Open-World Semantic Mechanical Search with Large Vision and Language Models

Anonymous Author(s)
Affiliation
Address
email

Abstract: Finding occluded or out-of-view objects, known as *mechanical search*, is a common yet challenging problem in robotics. Manipulation or navigation is often required to reveal the target object. A key to solving this problem is semantic understanding, which requires reasoning about object relationships. Large pretrained vision and language models (VLMs and LLMs) have shown promise in generalizing to uncommon objects and unseen real-world environments. In this work, we propose a novel framework called Open-World Semantic Mechanical Search (OWSMS). OWSMS detects objects without the need for object lists, reasons about semantic relationships explicitly using LLMs, and generates a semantic occupancy distribution for the target object. Compared to methods that rely on visual and semantic similarities offered by CLIP embeddings, OWSMS 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, OWSMS can serve as a plug-in semantic module for downstream manipulation or navigation policies in mechanical search tasks. We also study the different design considerations required for constrained environments such as shelves with known object lists and for open-world unconstrained environments without object lists. For mechanical search on shelves, OWSMS improves a geometric-based planner by 24% across three domains in simulation and 47.1% in physical experiments. For unconstrained environments, results suggest that our semantic distributions outperform those generated by CLIP-based methods. Videos and supplements are here.

1 Introduction

Mechanical search, where a robot manipulates and navