed distance towards the highest affinity region in the image. Afterward, it takes four new images by rotating in place. We first select the desired view direction amongst the four by choosing the one that has the highest 90-percentile affinity score to ensure we are more robust to outlier affinities that may result from not having an object list. Then, after selecting the view, we again select the highest affinity point and move to that location.

3.5 Experiments

We investigate two questions: with a given downstream search policy, (1) can a semantic distribution improve search performance? and (2) what is the best way to generate a semantic distribution?

Evaluation of Semantic Distributions Quality

We investigate the second question first to obtain a semantic distribution for the downstream policy. We evaluate semantic distribution generation both in closed-world and open-world environments. In close-world environments, we evaluate the affinity matrix quality where the semantic distribution is generated, for the given object list. In open-world environments, we evaluate the semantic distribution quality on a dataset of real-life scenes.

Closed-World Environments

We discuss the experimental setup and the results.

Experimental Setup: With an object list, the semantic distributions are directly generated from semantic affinity matrices, with rows and columns as objects from the list and the entries representing the affinities scores between objects calculated in Section 3.4. We use an object list of 27 objects in the pharmacy domain (included in [24]). We directly compare the affinity matrix quality. We approximate a ground truth affinity matrix with Google Taxonomy. The Google taxonomy provides semantic information for evaluation purposes to avoid human bias. Since it has limited categories, it cannot be used directly for objects that do not appear in the taxonomy.

Results: We compare the quality of the affinity matrix for the given object list generated by SMS and a CLIP-based baseline proposed by CoW [21], which uses the dot products of CLIP text embedding as the affinity scores. We compute the reduction of Jensen-Shannon Distance (JSD) [129] between the generated affinity matrix and the ground truth affinity matrix compared to the JSD between a uniform matrix and the ground truth. This quantifies the benefit SMS provides over a uniform distribution. From Table 3.1, we see that SMS significantly outperforms the CLIP-based method, while the SMS-LLM variant slightly outperforms the SMS-E. This suggests the reasoning capability of LLM models is more valuable for capturing semantics than CLIP embeddings.

Open-World Environments

Now, we discuss the experimental setup and the results for the open-world environments.

Experimental Setup: For the open-world environment, we evaluate the semantic distribution generation on a static image dataset consisting of 30 real scenes taken from 4

Metric	CLIP	SMS-E	SMS-LLM
Δ% JSD ↑	20.0%	33.8%	44.6%

Table 3.1: **Closed-world semantics evaluation.** Percentage improvement of the generated semantic distributions in the pharmacy domain compared to a uniform prior, measured based on the Jensen-Shannon Distance (JSD) from the ground truth distribution.

Method	OWL-ViT	CLIP	SMS-LLM	SMS-E
IoU	0.138 ± 0.031	0.221 ± 0.034	0.345 ± 0.039	**0.391** ± 0.039

Table 3.2: **Open-world semantics evaluation.** IoU results for different methods. The object detector-based method, OWL-ViT, performs poorly because even though the target objects are semantically related, many have very little visual similarity. CLIP performs worse than SMS because SMS is getting semantic similarity in a language-only latent space which can capture more nuance than the visual-language embedding space.

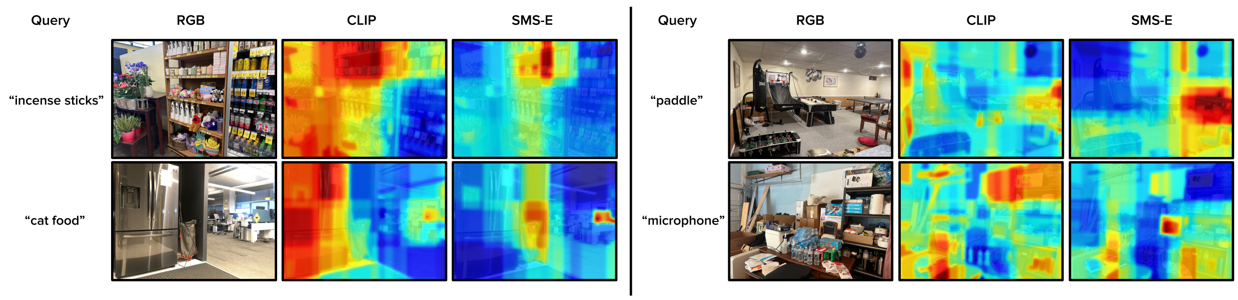

Figure 3.2: **Generating semantic distributions in open-world environments.** Four examples from the evaluation dataset with the 2D probability distributions generated for SMS-E and CLIP. These heatmaps are red for high-probability regions of finding the target object and blue for low probability. **Top Left:** An example of a grocery store, where the target object is "incense sticks." CLIP highlights both near the candles and near the flowers as they are somewhat visually similar to sticks, while SMS-E only highlights the candles. **Bottom Left:** An example of an office kitchen, where the target object is "cat food." CLIP gets distracted by the refrigerator and only slightly highlights the cat sign. **Top Right:** An example of a house, where the target object is "paddle." CLIP incorrectly highlights the wooden panels along the walls, while SMS-E highlights the ping pong table. **Bottom Right:** For the target word "microphone," SMS-E highlights the box with the speaker but CLIP struggles as the objects are not visually similar.

houses, 4 office buildings, and 3 local grocery stores. We sampled 90 objects across the three domains and chose those scenes based on our accessibility to those places. In all scenes, the objects' numbers and placements are set by their management. All scenes and the target object list are in included in [24]. Since these scenes are large, we are interested in quantifying the accuracy of the semantic distribution along both the x- and y-axes. We annotate the ground truth search area based on the real scene and use Intersection over Union (IoU) to quantitatively evaluate the accuracy of each method.

Results: We evaluate the following VLM-only baselines: CLIP and OWL-ViT, the two best-performing methods found by [21]. For CLIP, it uses the same crop-label pairs as SMS to generate a semantic distribution as described in Section 3.5 but with further augmentations (jittering and horizontal flipping) on those crops for better performance [100]. We threshold this distribution and create a mask to calculate IoU with the ground truth. OWL-ViT gives bounding boxes for its labels and we directly use them to calculate the IoU. We find that OWL-ViT performs better if the best bounding box is selected rather than weighting all bounding boxes by their score and thresholding that distribution. Table 3.2 shows the results. SMS generates semantic distributions within 35 to 45 seconds. We see that SMS outperforms the VLM-only methods including both CLIP and OWL-ViT. We hypothesize that this is because CLIP focuses more on the visual appearance of the objects rather than semantic relations. This would be less of a problem for searching visible objects but is not ideal for searching objects that are outside the field of view or occluded. For example, CLIP would associate incense sticks with sticks used for gardening while LLMs would associate the incense sticks with the candles. In addition, CLIP has a "bag of words" behaviour [101], causing it to incorrectly relate "cat food" with a fridge instead of a cat sign. In contrast, LLMs have better semantic reasoning as shown in Figure 3.2, where "cat food" highlights the cat sign as the highest region but also highlights the gray bag because cat food could be occluded inside of a bag. Since LLMs are trained on large corpora of human language, we hypothesize that they effectively encode the semantics of both common and rare objects and are also capable of semantic reasoning (e.g. cat food can be inside the bag) beyond just creating class categories and thus are better suited for searching fully-occluded objects. SMS-E slightly outperforms SMS-LLM as they are both bottlenecked by the quality of labels from BLIP-2.

We also conduct an ablation study for each module of SMS on semantic distribution quality with results in Table 3.3, indicating the effectiveness of cropping with SAM and CLIP weighting, and the impact of image captioning model choice. As mentioned previously that image captioning can be noisy, we use CLIP to verify the captions. We refer this as CLIP weighting. Without this, the performance drops by 21%. When we use BLIP-IC instead of BLIP-2 for image captioning, the performance drops by 21%. Finally, without cropping using SAM to get object-centered crops, the performance drops by 27%.

Ablations	**w/o CLIP Weighting**	**BLIP-IC**	**w/o SAM**	**SMS-E**
IoU	0.307± 0.038	0.310 ± 0.043	0.286± 0.038	**0.391**± 0.039

Table 3.3: IoU results for ablated SMS-E. **w/o CLIP Weighting** doesn't using CLIP to refine the generated captions as described in Section 3.4. **BLIP-IC** use BLIP-IC to get the descriptions for each crop instead of BLIP-2. **w/o SAM** doesn't use crops given by SAM and crops generated by multi-scale sliding windows are used.

BLIP-IC is the large image captioning model of BLIP.

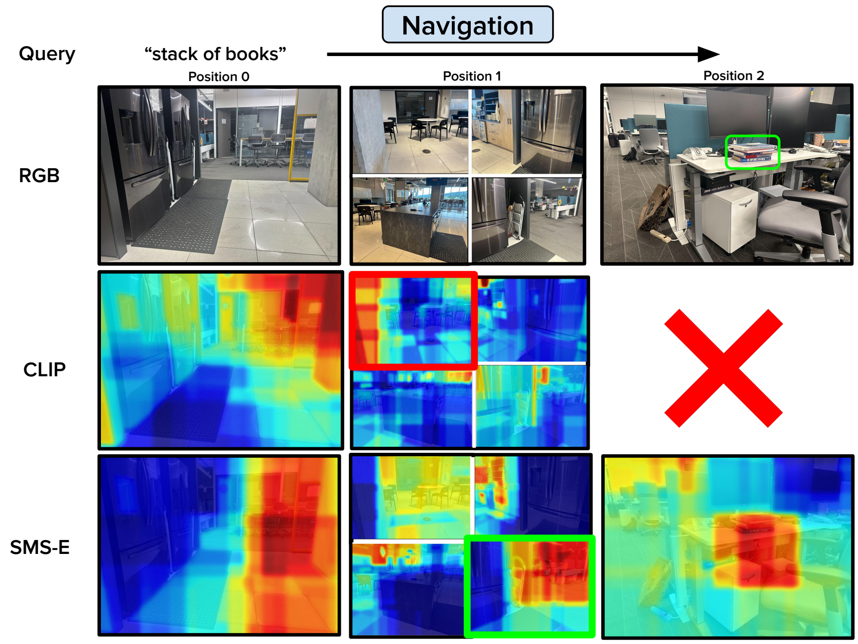

Figure 3.3: **Object navigation experiment in BAIR Office Kitchen.** A short horizon navigation example where we start at position 0 and end at position 2. SMS is able to correctly maneuver to the stack of books while CLIP fails because its bag-of-words nature is susceptible to incorrectly assigning high probability to the pillars in the scene as they are semantically related to "stack."

We also conduct a preliminary navigation experiment where a mobile robot follows the downstream navigation policy described in Section 3.4 and selects the physical location to move to based on the semantic distribution from CLIP and SMS-E. As shown in Figure 3.3, CLIP makes an incorrect turn (at position 1 it continues in the direction of the view with the red box) because of its bag-of-words behavior and attributes "stack of books" to having higher semantic similarity to concrete pillars in the scene rather than the area with office desks and chairs. SMS-E continues towards the office (green box in position 1) and finds a stack of books on a desk successfully.

Semantic Distribution Effect on Mechanical Search Performance

Given a semantic distribution, we investigate the first question by conducting simulation and real experiments in close-world environments to evaluate search performance improvement brought by the semantic distribution. We combine the semantic distribution with an existing mechanical search policy LAX-RAY as in Section 3.4.

Experimental Setup

For simulation, we consider a pharmacy, a kitchen, and an office domain. In addition to the 27 objects for the pharmacy domain, we consider 24 and 40 representative objects for the kitchen and office domains from the Google Product Taxonomy [130]. The taxonomy defines a tree where each category is a node and each object name is a leaf node. To create a scene with N objects in a given domain, we begin by uniformly sampling N objects without replacement from the total objects available in that domain. We then generate scenes in a top-down recursive manner using the taxonomy tree. At the root, we start with the whole shelf available to us. At each node, we split the shelf in half either horizontally or vertically with 50% probability each and recursively continue scene generation in these sub-shelves. If a node has more than 8 descendants, however, we always split the scene horizontally to avoid overcrowding resulting from the aspect ratio of the shelf. At each level of recursion, we accumulate random noise to the eventual placement of each object in the current branch, uniformly sampled from -2 cm to 2 cm. At the last non-leaf node, we place all leaves in random positions within the current level's sub-shelf. We resolve collisions by iteratively moving objects along the displacement vector between colliding objects and discard scenes where such a procedure takes longer than 1 second to run. We also discard scenes where there is no potential target object that is invisible from the camera's perspective at the start of the rollout. We reiterate that the taxonomy is *independent* of the language models used to generate affinities. The LLMs are applicable beyond manual semantic categorizations like the Google Taxonomy, but we use this resource for evaluation purposes. The scenes for all simulation, physical, and object detection experiments are generated by this procedure.

We use approximate sizes of these items to generate collision-free scenes. In simulation, we also scale these objects down in order to be able to run experiments on the same-sized shelf, which has an effect similar to running experiments in a larger shelf where more items could originally fit. The scaling factors for the pharmacy and kitchen domains are 0.7, but 0.4 in the office domain due to overall larger objects unable to easily fit and move within a small shelf. The simulation and real experiments take place within a $0.8\,\text{m} \times 0.35\,\text{m} \times 0.57\,\text{m}$ shelf environment.

Simulation Experiments

We run extensive experiments using the First Order Shelf Simulator (FOSS) from [74]. In simulation experiments, we assume perfect object detection but consider geometry for occlusion. For each domain, we generate semantically organized scenes (details in Section 3.5) with various numbers of objects N=12, 15, 18, 21 with 200 scenes for each. Termination occurs when the target object becomes visible or reaches maximum action number $2N$.

For each scene, we evaluate whether SMS improves the performance of LAX-RAY [74], which only uses geometric models. We consider both SMS-E and SMS-LLM for augmenting the geometric distribution from LAX-RAY. We report two metrics: **Success rates:** The ratio of trials where the target object is found within the maximum action limit to the total number of trials. **Number of actions:** The mean and standard error of the number of actions required to reveal the target object.

	Pharmacy Domain			Kitchen Domain			Office Domain		
	Successes	**# Actions**	**Δ%**	**Successes**	**# Actions**	**Δ%**	**Successes**	**# Actions**	**Δ%**
LAX-RAY	576/741	5.56 ± 0.20	N/A	703/770	3.32 ± 0.14	N/A	575/753	4.14 ± 0.19	N/A
SMS-E	591/741	4.18 ± 0.17	24.8	**725/770**	2.43 ± 0.10	26.8	580/753	4.10 ± 0.18	0.9
SMS-LLM	**606/741**	$\mathbf{3.76 \pm 0.14}$	**32.4**	710/770	$\mathbf{2.42 \pm 0.10}$	**27.1**	**598/753**	$\mathbf{3.63 \pm 0.16}$	**12.3**

Table 3.4: Simulation experiment results for three domains averaged over 12, 15, 18, 21 number of objects, also reported with Δ%, the percentage reduction in the number of actions compared to LAX-RAY.

Table 3.5: Simulation Experiment Results.

	Pharmacy Domain							
	12 objects		**15 objects**		**18 objects**		**21 objects**	
	Successes	**# Actions**	**Successes**	**# Actions**	**Successes**	**# Actions**	**Successes**	**# Actions**
LAX-RAY	168/190	4.06 ± 0.23	160/186	5.17 ± 0.28	144/188	5.78 ± 0.44	104/177	8.24 ± 0.67
SMS-E	**176/190**	2.90 ± 0.18	159/186	3.77 ± 0.26	146/188	5.05 ± 0.42	110/177	5.69 ± 0.54
SMS-LLM	**176/190**	$\mathbf{2.66 \pm 0.14}$	**162/186**	$\mathbf{3.26 \pm 0.19}$	**150/188**	$\mathbf{4.25 \pm 0.34}$	**118/177**	$\mathbf{5.47 \pm 0.43}$

	Kitchen Domain							
	12 objects		**15 objects**		**18 objects**		**21 objects**	
	Successes	**# Actions**	**Successes**	**# Actions**	**Successes**	**# Actions**	**Successes**	**# Actions**
LAX-RAY	185/192	2.15 ± 0.14	182/194	2.97 ± 0.23	177/193	3.99 ± 0.29	159/191	4.36 ± 0.38
SMS-E	**186/192**	$\mathbf{1.56 \pm 0.08}$	**188/194**	2.15 ± 0.15	**184/193**	3.00 ± 0.27	**167/191**	$\mathbf{3.07 \pm 0.25}$
SMS-LLM	184/192	1.60 ± 0.10	184/194	$\mathbf{2.04 \pm 0.13}$	179/193	$\mathbf{2.97 \pm 0.26}$	163/191	3.17 ± 0.28

	Office Domain							
	12 objects		**15 objects**		**18 objects**		**21 objects**	
	Successes	**# Actions**	**Successes**	**# Actions**	**Successes**	**# Actions**	**Successes**	**# Actions**
LAX-RAY	172/194	2.60 ± 0.18	152/188	4.15 ± 0.38	136/190	4.64 ± 0.37	115/181	5.86 ± 0.56
SMS-E	**173/194**	3.01 ± 0.22	152/188	3.80 ± 0.31	140/190	4.78 ± 0.44	115/181	$\mathbf{5.33 \pm 0.50}$
SMS-LLM	172/194	$\mathbf{2.33 \pm 0.13}$	**161/188**	$\mathbf{3.50 \pm 0.31}$	**142/190**	$\mathbf{3.75 \pm 0.32}$	**123/181**	5.50 ± 0.49

We report results for all numbers of objects N in Table 3.5 and the results averaged across all values of N in Table 3.4. In all domains, SMS-LLM and SMS-E improve LAX-RAY performances with higher success rates and fewer search actions. In the pharmacy and office domain, SMS-LLM outperforms SMS-E, while in the kitchen domain, they perform comparably. For the office experiments, the performance improvement is relatively small. We hypothesize that this is due to a majority of the office environment consisting of generic office supplies that do not have a clear semantic categorization, making semantic prior less effective. Overall, the results suggest that SMS-LLM can serve as a semantic plug-in module and improve LAX-RAY performance in semantically arranged environments by 32.4%, 27.1%, and 12.3% in the pharmacy, kitchen, and office domains respectively while improving success rates. SMS-LLM outperforms SMS-E, indicating the quality of the affinity matrix is directly correlated with the task performance.

In addition, we show SMS is effective on different downstream policies by using SMS as the plug-in module for Distribution Entropy Reduction (DER) from [74]. We multiply the semantic distribution with the geometric distribution as the input to DER. DER selects the action minimizing the distribution entropy after taking the action. We use the same setup and scenes as before. We report the results for 100 scenes with 12 objects in Pharmacy domain in Table 3.6.

Policy	12 objects		15 objects		18 objects		21 objects	
	Success	# Actions	Success	# Actions	Success	# Actions	Success	# Actions
LAX-RAY (DER)	84%	5.79±0.38	74%	7.69±0.54	62%	8.08±0.64	42%	9.52±0.72
SMS-LLM	90%	4.42 ± 0.39	81%	5.06±0.43	71%	7.11±0.60	45%	6.87±0.67

Table 3.6: Simulation experiments results of SMS-LLM with DER for the Pharmacy domain. We ablate the downstream policy and see that SMS-LLM outperforms LAX-RAY with DER. We report the number of rollouts that were successful and the mean actions to retrieve the occluded object and the standard error.

Method	**No noise**	**10% Noise**	**50% Noise**	**90% Noise**	**LAX-RAY**
# of Actions	3.81 ± 0.31	4.20 ± 0.38	4.44 ± 0.41	4.83 ± 0.47	5.12± 0.43

Table 3.7: Experiment to determine the impact of object detection noise on task performance (# of actions). For SMS-LLM, we randomly perturb the object detection (i.e. randomly select a label from the object list) with probability P. We do 400 rollouts over the categories of 12, 15, 18, 21 objects in the scene for the pharmacy domain. We report the average number of actions taken to reveal the target object and standard error. We see the general trend as object detection noise increases the task performance decreases.

Lastly, we also show a strong positive correlation between object detection accuracy and task performance with Table 3.7, indicating the benefits of SMS using OCR. We randomly change the object labels with a probability P. The results are shown in Table 3.7, where $P = 0.1, 0.5, 0.9$. The number of actions needed to find the occluded object increases as P increases. This is because random perturbations can cause the semantic distribution to approach a uniform distribution thus not modifying the existing action of the downstream policy. Therefore, Table 3.7 indicates there is also a strong positive correlation between object detection accuracy and task performance.

Physical Experiments

We conduct experiments on a physical pharmacy shelf. We use the Kinova Gen2 robot with a 3D-printed blade and suction tool [72]. An Intel RealSense depth camera mounted on the tool provides RGBD observations. We use 3 scenes each of $N = 7, 8, 9,$ and 10 objects for a total of 12 scenes and a threshold visibility of 50% for determining success.

Method	# Actions	Δ%	Method	# Actions	Δ%
LAX-RAY	4.25 ± 0.64	N/A	**SMS-LLM**	$\mathbf{2.25 \pm 0.46}$	**47.1**

Table 3.8: Physical experiment results (12 trials each). We report the average number of actions taken to reveal the target object as well as the percentage reduction in the number of actions over the spatial neural network.

As simulation results from Table 3.4 suggest SMS-LLM outperforms SMS-E, we evaluate SMS-LLM in physical experiments. An identical set of 12 semantically arranged scenes (starting configurations) is used for each method. Results are shown in Table 3.8. We observe that SMS significantly accelerates mechanical search on shelves, reducing the average number of actions by 47.1%. In physical experiments, the noises in the depth images result in worse spatial distribution than in simulation, making the semantic distribution more critical in identifying where a target object may lie.

Figure 3.4: Here are 3 example scenes from the physical mechanical search experiments in the constrained environment setting.

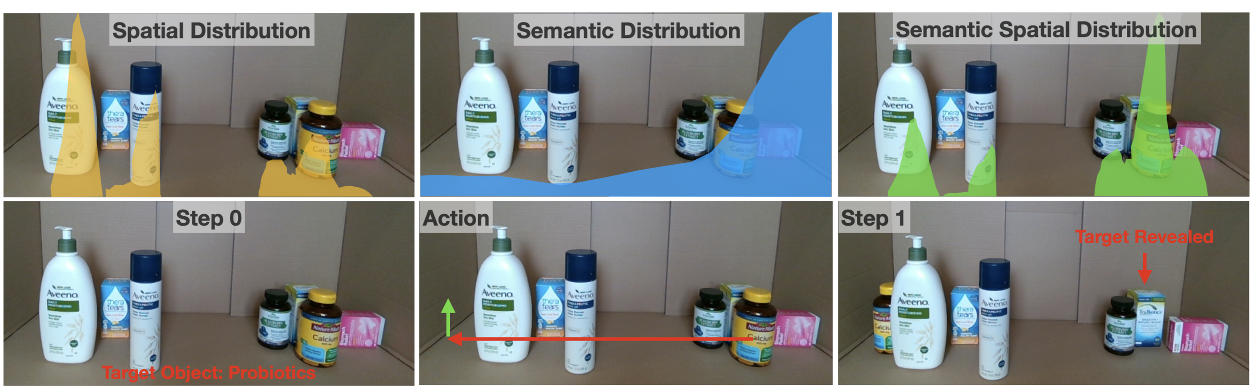

Figure 3.5: Physical rollout example with the target object being the probiotics. **Top:** the spatial distribution, semantic distribution and semantic spatial distribution for step 0. **Bottom:** RGB observations at step 0, the action given by DAR and the RGB observation after executing the action.

We show a physical experiment rollout with the target object being the probiotics as in Figure 3.5. In this rollout, the spatial distribution generated based on geometric information by LAX-RAY indicates the left side of the shelf occludes more area. However, the semantic distribution generated by SMS indicates the target object is more likely to be on the right. This is because other objects from the supplements category where the target object probiotics belongs to are visible on the right. Combining the spatial distribution and semantic distribution into the semantic spatial distribution takes into account both the geometry and semantic information and results in a more accurate distribution.

Preliminary Comparisons to GPT-4V

With the recent development of GPT-4V, we conduct a preliminary exploration to see if VLMs with strong reasoning abilities can create an explicit semantic distribution over an image of the scene. We use the following prompt with an image of the scene to extract a location within the image that should correspond to the highest activation. Since explicitly creating heatmaps is currently nontrivial, we ask GPT-4V to identify bounding boxes as it is an easier task. We use the prompt:

```
In this image, where are the couple most likely places in the
image I would find TARGET_OBJECT? List the places in decreasing
order of likelihood and explain why this place was chosen
(for example considering objects in that place). Explicitly
write one bounding box (written as a tuple) per place and code
with opencv2 to place the bounding boxes on the image. Fit the
bounding boxes to the object. The image has a width of WIDTH
pixels and a height of HEIGHT pixels.
```

where we substitute `TARGET_OBJECT, WIDTH, HEIGHT` for the target object, width of the image, and height of the image respectively. We design the prompt so the model has to explain its reasoning and write code, both of which have been shown to increase model performance [13].

We compare six examples where Figure 3.7 contains the responses of GPT-4V for each example and Figure 3.6 contains the visualization of the highest likelihood bounding box mentioned in the corresponding response and a comparison with SMS. We note that GPT-4V is very good at identifying objects in the image and semantically reasoning where the target object should be with respect to the objects it has identified in the image. Going through the examples, in example 1, it was able to correctly identify the stationary materials in the image and correspond the scotch tape to associated with that region. In example 2, it was able to perform OCR and identify the 'PATHFINDER 280' box and correctly reason that the microphone would be near that electronic packaging. However, the bounding box is not accurate as it only partially includes the 'PATHFINDER 280' box. In example 3, GPT-4V correctly identifies a trash can in the scene and that is the most likely location for an empty bottle, but fails to place an accurate bounding box around the trash can. A similar story happens in example 4 where GPT-4V identifies the ping pong table and reasons the paddle

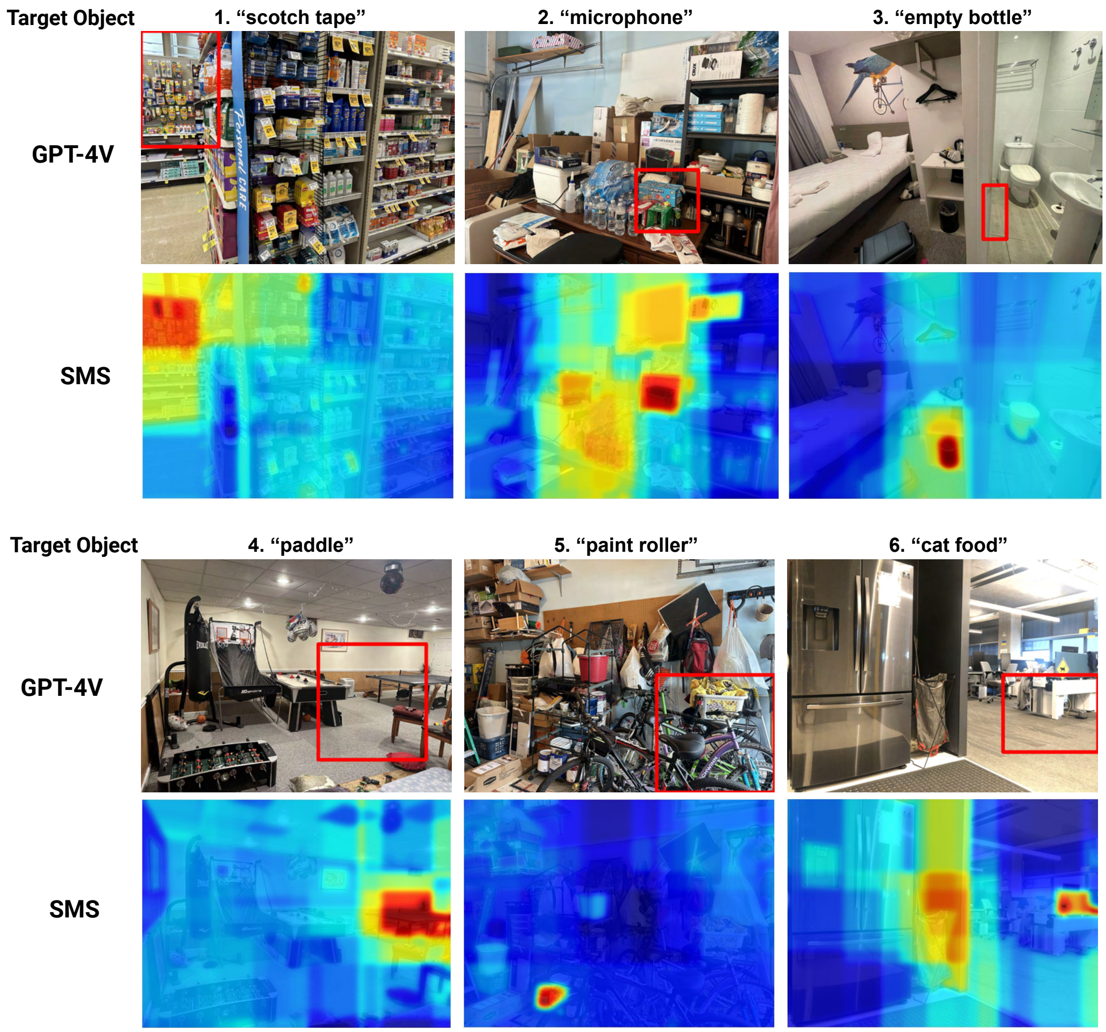

Figure 3.6: There are six examples, each example has the target object and the corresponding SMS semantic distribution and the GPT-4V comparison. The GPT-4V depicts a red bounding box, which is a visualization of bounding box (specified by the tuple) of the most likely place to find the target object, extracted from the corresponding response in Figure 3.7.

should be near the table but isn't able to place a tight bounding box around the table. In example 5, GPT-4V is able to reason the paint roller would be near the paint cans but the bounding box is predominantly encompassing the bicycles. Lastly for example 6, the bounding box primarily contains a desk and the floor rather than the cat sign.

This initial exploration suggests that GPT-4V is able to correctly reason about the objects in the image to determine what highly correlates to the target object, but GPT-4V is not able to reliably identify those regions in the image. The comparisons to SMS indicate that

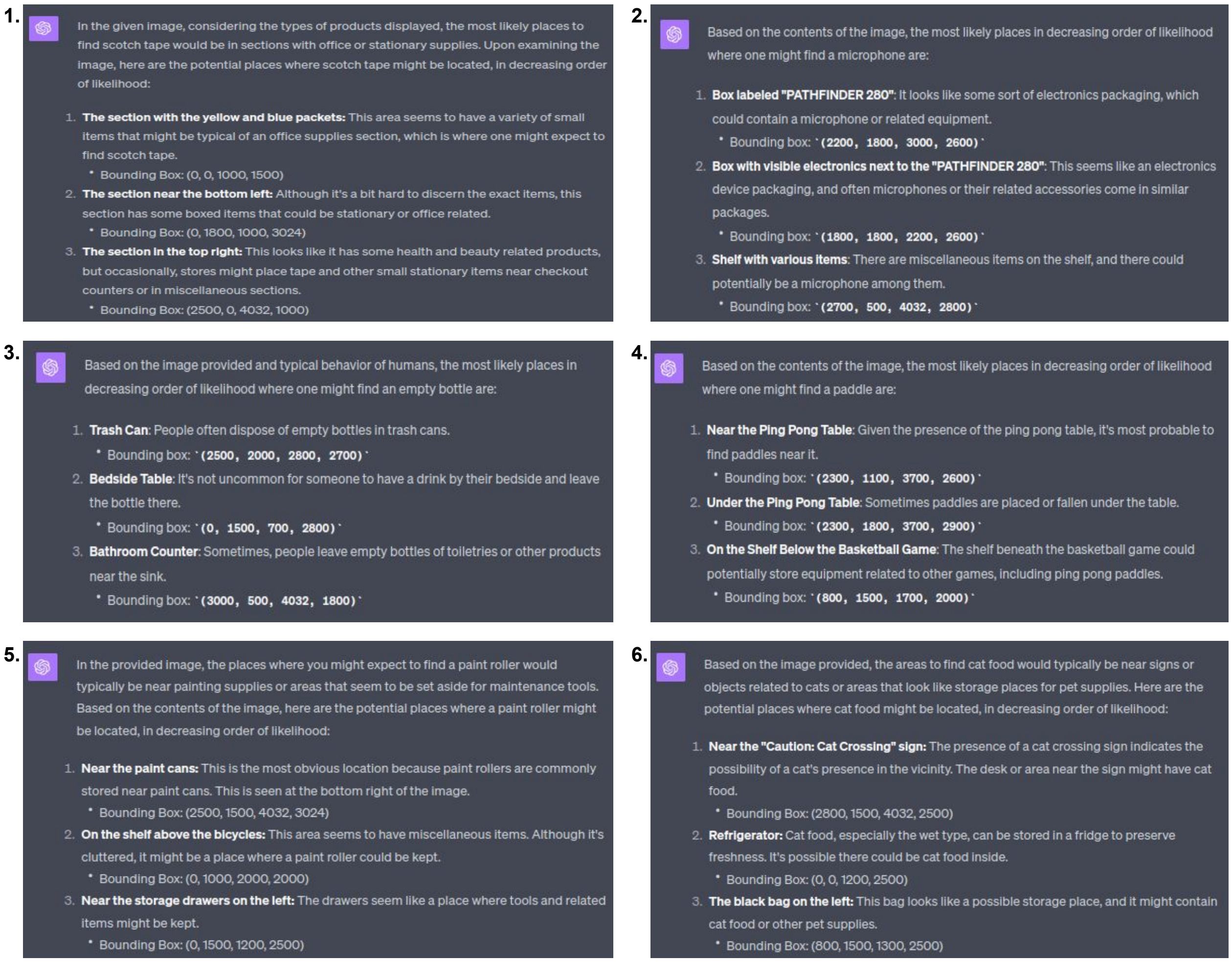

Figure 3.7: These six GPT-4V responses specify the bounding boxes that are visualized in Figure 3.6. The prompt used for each example is an image of the scene and what is stated in Section 3.5. These examples show that GPT-4V is good at object identification in the image and semantic reasoning to know near which objects in the image the target object would be. However, these bounding boxes do not reliably encompass the objects that GPT-4V references in the responses, indicating questionable object localization.

SMS is more reliable for creating explicit semantic distributions. Since the high performing closed-source VLMs (e.g. GPT-4V) can only be interacted through their language output, there is no current nontrivial method to extract accurate distributions from these models as the token probabilities are not available and the weights are not available to fine-tune the model for object localization. Future work could explore further prompt engineering, iterative adjustments with chain-of-thought prompting, and semantic distribution generation with diffusion.

Chapter 4

Language Embedded Radiance Fields for Zero-Shot Task-Oriented Grasping

4.1 Background

Many common objects must be grasped appropriately to avoid damage or facilitate performing a task: a knife by its handle, a flower by its stem, or sunglasses by their frame. Learning-based grasping systems exhibit impressive robustness on grasping arbitrary objects [131]–[139], but these systems typically measure grasp success based on whether the object was lifted [140]–[144]. Critically, these methods ignore an object's semantic properties: even if a robot could locate your favorite sunglasses, rather than safely grasp at the frame it may shatter the lenses. This ability to grasp an object part based on a desired task and constraints is called *task-oriented grasping*, and while well-studied [104], [105], [109], [110], [115], [145], previous methods collect specific object affordance datasets and struggle to scale to a diverse set of objects. Instead, the flexibility of natural language has the potential for specifying what and where to grasp. Thus, *LERF* for *Task-Oriented Grasping on Objects* (LERF-TOGO) enables task-oriented grasping through natural language by using large vision-language models in a zero-shot manner.

LERF-TOGO takes as input an object and a task-oriented object part name in natural language (i.e. *"flower; stem"*), and outputs a ranking over viable grasps on this object from which the robot should grasp. We build on recent work Language Embedded Radiance Fields (LERF) [101], which takes in calibrated RGB images and trains a standard NeRF in tandem with a scale-conditioned CLIP [146] feature field. Given a sentence prompt query, it outputs a 3D *relevancy* heatmap representing similarity to the query. However, these heatmaps may fail to highlight the full object (e.g., highlight only the bristles of a brush), which may cause issues when directly deployed to a task-oriented grasping task (grasp the "handle" of a brush). LERF-TOGO improves upon LERF's capabilities by predicting a 3D object mask using 3D DINO [147] features explicitly during inference. We propose a method of conditional LERF querying which restricts an object sub-part query to the

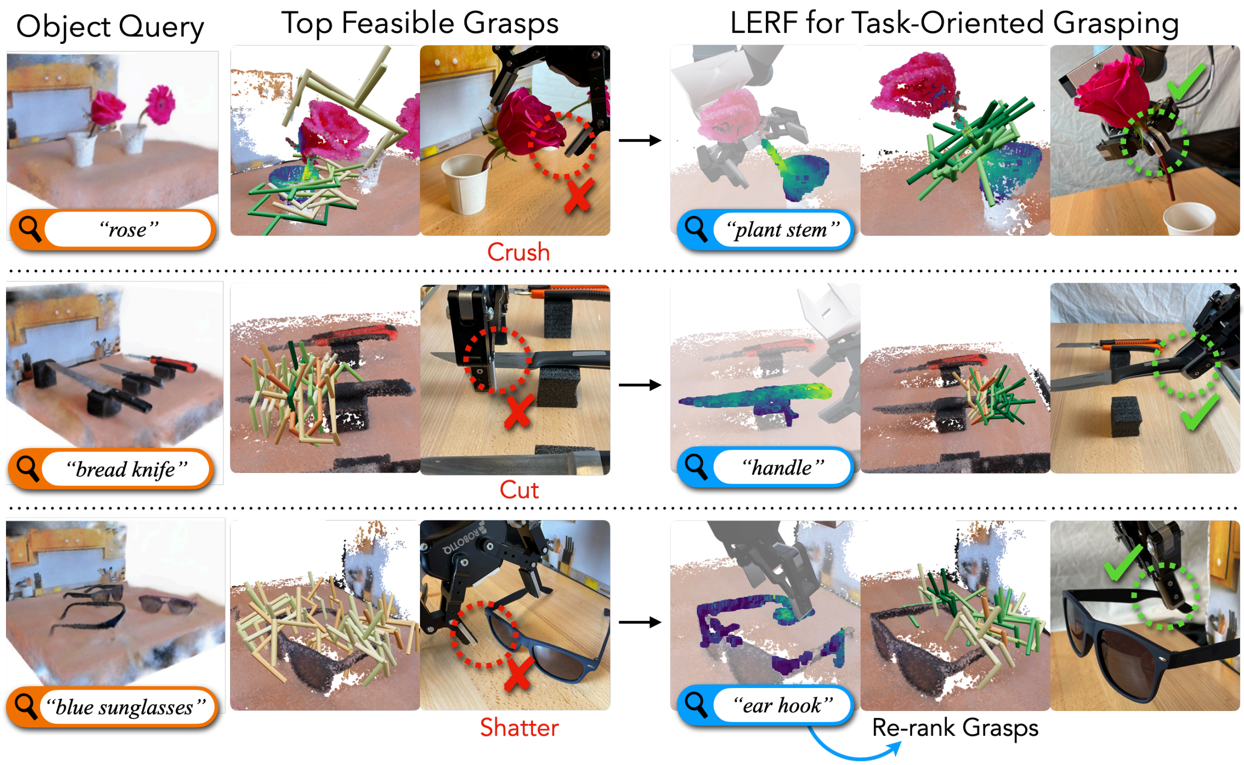

Figure 4.1: Learning-based grasp planners primarily consider object geometry, potentially yielding suboptimal grasps. LERF-TOGO uses natural language to select the target object with LERF [101] (in **orange**), and resamples grasps towards on object subparts using conditional LERF queries (in **blue**) for safe, task-oriented grasps.

object mask, leveraging the multi-scale nature of LERF to isolate specific regions within an object. LERF-TOGO then uses GraspNet [144] to generate grasps, re-ranking them based on the geometric and semantic distributions. We implemented a system with appropriate regularizations which allows LERF-TOGO to operate on a physical robot and evaluate on 39 common household objects. In experiments, 96% of grasps are on the correct object, 82% on the correct object part, and 69% result in a successful grasp. We design a robotic system that integrates LERF-TOGO on a physical robot to reconstruct a LERF of a scene, then execute task-oriented grasps through natural language to grasp semantically meaningful object parts.

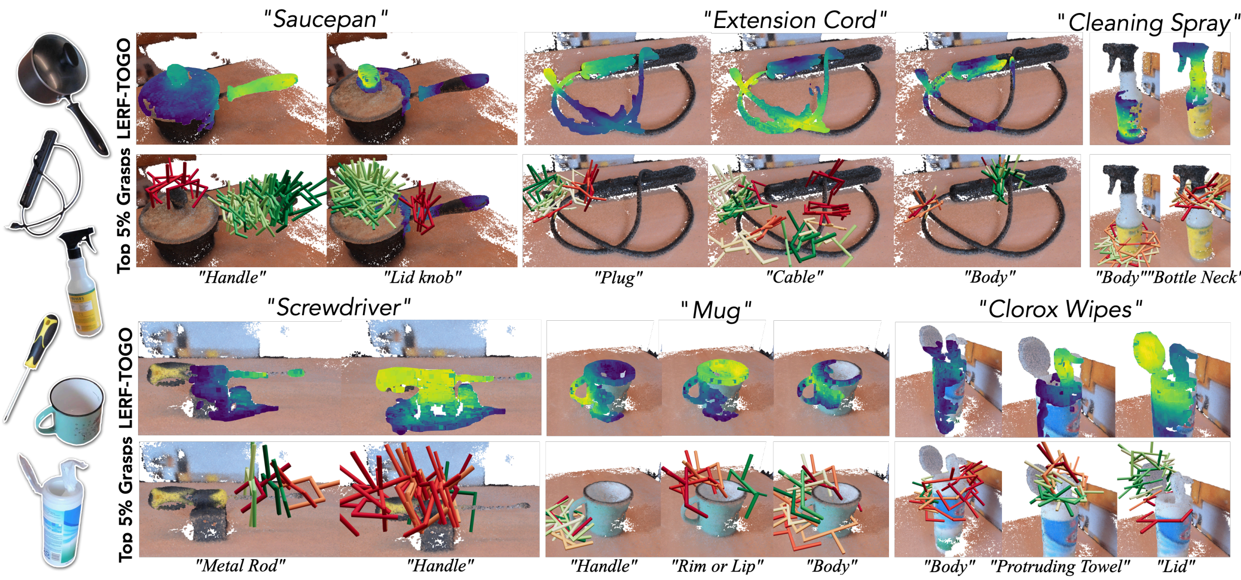

Figure 4.2: **Task-Oriented Grasps**: LERF-TOGO grasps target objects by different parts with different natural language queries (in quotes). Left: Crops of target objects. Right: Top row visualizes the relevancy distribution across the object mask, and bottom row shows top 5% of resampled grasps.

4.2 Preliminaries

Neural Radiance Fields (NeRF)

Neural Radiance Fields (NeRF) [148] are an attractive representation for high quality scene reconstruction from pose RGB images, with an explosion of recent work on visual quality [149]–[155], large-scale scenes [156]–[158], optimization speed [159]–[162], dynamic scenes [163]–[165], and more. Because of its high-quality reconstruction and differentiable properties, NeRF has been widely explored in robotics for navigation and mapping [149], [166]–[168], manipulation [169]–[172], and for synthetic data generation [173]. This work is most similar to works such as Evo-NeRF [171] which use NeRF as a real-time scene reconstruction to grasp objects. However, in contrast to previous works which only use RGB information, in this work we must include additional semantic information in 3D to select grasps falling on relevant target objects.

Several prior works explore using semantic outputs inside NeRF. Semantic-NeRF [174], Panoptic Lifting [175], and Panoptic Neural Fields [176] distill semantic categories from semantic segmentation networks into 3D to improve the 3D consistency of labels, particularly noting the denoising effect of averaging multiple views. Other works such as Distilled Feature Fields [177] or Neural Feature Fusion Fields [178] distill feature vectors from DINO and LSeg [56], and show they can be used for editing and scene segmentation. We build off of

LERF [101], which is described in the next section.

LERF Preliminaries

Language Embedded Radiance Fields (LERF) [101] is a recent representation that distills CLIP features into a NeRF. LERF inputs RGB images with camera poses and outputs a 3D field of DINO embeddings as well as a scale-conditioned CLIP field. This supports querying points in 3D for CLIP embeddings at different physical scales, capturing different semantics given different amounts of context. Given a text query, a relevancy value (from 0 to 1) can be generated at any 3D point by calculating the cosine similarity between LERF-queried embeddings and the CLIP embedding of query text. During this query process, a grid search on the scale parameter retrieves the scale with the highest activation. LERF is particularly attractive for task-oriented grasping because 1) its multi-scale parameterization allows queries at both object-level and part-level scales 2) LERF uses outputs from a pre-trained CLIP model without fine-tuning, which supports a variety of long-tail object queries not included in object or part segmentation datasets. LERF, however, tends to produce nonuniform activations on object queries because it lacks spatial grouping as shown in Fig. 4.8. In this work we show how to explicitly use the DINO feature field to obtain object masks to enable down-stream object part queries related to task-oriented grasping.

4.3 Problem Statement

Given a planar surface (table or workbench) containing a set of objects, the objective is for a robot to grasp and lift a target object specified using natural language. This query (e.g., *"sunglasses; ear hooks."*) includes both the object query (*"sunglasses"*) and the object part query, which specifies the part to grasp the object by (*"ear hooks"*). We experiment with lifting this assumption in Sec. 4.5 by leveraging an LLM for providing part queries. We assume access to a robot manipulator with a parallel jaw gripper and calibrated wrist-mounted RGB camera, and the objects in the scene are graspable by the robot. We also assume the object query specifies a present single object.

4.4 Algorithm

Given an object and object part query, LERF-TOGO outputs a ranking of viable grasps on the object part. To accomplish this, it first performs a robot capture of the scene and reconstructs a LERF (Sec. 4.4). Given a text query, LERF can generate a 3D relevancy map that highlights the relevant parts of the scene (Sec. 4.2). Second, a 3D object mask is generated using the LERF relevancy for the object query and DINO-based semantic grouping (Sec. 4.4). Third, a 3D part relevancy map is generated with a conditional LERF query over the object part query and the 3D object mask (Sec. 4.4). The part relevancy map is used to produce a semantic grasp distribution.

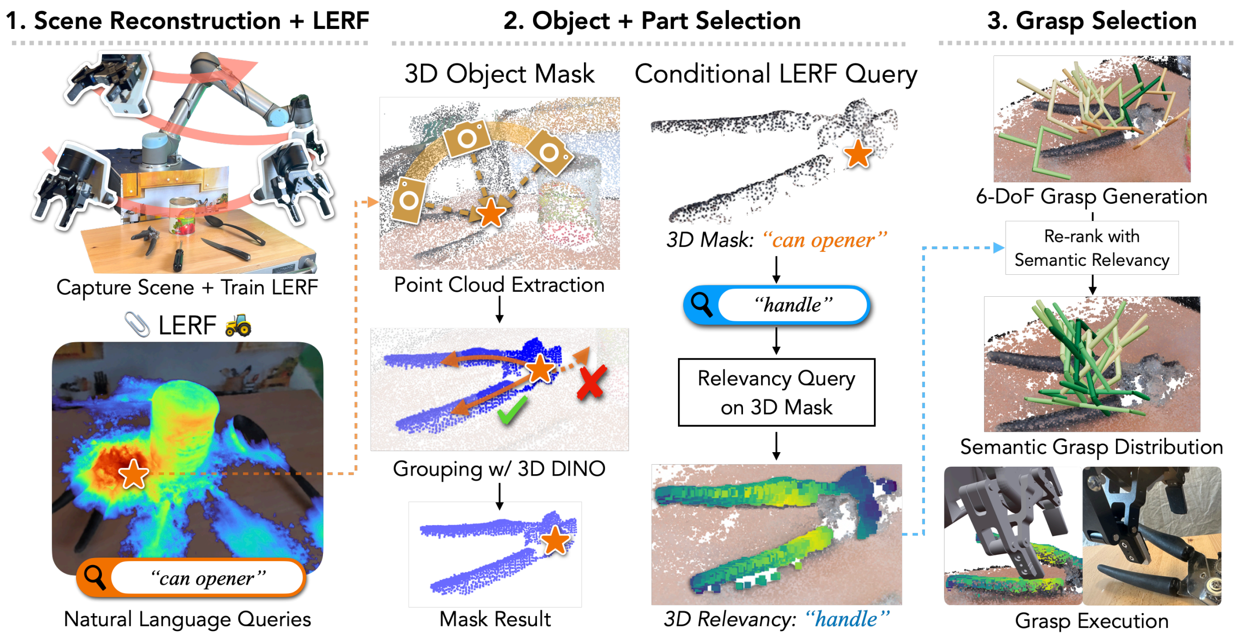

Figure 4.3: **LERF-TOGO Pipeline**: After reconstructing the scene with a wrist-mounted camera, we render an object-centric point cloud around the highest LERF activation. We next extract a 3D object mask by flood-filling the DINO features in this point cloud, condition an object part query on this object mask. Finally, we sample grasps and re-rank them according to 3D object part relevancy.

Robot Capture

Scene Reconstruction

The robot uses a wrist-mounted camera to capture the scene with a hemispherical trajectory centered at the workspace, similar to Evo-NeRF [171]. The capture has a radius of 45 cm and arcs from $\pm 100°$ around the workspace horizontally and an inclination range of $30°$ to $75°$. We capture images while the arm moves at 15 cm/sec at a rate of 3 hz, resulting in around 60 images per capture. We discard blurry images by analyzing the variance of the image Laplacian, ensuring the images are high quality. While the robot moves, we pre-process each image to extract DINO features, multi-scale CLIP, and ZoeDepth [179], which are used during LERF training.

Robot Capture Region Size

The robot captures the scene along a hemispherical trajectory arcing $\pm 100°$ around the workspace horizontally ("1/2 hemisphere" in Fig. 4.4.) When this horizontal sweep angle is reduced to a fraction of the range, the quality of the 3D object mask degrades, sometimes selecting the incorrect object altogether. LERF's semantic field is supervised on features of the scene images, thus the quality is heavily correlated with the distribution of images viewing the object. This lowers the quality of the 3D DINO embeddings used for the mask generation.

LERF Training Steps

In our experiments the LERF scene representation is trained to 2k steps. As shown in Fig. 4.4, objects/object parts (e.g., "spray nozzle", "bottle") can be detected in steps as low as 1k, but more fine-grained or smaller parts (e.g., "handle") may take longer (2-3k steps). This is consistent with what LERF reports: "fine-grained features take more steps [to emerge]".

3D Object Extraction

Given the initial 3D point with the highest LERF activation, we create an object-centric point cloud by rendering six different views looking at the 3D coordinate. The views are $\pm 90^\circ$ around the upwards vector through the 3D point. For DINO floodfill, the threshold DINO similarity is defined as first projecting the current DINO embedding onto the first PCA component of the top-down image, then taking the L2 norm of the difference between the current embedding and the DINO embedding at the initial 3D point.

NeRF Regularization

NeRF encounters difficulties in reconstructing texture-less planar surfaces, especially in the presence of specularities. This limitation is prominent in our table-top scenes, where the glossy surface and metallic objects can result in depth renderings with jagged missing regions. These missing regions can cause LERF renderings to spuriously activate and degrade the performance of grasp networks, so we apply depth regularization to mitigate this issue. We adopt the local depth ranking loss proposed in SparseNeRF [180] and use ZoeDepth [179] as the underlying depth model. We found this performs better than smoothness priors [162], [181] because it retains more fine-grained geometry. Additionally, we use the gradient scaling approach from [155], which significantly reduces the number of near-camera floaters and enables more robust grasping directly from point clouds rendered from the NeRF.

Poses obtained from cameras in motion are slightly inaccurate, which we found could result in oversmoothed geometry with depth regularization. To overcome this, we optimize the NeRF for the first 500 steps without any regularization to allow the camera poses to settle, then anneal the depth regularization loss term from 0 to 100% over the next 1500 steps. Interestingly, we find staged training not only preserves thin features better but also speeds LERF optimization. We hypothesize this is because supervising the language field on un-converged density in free space results in a poor network initialization, while beginning LERF optimization after geometry has been largely removed from free space allows a smoother learning signal.

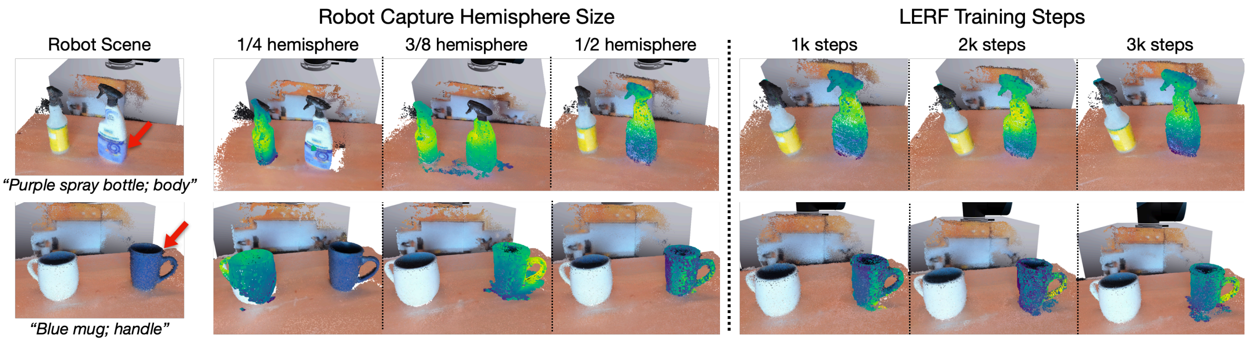

Figure 4.4: Left: Decreasing the robot scan region degrades the quality of 3D relevancy map generated by LERF-TOGO. Right: LERF-TOGO relevancy map converges earlier for larger objects and parts.

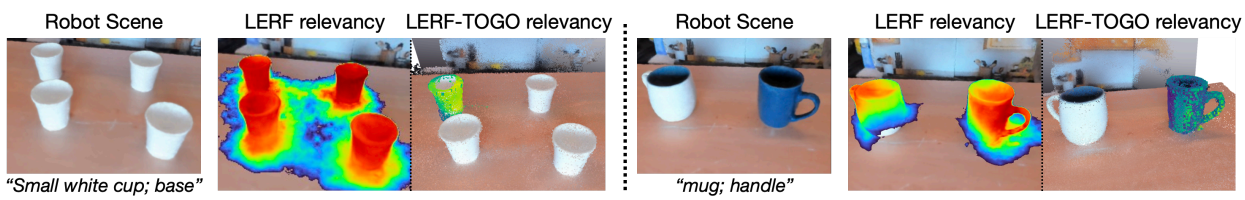

Figure 4.5: In the case of identical objects or ambiguous object queries, LERF-TOGO picks a single object in the scene but does not propose grasps for all the objects in the category.

3D Object Extraction

An important limitation of LERF is its lack of spatial grouping within objects: for example given *"can opener"*, LERF tends to highlight regions of the object that most obviously identify that object (e.g. the metal cogs on the can opener as shown by the orange star in Fig. 4.3). However, since the region that visually identifies the object and the desired grasp location (e.g. handle) can differ significantly, this is problematic. LERF inherently exhibits such local behavior because it trains on local crops of input images, causing CLIP embeddings surrounding the handle to be unaware if it belongs to a can opener. LERF-TOGO overcomes this by finding a 3D object mask given a language query, which groups the object part together with the LERF activation. To create the object mask, we leverage the 3D DINO embeddings [147] (self-**DI**stillation with **NO** labels) present within LERF during inference, because DINO embeddings have been shown to exhibit strong object awareness and foreground-background distinction [113], [147], [182].

First, we obtain a coarse object localization from LERF by rendering a top-down view of the scene and querying the object. We produce a foreground mask by thresholding the first principal component of the top-down rendered DINO embeddings, and constrain the relevancy query to this mask to find the most relevant 3D point. We then refine this single-point localization into a complete object mask. We render an object-centric point cloud around this 3D point by deprojecting NeRF depth from multiple views, and then iteratively grow the object mask by including neighboring points to the frontier which lie within a threshold DINO similarity (similar to floodfill). The output of this process is a set of 3D points lying on the target object.

Conditional LERF Queries

Another important challenge of using CLIP is its tendency to behave as a bag-of-words [146]: the activation for *"mug"* behaves very similarly to *"mug handle"* because CLIP latches onto individual words, not the grammatical structure of sentences. To mitigate this phenomenon, LERF-TOGO introduces a conditional method of querying LERF relevancy by composing two related queries, similarly to how composing prompts has shown promise in generative modeling for guiding specific properties [183]. Because LERF is scale-conditioned, during inference it searches over scales for a given query and returns the relevancy at the scale with the highest activation. To condition a LERF query, LERF-TOGO searches only on the points within the 3D object mask. Intuitively, this results in a distribution over the object's 3D geometry representing the likelihood that a given point is the desired object part, which can be used for biasing grasps towards this region.

Grasping

Grasp Sampling

Ensuring complete coverage of grasps on objects is critical to avoid missing specific object parts. We use GraspNet [144], which can generate 6-DOF parallel jaw grasps from a monocular RGBD point cloud, but from a single view it often misses key grasps on target object parts. To mitigate this, and to leverage the full 3D geometry available within NeRF, we create a hemisphere of virtual cameras oriented towards the scene's center. For every virtual camera, we convert the scene's point cloud to the camera coordinate frames before providing it as input to the pretrained GraspNet model. To obtain the final set of grasps for the scene, we combine the generated grasps from the virtual cameras using non-maximum suppression to remove duplicates.

Grasp Ranking

Given the grasps in the previous step (the *geometric* distribution), we now combine it with the semantic distribution across an object obtained from LERF-TOGO. The semantic score s_{sem} for a given grasp is computed as the median LERF relevancy of points within the grasp volume. The geometric score s_{geom} is the confidence output from GraspNet, indicating grasp quality based on geometric cues. To balance relevance and success likelihood, we combine the grasp score $s = 0.95s_{sem} + 0.05s_{geom}$ to ensure that we consider the most relevant grasps while slightly biasing towards confident grasps.

Figure 4.6: **Left**: Select scenes used during experiments, **Right**: example grasps using LERF-TOGO.

Motion Planning

A grasp is considered feasible if the robot can perform a collision-free trajectory with the following poses: the pre-grasp, grasp, and post-grasp configurations. The pre-grasp pose is positioned 5cm along the z-axis of the robot end effector, which allows the gripper to approach the target grasp pose with minimal additional motion. The post-grasp pose is located 10cm above the grasp pose, along the z-axis of the world frame. The UR5's IK solver calculates the set of viable joint configurations at these poses, and we calculate the trajectory as a linear interpolation between them. We additionally allow for a 180 degree rotation at the last wrist joint, as parallel-jaw grasps are rotationally symmetric. This facilitates the motion planning process, as the robot's camera mount is prone to colliding with the robot arm.

4.5 Experiments

Part-Oriented Grasping

We evaluate LERF-TOGO on a wide variety of 31 different objects and 49 total object parts to grasp (Fig. 4.6). For each object, we select an object query by describing it sufficiently to unambiguously differentiate between other objects in the scene. We use semantic descriptions when possible, and add visual descriptions *only* when such descriptions are ambiguous (i.e using color to differentiate multiple mugs in a scene). We provide a part query for each object by describing a natural place for a robot to grasp and lift (i.e. "handle", "plant stem", "ear hook", "frame"). In addition, several objects include different grasp locations. A grasp is successful if it lifts the correct object using the appropriate subpart at least 10cm vertically, and the object remains securely within the gripper jaws throughout. For each query, we measure 1) whether the selected grasp was on the correct *object*, 2) whether the selected grasp was on the correct object *part*, and 3) whether the grasp successfuly *lifted* the object

	ConceptFusion	LERF-TOGO
Correct Object	77%	**96%**
Correct Part	39%	**82%**
Successfuly Lifted	–	**69%**

Table 4.1: **Part-Based Grasping Results:** Results are reported across 49 different prompts and 12 scenes. [25] includes a complete list of scenes and queries.

from the table. Every scene is reconstructed once in the beginning, after which the objects are removed sequentially (i.e., objects are removed one-by-one) with no updates in the scene representation.

LERF-TOGO overall achieves a 69% success rate for physically grasping and lifting objects by the correct part. The selected grasp was located around the correct object part 82% of the time, with the remaining failures being grasp execution failures. For context, the highest confidence geometric grasp on an object mask only lies on the correct part 18% of the time, suggesting LERF-TOGO meaningfully biases the grasp distribution to the object part. Selected task-oriented queries are visualized in Fig. 4.2: the distribution of grasps drastically shifts based on the given part query, and can focus task-oriented grasps on multiple different regions per object based on the language prompt. LERF-TOGO shows strong language understanding performance for object selection (96%), able to differentiate between very fine-grained language queries like color, appearance ("matte" vs "shiny"), or semantically similar categories (*"popsicle"* vs *"lollipop"*). It also can recognize long-tail object queries like *"ethernet dongle"*, *"cork"*, or *"martini glass"*, owing to its usage of CLIP zero-shot.

Task-Oriented Grasping

LERF-TOGO accepts a natural language part query as input, allowing it to be used alongside large language models (LLMs) to generate parts based on the task. To investigate if the LLM can also generate the object part, we use an LLM (ChatGPT) to generate the object and part query automatically via few-shot prompting. Results are shown in Table 4.3. The prompt and all tasks are included in [25]. Given the task and the list of objects in the scene, the LLM is tasked with generating the correct object and object part pair (object, part). We used a majority voting scheme to query the LLM. Given the task, the LLM provides seven candidates that we use to select the pair (object, part) that appears in a majority of the responses. We also mention details to integrate with an LLM planner in Section 4.5.

Combining few-shot LLM prompting with LERF-TOGO identifies the correct primitive with 92% success and produces grasps on the correct object with 71% success across 49 tasks on 39 different objects. The LLM could identify the object in the scene with the same success rate as the human, giving the correct object and part pairs for tasks like "scrub the dishes" and "cut the steak". However, the LLM had a lower success rate (71%) compared to the human (82%) for object part selection. This is because CLIP, and by extension LERF-TOGO, can be sensitive to subtle variations in wording like "body" vs. "base" resulting in different LERF activations and thus grasps.

Baselines

	SemAbs	OWL-ViT
Correct Object	80%	85%
Correct Part	35%	50%

Table 4.2: **Single View Comparisons:** Results are reported across 20 different prompts and 5 scenes. [25] includes a complete list of scenes and queries.

	Human-Query	LLM-Query
Correct Object	**96%**	**96%**
Correct Part	**82%**	71%

Table 4.3: **Task-Oriented Success:** Results using an LLM to choose the object part given a task specification. Results are reported across 49 different prompts and 12 scenes.

ConceptFusion [184]

ConceptFusion generates a multimodal point cloud of a scene by fusing RGBD images and their extracted features together. To query ConceptFusion, we provide it with the concatenated object and part prompt (i.e. *"mug handle"*) and rank grasps via the highest similarity. We report the object and part success without physical evaluation. We provide ConceptFusion with depth generated from the NeRF, which results in high-quality point clouds for grasping. To represent the paper, we use the OpenCLIP ViT-H/14 model, which is larger than the ViT-B/16 model for LERF-TOGO.

LERF-TOGO out-performs ConceptFusion by 43% (Table 4.1) at task-oriented grasping because it can capture multi-scale semantics, while ConceptFusion is limited to one CLIP embedding per point. This makes hierarchical querying difficult, and is reflected by the fact that ConceptFusion performs similarly to LERF-TOGO at selecting the correct object, but suffers at selecting the right object part. Due to its lack of scale-conditioning, ConceptFusion frequently emphasizes sections of the table due to the inclusion of both the objects and the table itself in the mask proposals (Fig. 4.7).

Semantic Abstraction [60]

Semantic Abstraction takes a single RGBD frame as input and a text query and outputs a relevancy heat map over the image. A query is a success if the majority of the heatmap overlaps with the object part. In instances where activations are detected on other objects,

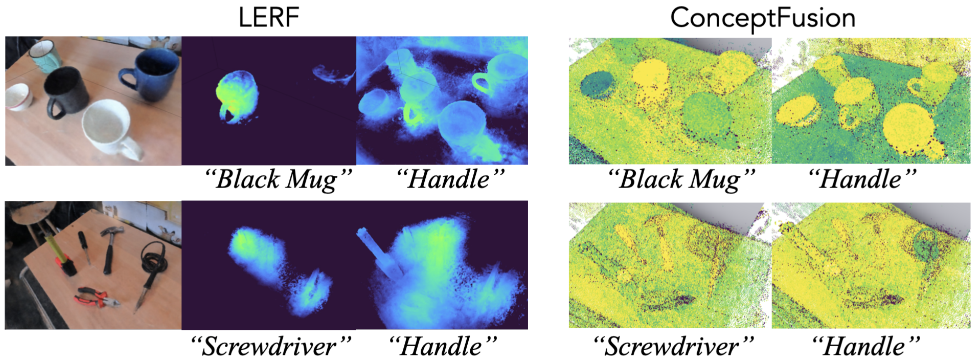

Figure 4.7: **Comparison with LERF and ConceptFusion**: ConceptFusion performs well on object-level queries, but struggles with sub-object part queries because of its lack of multi-scale semantics.

it is considered successful if the highest activation is on the desired object part. Detailed results can be found in Table 4.2. Since the method takes a single image, we provide the method with an input image observing all object parts for a fair comparison. We provide it with part queries 2 ways and take the best performance: 1) the concatenated object and part prompt (i.e. *"mug handle"*), and 2) the object and part as separate queries.

Semantic Abstraction achieves an overall object detection rate of 80% and part detection of 35%. The method tends to produce empty relevancy responses when queried for specific object parts, potentially owing to its averaging across multiple scales which drowns out smaller part features. When presented with the object and part, the method highlights all of the object, owing to CLIP's bag of words behavior, a characteristic addressed by LERF-TOGO's compositional queries. (Fig. 4.9).

OWL-ViT [102]

OWL-ViT is an open-vocab detector which takes in an RGB image and text prompts and outputs segmentation maps. We provide OWL-ViT a single input image that encompasses all object parts for a fair comparison. To obtain an object mask, we use the object prompt to establish an initial bounding box. This box serves as a region to identify the highest-scoring part within the region. In order to deem the part box as successful, we visually confirm that it aligns with the object part. Results can be found in Table 4.2.

OWL-ViT achieves 85% accuracy for object localization, struggling on very long-tail objects that were not encountered within the detection datasets. This behavior is amplified for object part queries, where queries tend to be long-tail such as *"measuring tape"* and *"ethernet dongle"*. Example queries are shown in Figure 4.10.

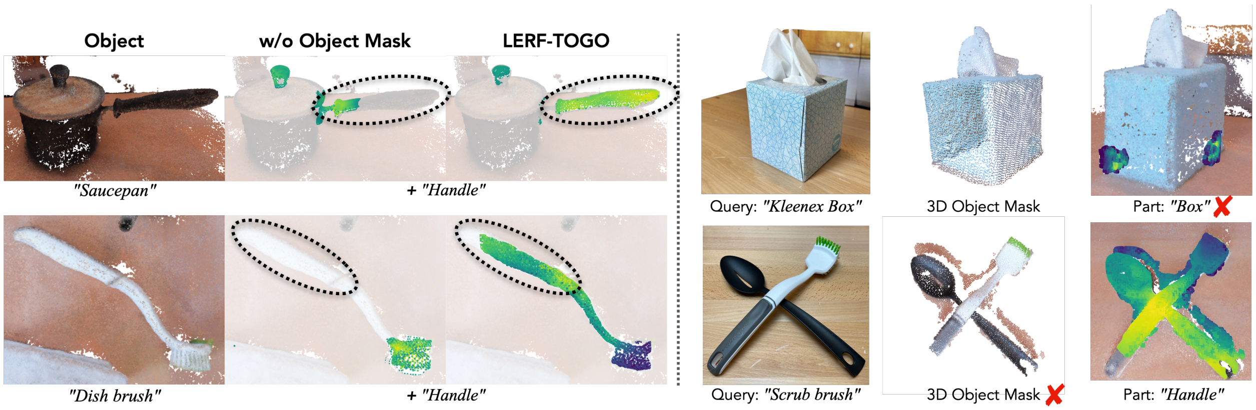

Figure 4.8: **Ablation and Limitations**: *Left*: Without 3D object masking and conditional querying, LERF cannot capture oblong object shapes. *Right*: CLIP can sometimes fail on generic prompt queries, like the poor activation on the box. Additionally, LERF-TOGO struggles with groups of connected objects as the 3D object mask groups them all together.

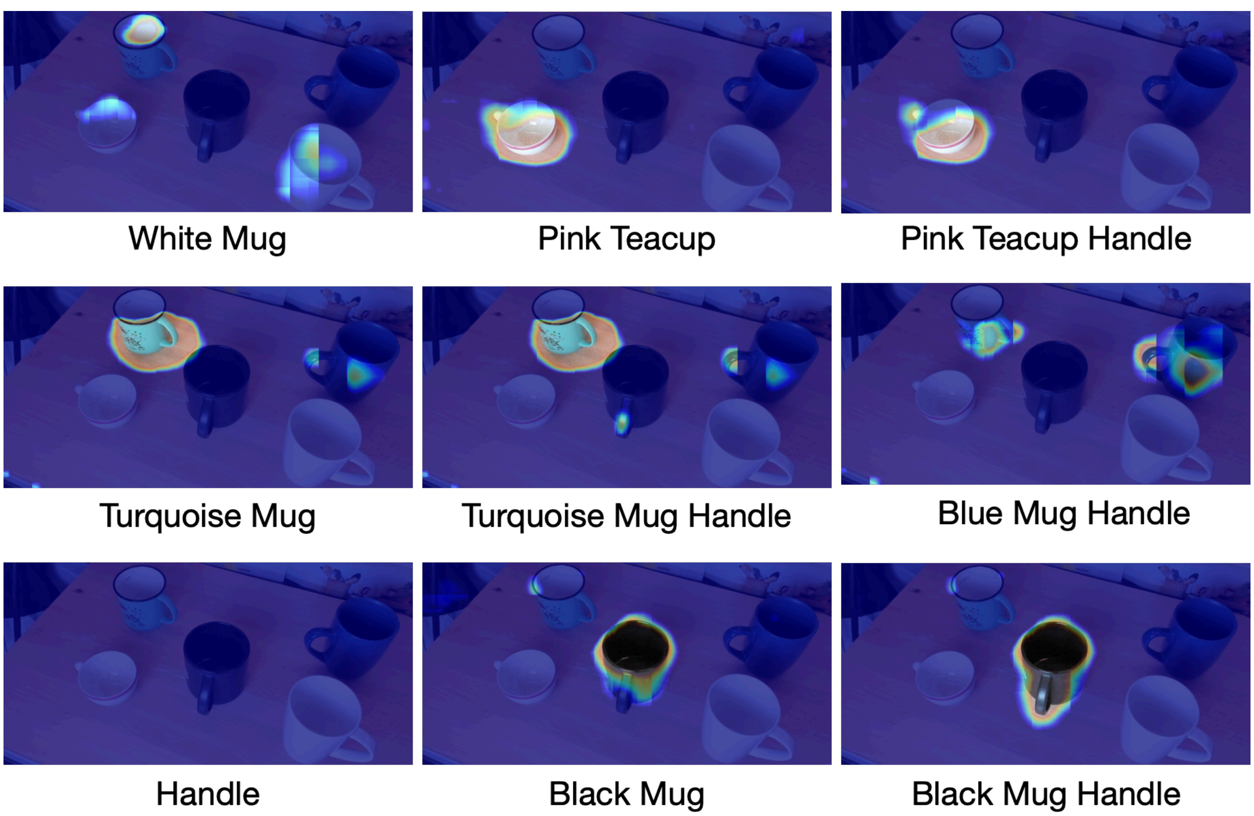

Figure 4.9: Semantic Abstraction results for object and object part localization

Ablations and Failure Modes

We ablate object extraction by removing 3D object masking and conditional querying, and found that LERF-TOGO suffers with oblong objects, as shown in Fig. 4.8. We compare against querying LERF individually for the object and part, and multiplying their results together. This produces fragmented results which can ignore relevant parts of the object for part queries. We also ablate the grasp weighting in Figure 4.11 to see a $w = 0.95$ is needed to heavily skew grasps to the semantic distribution, allowing for grasps on the plug and stem. We also ablate robot capture hemisphere and LERF training steps to see how LERF quality degrades with smaller hemispheres and lower training steps.

The primary failure modes of LERF-TOGO are mistaking visually similar object parts for one another (eg teapot spout for a handle), missing subtle geometries like the small teacup handle or spray trigger, or confusing very close categories like steak and bread knives. We also observe prompt sensitivity with part queries: for example *"bottle neck"* more strongly localizes grasps than *"neck"*, and without more prompt tuning *"body"* sometimes fails to highlight the bases of bottles.

Integration with an LLM Planner

LERF-TOGO can integrate as a module with an LLM planner to combine task-oriented grasps for robotic manipulation tasks. We define a set of robotic manipulation primitives (grasp, press, twist, pick&place, pour) and prompt the LLM to output the correct primitive for a given task. We use the same majority voting scheme in the previous section to select both the correct robotic primitive and the pair (object, part). Now, given a task (e.g. 'uncork the wine'), an LLM can specify the action to accomplish the task ('grasp') and the pair of object and object part (e.g. 'wine' and 'cork').

We provide the full prompt to the LLM below. For any given task and scene the OBJECT_LIST is replaced with a list of objects within the scene and TASK is replaced with the desired task:

```
    Answer the question as if you are a robot with a parallel jaw gripper
    that has access to only the objects in the object list. Follow the
    exact format. First line should describe what basic action is needed
    to do the task from the following set of actions: press, grasp, twist,
    pick & place. The second line should only be an object from the object
    list followed by 1 object part that the robot would touch to do this
    task. VERY IMPORTANT: If the basic action is pick & place, only then
    have a third line with 'Place: ' to specify the object to place on. \
Object list: ['pot', 'knife', 'spoon', 'black pan'] \
Q: How can I safely pick up a pan? \
Basic Action: grasp \
Sequence: 1. black pan 2. handle \
\
```

```
Object list: ['mechanical keyboard', 'knife', 'TV', 'camera'] \
Q: How can I safely hit the spacebar on a keyboard? \
Basic action: press \
Sequence: 1. mechanical keyboard 2. spacebar\
  \
Object list: ['green mug', 'blue spoon', 'fork', 'knife'] \
Q: How can I cut a block of cheese? \
Basic action: grasp \
Sequence: 1. knife 2. handle \
  \
Object list: ['salt shaker', 'knife', 'fork', 'white pan'] \
Q: How can I safely lift a salt shaker? \
Basic Action: grasp \
Sequence: 1. salt shaker 2. base \
\
Object list: ['red cup', 'blue cup', 'mug', 'bowl'] \
Q: How do I stack the red cup on the blue cup? \
Basic action: pick & place \
Sequence: 1. red cup 2. rim \
Place: blue cup \
  \
Object list: ['door knob', 'black mug', 'green dish brush', 'shiny knife'] \
Q: How do I open a door knob? \
Basic action: twist \
Sequence: 1. door knob 2. rim \
  \
Object list: ['dryer', 'washing machine', 'sunglasses'] \
Q: How do I turn on the washing machine? \
Basic action: twist \
Sequence: 1. washing machine 2. dial \
  \
Object list: ['paper towel roll', 'mug', 'teacup', 'headphones', 'pen'] \
Q: How do I grab a paper towel? \
Basic action: grasp \
Sequence: 1. paper towel roll 2. paper towel \
  \
Object list: ['magnifying glass', 'blue spoon', 'fork', 'knife'] \
Q: How do I pick up a magnifying glass? \
Basic action: grasp \
Sequence: 1. magnifying glass 2. handle \
  \
Object list: ['teddy bear', 'toy block', 'mouse', 'saucepan', 'hammer'] \
```

```
Q: How do I grab a teddy bear? \
Basic action: grasp \
Sequence: 1. teddy bear 2. head \
\
Object list: ['green mug', 'blue spoon', 'fork', 'knife'] \
Q: How do I put the mug in the cabinet? \
Basic action: pick & place \
Sequence: 1. green mug 2. handle \
Place: cabinet \
\
Object list: {OBJECT_LIST} \
Q: How can I safely {TASK}? \
Basic action: "
```

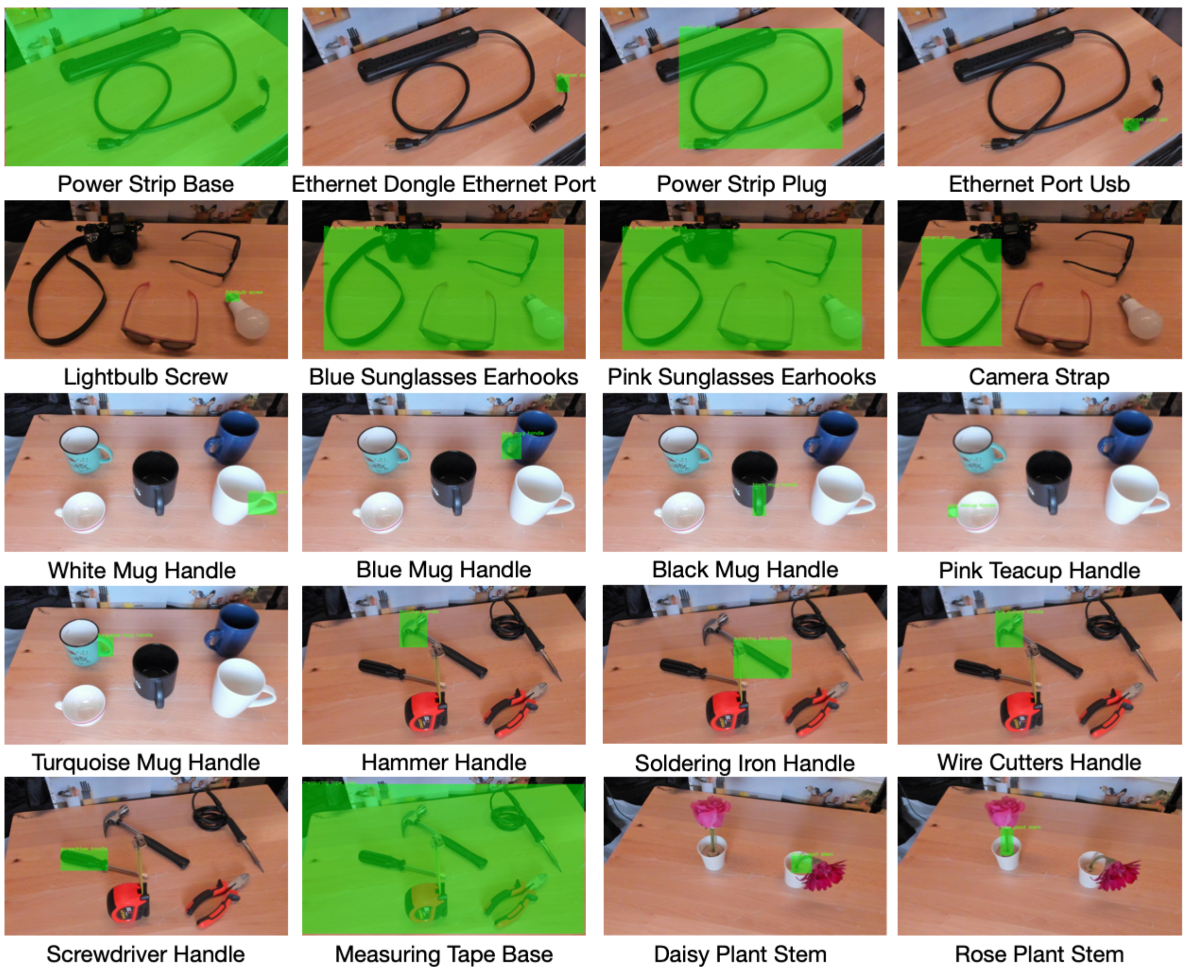

Figure 4.10: OWL-ViT results for object and object part localization

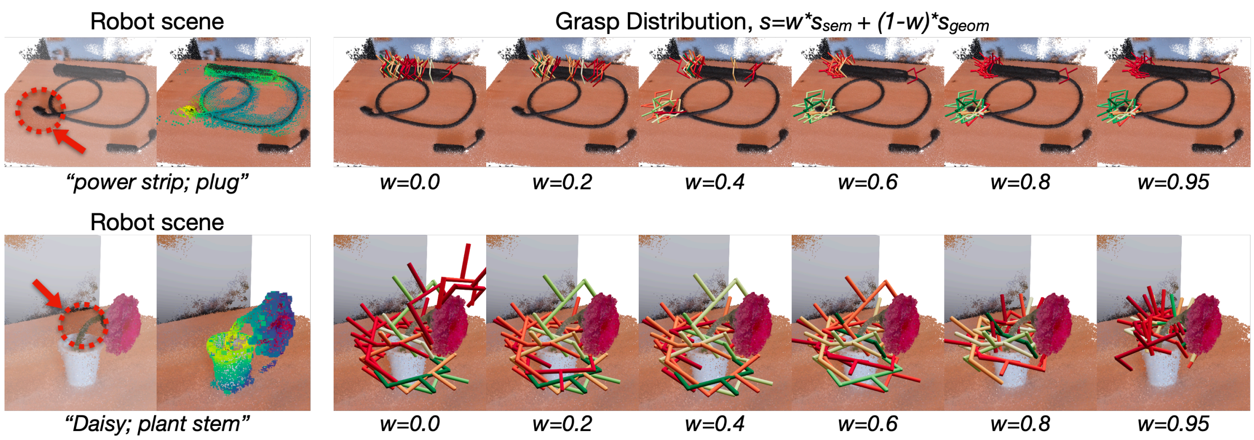

Figure 4.11: **Grasp score weighting**: Varying weight between geometric grasp score and semantic grasp score shifts the grasp distribution. A high semantic grasp weight ($w = 0.95$) is required, since geometric grasps may be biased away from small and fine-grained object parts of interest. Both geometric and semantic scores are in the range [0, 1].

Chapter 5

Limitations and Conclusion

5.1 Semantic Mechanical Search

We first presented Semantic Mechanical Search (SMS), an algorithm for semantic distribution generation using VLMs zero-shot for a fully-occluded target object. SMS facilitates mechanical search in closed-world environments and improves semantic distribution quality for open-world environments. SMS has several limitations, which open up possibilities for future work: (1) We only evaluate open-world semantic distribution quality on a static dataset; (2) While SMS, which operates in the LLM feature space, generates better semantic distributions than CLIP-based method, we have not compared to other VLMs such as GPT-4 [185] or LLaVa [186] due to their inaccessibility to obtain affinity scores. VLMs with strong reasoning abilities, such as GPT-4V [187], have the potential to directly generate high-quality semantic distributions. Further applying GPT-4V in object search would be an interesting future direction. We conduct an initial exploration in Section 3.5. (3) SMS for closed-world relies on creating an offline affinity matrix which can take a few minutes with large object lists, while SMS for open-world takes 35 to 45 seconds for each semantic distribution (4) The performance of SMS is sensitive to the quality of each module in the framework.

5.2 LERF-TOGO

Then, we presented LERF-TOGO, a method for using VLMs zero-shot with Language Embedded Radiance Fields to grasp objects and their parts via language. By improving the spatial grouping of LERF relevancy outputs, LERF-TOGO can support hierarchical part queries conditioned on the full object. Results indicate it performs strongly at language-guided grasping, with grasps landing on the correct object 96% of the time, and furthermore can direct grasps to the correct object parts 81% of the time. The main limitation of LERF-TOGO is speed: the entire end-to-end the process takes a few minutes which can be impractical for time-sensitive applications. Future work on additional regularizations and optimizations to LERF training may reduce computation time. Another key limitation of LERF-TOGO is with groups of connected foreground objects, as the DINO flood-fill mask will include the whole foreground group instead of isolating the individual components. Supporting hierarchy within foreground groups is critical to enable such cases. If the object query matches multiple objects in the scene (identical objects, or generic query like "mug" in the mugs scene), the system will arbitrarily choose only one of them. LERF-TOGO does not consider referring/comparative expressions (e.g., "mug next to the plate", "biggest mug"). In addition, though we present a method for obtaining object part queries from input task descriptions via LLMs, in future work we will evaluate its performance on a diverse set of tasks.

5.3 Conclusion

With both of these projects, we attempt to address the underlying question of how can we use VLMs zero-shot to create useful state representations for robotic tasks. SMS creates a 2D semantic distribution relatively quickly but with less precision as that is sufficient for the task of occluded object search. On the other hand, LERF-TOGO offers more precision as is necessary for dexterous manipulation but is slower and only works for static scenes. Thus, the characteristics of the robotic task that we are trying to solve dictate the inductive biases for our visual-language state representation. In LERF-TOGO, since the task is grasping, we use 3D representations for precise, denoised semantic embeddings. In SMS, the task is occluded object search, so we explicitly use LLMs with VLMs to better semantically reason about visually occluded objects. Both of these projects have robotic perception modules leverage VLMs but with different inductive biases depending on the task. Thus, these methods further generality along a particular axis, namely being able to handle diverse environments and long-tail objects. These methods offer an improvement over previous perception modules that rely on an intricate system of trained classifiers and traditional vision techniques as they do not scale well along this axis. Now, the goal we should aim for is to have generality along a different axis: across different robotic tasks. In my work, I largely focus on developing a general perception module for individual robot tasks, but the goal remains to have a single robot perception module that is task-agnostic and can be used with any robotic system. This module should ideally create a real-time 3D state representation that can incorporate new visual and semantic information at every timestep. Recent work in real-time 3D rendering [188] indicates promise for the near future.

However, let's take a step back and consider if we should be using VLMs to create **explicit** state representations, thereby incorporating some potentially suboptimal inductive bias into the perception modules. Explicit representations provide numerous advantages; they are interpretable and necessitate the development of modular algorithms, making them suitable for safe integration in any existing robotic system. However, are explicit representations the correct way to approach robot perception? Recently, roboticists have entertained building vision-language-action (VLA) models [189] trained with scaled-up robotic datasets collected with robots in the real world [190]. This approach incorporates minimal inductive bias for robotic perception as it goes directly from images (or other sensor measurements) to actions. The obvious drawback of this approach is the lack of interpretability as we do not know how the model is reasoning about the scene to influence its predicted action. However, this approach has seen some promise as these VLA models seem to be learning a task-agnostic robotic perception system simultaneously while learning a general policy to output actions conditioned on a particular task. Here there is no explicit state representation that captures semantics, rather the model implicitly reasons about semantics and directly influences the action for the task accordingly. More generally, the use of explicit 3D representations is beginning to be questioned as video generation models like Sora from OpenAI [191] can generate realistic, 3D consistent videos without explicitly adding specific inductive biases for 3D reasoning. More generally than robotic perception, there remains an open research question in robotic learning: should we create modular systems with intermediate explicit representations or train end-to-end models that go from images to actions? As roboticists, we are now starting to tackle this fundamental question in hopes of creating general, useful robots for people.

5.4 Final Notes

I am incredibly fortunate that I had an amazing research environment in the AUTOLab where I had exceptional mentors and great friends. I am forever grateful to Professor Goldberg for helping me to develop my research skills and for starting me on my research journey. In the fall, I will start as a PhD student at Stanford where I will continue research in robot learning.

Bibliography

[1] S. Parisi, A. Rajeswaran, S. Purushwalkam, and A. Gupta, "The unsurprising effectiveness of pre-trained vision models for control," in *International Conference on Machine Learning*, PMLR, 2022, pp. 17 359–17 371.

[2] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, "R3m: A universal visual representation for robot manipulation," *arXiv preprint arXiv:2203.12601*, 2022.

[3] I. Radosavovic, T. Xiao, S. James, P. Abbeel, J. Malik, and T. Darrell, "Real-world robot learning with masked visual pre-training," in *Conference on Robot Learning*, PMLR, 2023, pp. 416–426.

[4] Y. J. Ma, S. Sodhani, D. Jayaraman, O. Bastani, V. Kumar, and A. Zhang, "Vip: Towards universal visual reward and representation via value-implicit pre-training," *arXiv preprint arXiv:2210.00030*, 2022.

[5] A. Stone, T. Xiao, Y. Lu, K. Gopalakrishnan, K.-H. Lee, Q. Vuong, P. Wohlhart, B. Zitkovich, F. Xia, C. Finn, *et al.*, "Open-world object manipulation using pre-trained vision-language models," *arXiv preprint arXiv:2303.00905*, 2023.

[6] S. Nair, E. Mitchell, K. Chen, B. Ichter, S. Savarese, and C. Finn, "Learning language-conditioned robot behavior from offline data and crowd-sourced annotation," in *Conference on Robot Learning (CoRL)*, 2021.

[7] E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, and C. Finn, "Bc-z: Zero-shot task generalization with robotic imitation learning," in *Conference on Robot Learning*, PMLR, 2022, pp. 991–1002.

[8] M. Shridhar, L. Manuelli, and D. Fox, "Perceiver-actor: A multi-task transformer for robotic manipulation," *Conference on Robot Learning (CoRL)*, 2022.

[9] C. Lynch and P. Sermanet, "Language conditioned imitation learning over unstructured data," *arXiv preprint arXiv:2005.07648*, 2020.

[10] C. Lynch and P. Sermanet, "Language conditioned imitation learning over unstructured data," *Robotics: Science and Systems (RSS)*, 2021.

[11] C. Lynch, A. Wahid, J. Tompson, T. Ding, J. Betker, R. K. Baruch, T. Armstrong, and P. R. Florence, "Interactive language: Talking to robots in real time," *ArXiv preprint arXiv:2210.06407*, 2022.

[12] Y. Jiang, A. Gupta, Z. Zhang, G. Wang, Y. Dou, Y. Chen, L. Fei-Fei, A. Anandkumar, Y. Zhu, and L. Fan, "Vima: General robot manipulation with multimodal prompts," *arXiv preprint arXiv:2210.03094*, 2022.

[13] J. Liang, W. Huang, F. Xia, P. Xu, K. Hausman, B. Ichter, P. R. Florence, and A. Zeng, "Code as policies: Language model programs for embodied control," *ArXiv preprint arXiv:2209.07753*, 2022.

[14] B. I. et al., "Do as i can, not as i say: Grounding language in robotic affordances," in *6th Annual Conference on Robot Learning*, 2022.

[15] W. Huang, P. Abbeel, D. Pathak, and I. Mordatch, "Language models as zero-shot planners: Extracting actionable knowledge for embodied agents," in *International Conference on Machine Learning*, PMLR, 2022, pp. 9118–9147.

[16] A. Zeng, A. Wong, S. Welker, K. Choromanski, F. Tombari, A. Purohit, M. Ryoo, V. Sindhwani, J. Lee, V. Vanhoucke, *et al.*, "Socratic models: Composing zero-shot multimodal reasoning with language," *arXiv preprint arXiv:2204.00598*, 2022.

[17] W. H. et al., "Inner monologue: Embodied reasoning through planning with language models," *arXiv preprint arXiv:2207.05608*, 2022.

[18] I. Singh, V. Blukis, A. Mousavian, A. Goyal, D. Xu, J. Tremblay, D. Fox, J. Thomason, and A. Garg, "Progprompt: Generating situated robot task plans using large language models," *arXiv preprint arXiv:2209.11302*, 2022.

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

[20] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual language maps for robot navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, London, UK, 2023.

[21] S. Y. Gadre, M. Wortsman, G. Ilharco, L. Schmidt, and S. Song, *Cows on pasture: Baselines and benchmarks for language-driven zero-shot object navigation*, 2022. arXiv: `2203.10421 [cs.CV]`.

[22] B. Chen, F. Xia, B. Ichter, K. Rao, K. Gopalakrishnan, M. S. Ryoo, A. Stone, and D. Kappler, *Open-vocabulary queryable scene representations for real world planning*, 2023. eprint: `2209.09874` (cs.RO).

[23] D. Shah, B. Osiński, brian ichter, and S. Levine, "LM-nav: Robotic navigation with large pre-trained models of language, vision, and action," in *6th Annual Conference on Robot Learning*, 2022. [Online]. Available: `https://openreview.net/forum?id=UW5A3SweAH`.

[24] S. Sharma, H. Huang, K. Shivakumar, L. Y. Chen, R. Hoque, B. Ichter, and K. Goldberg, *Semantic mechanical search with large vision and language models*, 2023. arXiv: `2302.12915 [cs.RO]`.

[25] A. Rashid, S. Sharma, C. M. Kim, J. Kerr, L. Y. Chen, A. Kanazawa, and K. Goldberg, "Language embedded radiance fields for zero-shot task-oriented grasping," in *7th Annual Conference on Robot Learning*, 2023. [Online]. Available: `https://openreview.net/forum?id=k-Fg8JDQmc`.

[26] D. Chen and R. Mooney, "Learning to interpret natural language navigation instructions from observations," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 25, 2011, pp. 859–865.

[27] F. Duvallet, T. Kollar, and A. Stentz, "Imitation learning for natural language direction following through unknown environments," in *2013 IEEE International Conference on Robotics and Automation*, IEEE, 2013, pp. 1047–1053.

[28] F. Duvallet, M. R. Walter, T. Howard, S. Hemachandra, J. Oh, S. Teller, N. Roy, and A. Stentz, "Inferring maps and behaviors from natural language instructions," in *Experimental Robotics: The 14th International Symposium on Experimental Robotics*, Springer, 2016, pp. 373–388.

[29] D. Fried, R. Hu, V. Cirik, A. Rohrbach, J. Andreas, L.-P. Morency, T. Berg-Kirkpatrick, K. Saenko, D. Klein, and T. Darrell, "Speaker-follower models for vision-and-language navigation," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[30] S. Hemachandra, F. Duvallet, T. M. Howard, N. Roy, A. Stentz, and M. R. Walter, "Learning models for following natural language directions in unknown environments," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2015, pp. 5608–5615.

[31] T. M. Howard, S. Tellex, and N. Roy, "A natural language planner interface for mobile manipulators," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2014, pp. 6652–6659.

[32] T. Kollar, S. Tellex, D. Roy, and N. Roy, "Toward understanding natural language directions," in *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, IEEE, 2010, pp. 259–266.

[33] M. MacMahon, B. Stankiewicz, and B. Kuipers, "Walk the talk: Connecting language, knowledge, and action in route instructions," *Def*, vol. 2, no. 6, p. 4, 2006.

[34] C. Matuszek, D. Fox, and K. Koscher, "Following directions using statistical machine translation," in *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, IEEE, 2010, pp. 251–258.

[35] C. Matuszek, E. Herbst, L. Zettlemoyer, and D. Fox, "Learning to parse natural language commands to a robot control system," in *Experimental robotics: the 13th international symposium on experimental robotics*, Springer, 2013, pp. 403–415.

[36] H. Mei, M. Bansal, and M. Walter, "Listen, attend, and walk: Neural mapping of navigational instructions to action sequences," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, 2016.

[37] S. Tellex, T. Kollar, S. Dickerson, M. Walter, A. Banerjee, S. Teller, and N. Roy, "Understanding natural language commands for robotic navigation and mobile manipulation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 25, 2011, pp. 1507–1514.

[38] J. Thomason, S. Zhang, R. J. Mooney, and P. Stone, "Learning to interpret natural language commands through human-robot dialog," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

[39] M. Shridhar and D. Hsu, "Interactive visual grounding of referring expressions for human-robot interaction," *arXiv preprint arXiv:1806.03831*, 2018.

[40] D. K. Misra, J. Sung, K. Lee, and A. Saxena, "Tell me dave: Context-sensitive grounding of natural language to manipulation instructions," *The International Journal of Robotics Research*, vol. 35, no. 1-3, pp. 281–300, 2016.

[41] R. Paul, J. Arkin, D. Aksaray, N. Roy, and T. M. Howard, "Efficient grounding of abstract spatial concepts for natural language interaction with robot platforms," *The International Journal of Robotics Research*, vol. 37, no. 10, pp. 1269–1299, 2018.

[42] S. Patki, E. Fahnestock, T. M. Howard, and M. R. Walter, "Language-guided semantic mapping and mobile manipulation in partially observable environments," in *Conference on Robot Learning*, PMLR, 2020, pp. 1201–1210.

[43] O. Mees and W. Burgard, *Composing pick-and-place tasks by grounding language*, 2021. arXiv: `2102.08094 [cs.RO]`.

[44] M. Shridhar, L. Manuelli, and D. Fox, "Cliport: What and where pathways for robotic manipulation," *Conference on Robot Learning (CoRL)*, 2021.

[45] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, *et al.*, "Rt-1: Robotics transformer for real-world control at scale," *arXiv preprint arXiv:2212.06817*, 2022.

[46] D. K. Misra, J. Langford, and Y. Artzi, "Mapping instructions and visual observations to actions with reinforcement learning," in *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2017.

[47] Y. Jiang, S. S. Gu, K. P. Murphy, and C. Finn, "Language as an abstraction for hierarchical deep reinforcement learning," *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

[48] P. Sharma, B. Sundaralingam, V. Blukis, C. Paxton, T. Hermans, A. Torralba, J. Andreas, and D. Fox, "Correcting robot plans with natural language feedback," *Robotics: Science and Systems (RSS)*, 2022.

[49] Y. Cui, S. Karamcheti, R. Palleti, N. Shivakumar, P. Liang, and D. Sadigh, "No, to the right: Online language corrections for robotic manipulation via shared autonomy," *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2023.

[50] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "Holm: Hallucinating objects with language models for referring expression recognition in partially-observed scenes," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020.

[51] S. Peng, K. Genova, C. Jiang, A. Tagliasacchi, M. Pollefeys, T. Funkhouser, *et al.*, "Openscene: 3d scene understanding with open vocabularies," *arXiv preprint arXiv:2211.15654*, 2022.

[52] D. Driess, F. Xia, M. S. M. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu, W. Huang, Y. Chebotar, P. Sermanet, D. Duckworth, S. Levine, V. Vanhoucke, K. Hausman, M. Toussaint, K. Greff, A. Zeng, I. Mordatch, and P. Florence, "Palm-e: An embodied multimodal language model," in *arXiv preprint arXiv:2303.03378*, 2023.

[53] W. Huang, F. Xia, D. Shah, D. Driess, A. Zeng, Y. Lu, P. Florence, I. Mordatch, S. Levine, K. Hausman, *et al.*, "Grounded decoding: Guiding text generation with grounded models for robot control," *arXiv preprint arXiv:2303.00855*, 2023.

[54] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual language maps for robot navigation," *arXiv preprint arXiv:2210.05714*, 2022.

[55] G. Ghiasi, X. Gu, Y. Cui, and T.-Y. Lin, "Open-vocabulary image segmentation," *arXiv preprint arXiv:2112.12143*, 2021.

[56] B. Li, K. Q. Weinberger, S. Belongie, V. Koltun, and R. Ranftl, "Language-driven semantic segmentation," *arXiv preprint arXiv:2201.03546*, 2022.

[57] N. M. M. Shafiullah, C. Paxton, L. Pinto, S. Chintala, and A. Szlam, "Clip-fields: Weakly supervised semantic fields for robotic memory," *arXiv preprint arXiv:2210.05663*, 2022.

[58] K. M. Jatavallabhula, A. Kuwajerwala, Q. Gu, M. Omama, T. Chen, S. Li, G. Iyer, S. Saryazdi, N. Keetha, A. Tewari, *et al.*, "Conceptfusion: Open-set multimodal 3d mapping," *arXiv preprint arXiv:2302.07241*, 2023.

[59] B. Chen, F. Xia, B. Ichter, K. Rao, K. Gopalakrishnan, M. S. Ryoo, A. Stone, and D. Kappler, "Open-vocabulary queryable scene representations for real world planning," *arXiv preprint arXiv:2209.09874*, 2022.

[60] H. Ha and S. Song, "Semantic abstraction: Open-world 3D scene understanding from 2D vision-language models," in *Proceedings of the 2022 Conference on Robot Learning*, 2022.

[61] H. Chefer, S. Gur, and L. Wolf, “Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 397–406.

[62] L. Yu, Z. Lin, X. Shen, J. Yang, X. Lu, M. Bansal, and T. L. Berg, “Mattnet: Modular attention network for referring expression comprehension,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1307–1315.

[63] M. Shridhar, L. Manuelli, and D. Fox, “Cliport: What and where pathways for robotic manipulation,” in *Proceedings of the 5th Conference on Robot Learning (CoRL)*, 2021.

[64] M. Shridhar, L. Manuelli, and D. Fox, “Perceiver-actor: A multi-task transformer for robotic manipulation,” in *Proceedings of the 6th Conference on Robot Learning (CoRL)*, 2022.

[65] M. Danielczuk, A. Kurenkov, A. Balakrishna, M. Matl, D. Wang, R. Martin-Martin, A. Garg, S. Savarese, and K. Goldberg, “Mechanical search: Multi-step retrieval of a target object occluded by clutter,” in *2019 International Conference on Robotics and Automation (ICRA)*, 2019.

[66] A. Kurenkov, R. Martín-Martín, J. Ichnowski, K. Goldberg, and S. Savarese, “Semantic and geometric modeling with neural message passing in 3d scene graphs for hierarchical mechanical search,” *International Conference on Robotics and Automation (ICRA)*, 2021.

[67] M. Gupta, T. Rühr, M. Beetz, and G. S. Sukhatme, “Interactive environment exploration in clutter,” 2013, pp. 5265–5272.

[68] M. R. Dogar, M. C. Koval, A. Tallavajhula, and S. S. Srinivasa, “Object search by manipulation,” *Autonomous Robots*, vol. 36, no. 1, pp. 153–167, 2014.

[69] J. K. Li, D. Hsu, and W. S. Lee, “Act to see and see to act: Pomdp planning for objects search in clutter,” 2016, pp. 5701–5707.

[70] Y. Xiao, S. Katt, A. ten Pas, S. Chen, and C. Amato, “Online planning for target object search in clutter under partial observability,” 2019, pp. 8241–8247.

[71] W. Bejjani, W. C. Agboh, M. R. Dogar, and M. Leonetti, “Occlusion-aware search for object retrieval in clutter,” 2021, pp. 4678–4685.

[72] H. Huang, M. Danielczuk, C. M. Kim, L. Fu, Z. Tam, J. Ichnowski, A. Angelova, B. Ichter, and K. Goldberg, “Mechanical search on shelves using a novel “bluction” tool,” *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

[73] H. Huang, L. Fu, M. Danielczuk, C. M. Kim, Z. Tam, J. Ichnowski, A. Angelova, B. Ichter, and K. Goldberg, “Mechanical search on shelves with efficient stacking and destacking of objects,” *International Symposium on Robotics Research (ISRR)*, 2022.

[74] H. Huang, M. Dominguez-Kuhne, V. Satish, M. Danielczuk, K. Sanders, J. Ichnowski, A. Lee, A. Angelova, V. Vanhoucke, and K. Goldberg, "Mechanical search on shelves using lateral access x-ray," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 2045–2052.

[75] L. Y. Chen, H. Huang, M. Danielczuk, J. Ichnowski, and K. Goldberg, "Optimal shelf arrangement to minimize robot retrieval time," *IEEE International Conference on Automation Science and Engineering (CASE)*, 2022.

[76] L. E. Wixson and D. H. Ballard, "Using intermediate objects to improve the efficiency of visual search," *International Journal of Computer Vision*, vol. 12, no. 2-3, pp. 209–230, 1994.

[77] T. Kollar and N. Roy, "Utilizing object-object and object-scene context when planning to find things," in *2009 IEEE International Conference on Robotics and Automation*, IEEE, 2009, pp. 2168–2173.

[78] L. L. Wong, L. P. Kaelbling, and T. Lozano-Pérez, "Manipulation-based active search for occluded objects," in *2013 IEEE International Conference on Robotics and Automation*, IEEE, 2013, pp. 2814–2819.

[79] D. S. Chaplot, D. Gandhi, S. Gupta, A. Gupta, and R. Salakhutdinov, "Learning to explore using active neural slam," *arXiv preprint arXiv:2004.05155*, 2020.

[80] P. Chattopadhyay, J. Hoffman, R. Mottaghi, and A. Kembhavi, "Robustnav: Towards benchmarking robustness in embodied navigation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 691–15 700.

[81] D. Gordon, A. Kadian, D. Parikh, J. Hoffman, and D. Batra, "Splitnet: Sim2sim and task2task transfer for embodied visual navigation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1022–1031.

[82] L. Mezghan, S. Sukhbaatar, T. Lavril, O. Maksymets, D. Batra, P. Bojanowski, and K. Alahari, "Memory-augmented reinforcement learning for image-goal navigation," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2022, pp. 3316–3323.

[83] Y. Zhu, R. Mottaghi, E. Kolve, J. J. Lim, A. Gupta, L. Fei-Fei, and A. Farhadi, "Target-driven visual navigation in indoor scenes using deep reinforcement learning," in *2017 IEEE international conference on robotics and automation (ICRA)*, IEEE, 2017, pp. 3357–3364.

[84] Z. Al-Halah, S. K. Ramakrishnan, and K. Grauman, "Zero experience required: Plug & play modular transfer learning for semantic visual navigation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 031–17 041.

[85] M. Chang, A. Gupta, and S. Gupta, "Semantic visual navigation by watching youtube videos," *Advances in Neural Information Processing Systems*, vol. 33, pp. 4283–4294, 2020.

[86] J. K. Tsotsos, "On the relative complexity of active vs. passive visual search," *International journal of computer vision*, vol. 7, no. 2, pp. 127–141, 1992.

[87] A. Aydemir, A. Pronobis, M. Göbelbecker, and P. Jensfelt, "Active visual object search in unknown environments using uncertain semantics," *IEEE Transactions on Robotics*, vol. 29, no. 4, pp. 986–1002, 2013.

[88] S. Thrun, "Learning metric-topological maps for indoor mobile robot navigation," *Artificial Intelligence*, vol. 99, no. 1, pp. 21–71, 1998.

[89] H. J. S. Feder, J. J. Leonard, and C. M. Smith, "Adaptive mobile robot navigation and mapping," *The International Journal of Robotics Research*, vol. 18, no. 7, pp. 650–668, 1999.

[90] B. Kuipers and Y.-T. Byun, "A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations," *Robotics and autonomous systems*, vol. 8, no. 1-2, pp. 47–63, 1991.

[91] B. H. Wilcox, "Robotic vehicles for planetary exploration," *Applied Intelligence*, vol. 2, pp. 181–193, 1992.

[92] P. Anderson, Q. Wu, D. Teney, J. Bruce, M. Johnson, N. Sünderhauf, I. Reid, S. Gould, and A. Van Den Hengel, "Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3674–3683.

[93] D. S. Chaplot, D. P. Gandhi, A. Gupta, and R. R. Salakhutdinov, "Object goal navigation using goal-oriented semantic exploration," *Advances in Neural Information Processing Systems*, vol. 33, pp. 4247–4258, 2020.

[94] M. Deitke, E. VanderBilt, A. Herrasti, L. Weihs, K. Ehsani, J. Salvador, W. Han, E. Kolve, A. Kembhavi, and R. Mottaghi, "Procthor: Large-scale embodied ai using procedural generation," *Advances in Neural Information Processing Systems*, vol. 35, pp. 5982–5994, 2022.

[95] Y. Liang, B. Chen, and S. Song, "Sscnav: Confidence-aware semantic scene completion for visual semantic navigation," in *2021 IEEE international conference on robotics and automation (ICRA)*, IEEE, 2021, pp. 13 194–13 200.

[96] S. Wani, S. Patel, U. Jain, A. Chang, and M. Savva, "Multion: Benchmarking semantic map memory using multi-object navigation," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9700–9712, 2020.

[97] M. Wortsman, K. Ehsani, M. Rastegari, A. Farhadi, and R. Mottaghi, "Learning to learn how to learn: Self-adaptive visual navigation using meta-learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 6750–6759.

[98] W. Yang, X. Wang, A. Farhadi, A. Gupta, and R. Mottaghi, "Visual semantic navigation using scene priors," *arXiv preprint arXiv:1810.06543*, 2018.

[99] M. Samadi, T. Kollar, and M. Veloso, "Using the web to interactively learn to find objects," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 26, 2012, pp. 2074–2080.

[100] H. Ha and S. Song, "Semantic abstraction: Open-world 3d scene understanding from 2d vision-language models," in *6th Annual Conference on Robot Learning*, 2022.

[101] J. Kerr, C. M. Kim, K. Goldberg, A. Kanazawa, and M. Tancik, "Lerf: Language embedded radiance fields," *arXiv preprint arXiv:2303.09553*, 2023.

[102] M. Minderer, A. Gritsenko, A. Stone, M. Neumann, D. Weissenborn, A. Dosovitskiy, A. Mahendran, A. Arnab, M. Dehghani, Z. Shen, *et al.*, "Simple open-vocabulary object detection with vision transformers," *arXiv preprint arXiv:2205.06230*, 2022.

[103] H. Chefer, S. Gur, and L. Wolf, "Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2021, pp. 397–406.

[104] D. Song, C. H. Ek, K. Huebner, and D. Kragic, "Task-based robot grasp planning using probabilistic inference," *IEEE transactions on robotics*, vol. 31, no. 3, pp. 546–561, 2015.

[105] A. Myers, C. L. Teo, C. Fermüller, and Y. Aloimonos, "Affordance detection of tool parts from geometric features," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2015, pp. 1374–1381.

[106] F.-J. Chu, R. Xu, and P. A. Vela, "Learning affordance segmentation for real-world robotic manipulation via synthetic images," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1140–1147, 2019. DOI: `10.1109/LRA.2019.2894439`.

[107] N. Vahrenkamp, L. Westkamp, N. Yamanobe, E. E. Aksoy, and T. Asfour, "Part-based grasp planning for familiar objects," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, 2016, pp. 919–925. DOI: `10.1109/HUMANOIDS.2016.7803382`.

[108] Y. Song, P. Sun, Y. Ren, Y. Zheng, and Y. Zhang, "Learning 6-dof fine-grained grasp detection based on part affordance grounding," *arXiv preprint arXiv:2301.11564*, 2023.

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

[110] M. Kokic, D. Kragic, and J. Bohg, "Learning task-oriented grasping from human activity datasets," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3352–3359, 2020.

[111] D. S. Hochbaum and V. Singh, "An efficient algorithm for co-segmentation," in *2009 IEEE 12th International Conference on Computer Vision*, IEEE, 2009, pp. 269–276.

[112] Y. Chai, V. Lempitsky, and A. Zisserman, "Bicos: A bi-level co-segmentation method for image classification," in *2011 International conference on computer vision*, IEEE, 2011, pp. 2579–2586.

[113] S. Amir, Y. Gandelsman, S. Bagon, and T. Dekel, "Deep vit features as dense visual descriptors," *ECCVW What is Motion For?*, 2022.

[114] P. Sun, S. Chen, C. Zhu, F. Xiao, P. Luo, S. Xie, and Z. Yan, "Going denser with open-vocabulary part segmentation," *arXiv preprint arXiv:2305.11173*, 2023.

[115] G. Li, V. Jampani, D. Sun, and L. Sevilla-Lara, "Locate: Localize and transfer object parts for weakly supervised affordance grounding," in *Proceedings of the IEEE International Conference on Computer Vision*, 2023.

[116] X. Gu, T.-Y. Lin, W. Kuo, and Y. Cui, *Open-vocabulary object detection via vision and language knowledge distillation*, 2022. arXiv: `2104.13921 [cs.CV]`.

[117] F. Liang, B. Wu, X. Dai, K. Li, Y. Zhao, H. Zhang, P. Zhang, P. Vajda, and D. Marculescu, *Open-vocabulary semantic segmentation with mask-adapted clip*, 2023. arXiv: `2210.04150 [cs.CV]`.

[118] X. Zhou, R. Girdhar, A. Joulin, P. Krähenbühl, and I. Misra, *Detecting twenty-thousand classes using image-level supervision*, 2022. arXiv: `2201.02605 [cs.CV]`.

[119] J. Wei, X. Wang, D. Schuurmans, M. Bosma, E. Chi, Q. Le, and D. Zhou, "Chain of thought prompting elicits reasoning in large language models," *arXiv preprint arXiv:2201.11903*, 2022.

[120] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation CIRA'97.'Towards New Computational Principles for Robotics and Automation'*, IEEE, 1997, pp. 146–151.

[121] J. Maja, "Integration of representation into goal-driven behavior-based robots," *IEEE transactions on robotics and automation*, vol. 8, no. 3, pp. 304–312, 1992.

[122] S. Thrun and A. Bücken, "Integrating grid-based and topological maps for mobile robot navigation," in *Proceedings of the national conference on artificial intelligence*, 1996, pp. 944–951.

[123] B. Yamauchi and R. Beer, "Spatial learning for navigation in dynamic environments," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 26, no. 3, pp. 496–505, 1996.

[124] X. Gu, T.-Y. Lin, W. Kuo, and Y. Cui, "Open-vocabulary object detection via vision and language knowledge distillation," *International Conference on Learning Representations (ICLR)*, 2021.

[125] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," *arXiv:2304.02643*, 2023.

[126] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," *ArXiv*, vol. abs/2301.12597, 2023.

[127] *Keras ocr*, `https://support.google.com/merchants/answer/6324436?hl=en`.

[128] A. N. et al., "Text and code embeddings by contrastive pre-training," *arXiv preprint arXiv:2201.10005*, 2022.

[129] J. Lin, "Divergence measures based on the shannon entropy," *IEEE Transactions on Information Theory*, vol. 37, no. 1, pp. 145–151, 1991. DOI: `10.1109/18.61115`.

[130] G. M. Center, *Google product category*, `https://support.google.com/merchants/answer/6324436?hl=en`, Accessed: 2023-01-31.

[131] K. Kleeberger and M. F. Huber, *Single shot 6d object pose estimation*, 2020. arXiv: `2004.12729 [cs.CV]`.

[132] K. Bousmalis, A. Irpan, P. Wohlhart, Y. Bai, M. Kelcey, M. Kalakrishnan, L. Downs, J. Ibarz, P. Pastor, K. Konolige, S. Levine, and V. Vanhoucke, *Using simulation and domain adaptation to improve efficiency of deep robotic grasping*, 2017. arXiv: `1709.07857 [cs.LG]`.

[133] S. Song, A. Zeng, J. Lee, and T. Funkhouser, "Grasping in the wild: Learning 6dof closed-loop grasping from low-cost demonstrations," *Robotics and Automation Letters*, 2020.

[134] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, *Domain randomization for transferring deep neural networks from simulation to the real world*, 2017. arXiv: `1703.06907 [cs.RO]`.

[135] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, and S. Levine, *Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation*, 2018. arXiv: `1806.10293 [cs.LG]`.

[136] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 421–436, 2018. DOI: `10.1177/0278364917710318`.

[137] A. Mousavian, C. Eppner, and D. Fox, *6-dof graspnet: Variational grasp generation for object manipulation*, 2019. arXiv: `1905.10520 [cs.CV]`.

[138] J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, and K. Goldberg, "Learning ambidextrous robot grasping policies," *Science Robotics*, vol. 4, no. 26, eaau4984, 2019.

[139] L. Pinto and A. Gupta, "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours," in *2016 IEEE international conference on robotics and automation (ICRA)*, IEEE, 2016, pp. 3406–3413.

[140] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, *Microsoft coco: Common objects in context*, 2015. arXiv: `1405.0312 [cs.CV]`.

[141] A. Depierre, E. Dellandréa, and L. Chen, "Jacquard: A large scale dataset for robotic grasp detection," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2018, pp. 3511–3516.

[142] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.

[143] B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar, "The ycb object and model set: Towards common benchmarks for manipulation research," in *2015 international conference on advanced robotics (ICAR)*, IEEE, 2015, pp. 510–517.

[144] H.-S. Fang, C. Wang, M. Gou, and C. Lu, "Graspnet-1billion: A large-scale benchmark for general object grasping," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 444–11 453.

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

[146] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, *Learning transferable visual models from natural language supervision*, 2021. arXiv: `2103.00020 [cs.CV]`.

[147] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 9650–9660.

[148] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," in *European Conference on Computer Vision*, Springer, 2020, pp. 405–421.

[149] M. Adamkiewicz, T. Chen, A. Caccavale, R. Gardner, P. Culbertson, J. Bohg, and M. Schwager, "Vision-only robot navigation in a neural radiance world," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4606–4613, 2022.

[150] J. T. Barron, B. Mildenhall, M. Tancik, P. Hedman, R. Martin-Brualla, and P. P. Srinivasan, "Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5855–5864.

[151] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Mip-nerf 360: Unbounded anti-aliased neural radiance fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 5470–5479.

[152] L. Ma, X. Li, J. Liao, Q. Zhang, X. Wang, J. Wang, and P. V. Sander, "Deblur-nerf: Neural radiance fields from blurry images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 861–12 870.

[153] X. Huang, Q. Zhang, Y. Feng, H. Li, X. Wang, and Q. Wang, "Hdr-nerf: High dynamic range neural radiance fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 398–18 408.

[154] S. Sabour, S. Vora, D. Duckworth, I. Krasin, D. J. Fleet, and A. Tagliasacchi, "Robustnerf: Ignoring distractors with robust losses," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 20 626–20 636.

[155] J. Philip and V. Deschaintre, "Radiance field gradient scaling for unbiased near-camera training," *arXiv preprint arXiv:2305.02756*, 2023.

[156] M. Tancik, E. Weber, E. Ng, R. Li, B. Yi, J. Kerr, T. Wang, A. Kristoffersen, J. Austin, K. Salahi, *et al.*, "Nerfstudio: A modular framework for neural radiance field development," *arXiv preprint arXiv:2302.04264*, 2023.

[157] P. Wang, Y. Liu, Z. Chen, L. Liu, Z. Liu, T. Komura, C. Theobalt, and W. Wang, "F2-nerf: Fast neural radiance field training with free camera trajectories," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4150–4159.

[158] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Zip-nerf: Anti-aliased grid-based neural radiance fields," *arXiv preprint arXiv:2304.06706*, 2023.

[159] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Transactions on Graphics (ToG)*, vol. 41, no. 4, pp. 1–15, 2022.

[160] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su, "Tensorf: Tensorial radiance fields," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXII*, Springer, 2022, pp. 333–350.

[161] S. Fridovich-Keil, G. Meanti, F. R. Warburg, B. Recht, and A. Kanazawa, "K-planes: Explicit radiance fields in space, time, and appearance," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 479–12 488.

[162] A. Yu, S. Fridovich-Keil, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa, "Plenoxels: Radiance fields without neural networks," *arXiv preprint arXiv:2112.05131*, 2021.

[163] K. Park, U. Sinha, P. Hedman, J. T. Barron, S. Bouaziz, D. B. Goldman, R. Martin-Brualla, and S. M. Seitz, "Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields," *arXiv preprint arXiv:2106.13228*, 2021.

[164] Z. Li, Q. Wang, F. Cole, R. Tucker, and N. Snavely, "Dynibar: Neural dynamic image-based rendering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4273–4284.

[165] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer, "D-NeRF: Neural Radiance Fields for Dynamic Scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[166] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M. R. Oswald, and M. Pollefeys, "Nice-slam: Neural implicit scalable encoding for slam," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2022, pp. 12 786–12 796.

[167] E. Sucar, S. Liu, J. Ortiz, and A. J. Davison, "Imap: Implicit mapping and positioning in real-time," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2021, pp. 6229–6238.

[168] A. Rosinol, J. J. Leonard, and L. Carlone, *Nerf-slam: Real-time dense monocular slam with neural radiance fields*, 2022. arXiv: `2210.13641 [cs.CV]`.

[169] Y. Li, S. Li, V. Sitzmann, P. Agrawal, and A. Torralba, "3d neural scene representations for visuomotor control," in *Conference on Robot Learning*, PMLR, 2022, pp. 112–123.

[170] D. Driess, I. Schubert, P. Florence, Y. Li, and M. Toussaint, "Reinforcement learning with neural radiance fields," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[171] J. Kerr, L. Fu, H. Huang, Y. Avigal, M. Tancik, J. Ichnowski, A. Kanazawa, and K. Goldberg, "Evo-neRF: Evolving neRF for sequential robot grasping of transparent objects," in *6th Annual Conference on Robot Learning*, 2022. [Online]. Available: `https://openreview.net/forum?id=Bxr45keYrf`.

[172] J. Ichnowski*, Y. Avigal*, J. Kerr, and K. Goldberg, "Dex-NeRF: Using a neural radiance field to grasp transparent objects," in *Conference on Robot Learning (CoRL)*, 2020.

[173] A. Byravan, J. Humplik, L. Hasenclever, A. Brussee, F. Nori, T. Haarnoja, B. Moran, S. Bohez, F. Sadeghi, B. Vujatovic, and N. Heess, *Nerf2real: Sim2real transfer of vision-guided bipedal motion skills using neural radiance fields*, 2022. arXiv: `2210.04932 [cs.RO]`.

[174] S. Zhi, T. Laidlow, S. Leutenegger, and A. J. Davison, "In-place scene labelling and understanding with implicit scene representation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 838–15 847.

[175] Y. Siddiqui, L. Porzi, S. R. Buló, N. Müller, M. Nießner, A. Dai, and P. Kontschieder, *Panoptic lifting for 3d scene understanding with neural fields*, 2022. eprint: `arXiv:2212.09802`.

[176] A. Kundu, K. Genova, X. Yin, A. Fathi, C. Pantofaru, L. Guibas, A. Tagliasacchi, F. Dellaert, and T. Funkhouser, "Panoptic Neural Fields: A Semantic Object-Aware Neural Scene Representation," in *CVPR*, 2022.

[177] S. Kobayashi, E. Matsumoto, and V. Sitzmann, "Decomposing nerf for editing via feature field distillation," in *NeurIPS*, vol. 35, 2022. [Online]. Available: `https://arxiv.org/pdf/2205.15585.pdf`.

[178] V. Tschernezki, I. Laina, D. Larlus, and A. Vedaldi, "Neural feature fusion fields: 3d distillation of self-supervised 2d image representations," in *Proceedings of the International Conference on 3D Vision (3DV)*, 2022.

[179] S. F. Bhat, R. Birkl, D. Wofk, P. Wonka, and M. Müller, "Zoedepth: Zero-shot transfer by combining relative and metric depth," *arXiv preprint arXiv:2302.12288*, 2023.

[180] Guangcong, Z. Chen, C. C. Loy, and Z. Liu, "Sparsenerf: Distilling depth ranking for few-shot novel view synthesis," *Technical Report*, 2023.

[181] M. Niemeyer, J. T. Barron, B. Mildenhall, M. S. M. Sajjadi, A. Geiger, and N. Radwan, *Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs*, 2021. arXiv: `2112.00724 [cs.CV]`.

[182] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, *et al.*, "Dinov2: Learning robust visual features without supervision," *arXiv preprint arXiv:2304.07193*, 2023.

[183] N. Liu, S. Li, Y. Du, A. Torralba, and J. B. Tenenbaum, "Compositional visual generation with composable diffusion models," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVII*, Springer, 2022, pp. 423–439.

[184] K. Jatavallabhula, A. Kuwajerwala, Q. Gu, M. Omama, T. Chen, S. Li, G. Iyer, S. Saryazdi, N. Keetha, A. Tewari, J. Tenenbaum, C. de Melo, M. Krishna, L. Paull, F. Shkurti, and A. Torralba, "Conceptfusion: Open-set multimodal 3d mapping," *arXiv*, 2023.

[185] OpenAI, *Gpt-4 technical report*, 2023. arXiv: `2303.08774 [cs.CL]`.

[186] H. Liu, C. Li, Q. Wu, and Y. J. Lee, *Visual instruction tuning*, 2023. arXiv: `2304.08485 [cs.CV]`.

[187] OpenAI, *Gpt-4v(ision) system card*, 2023.

[188] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics*, vol. 42, no. 4, Jul. 2023. [Online]. Available: `https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/`.

[189] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, P. Florence, C. Fu, M. G. Arenas, K. Gopalakrishnan, K. Han, K. Hausman, A. Herzog, J. Hsu, B. Ichter, A. Irpan, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, L. Lee, T.-W. E. Lee, S. Levine, Y. Lu, H. Michalewski, I. Mordatch, K. Pertsch, K. Rao, K. Reymann, M. Ryoo, G. Salazar, P. Sanketi, P. Sermanet, J. Singh, A. Singh, R. Soricut, H. Tran, V. Vanhoucke, Q. Vuong, A. Wahid, S. Welker, P. Wohlhart, J. Wu, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich, *Rt-2: Vision-language-action models transfer web knowledge to robotic control*, 2023. arXiv: `2307.15818 [cs.RO]`.

[190] E. Collaboration, A. O'Neill, A. Rehman, *et al.*, *Open x-embodiment: Robotic learning datasets and rt-x models*, 2024. arXiv: `2310.08864 [cs.RO]`.

[191] T. Brooks, B. Peebles, C. Holmes, W. DePue, Y. Guo, L. Jing, D. Schnurr, J. Taylor, T. Luhman, E. Luhman, C. Ng, R. Wang, and A. Ramesh, "Video generation models as world simulators," 2024. [Online]. Available: `https://openai.com/research/video-generation-models-as-world-simulators`.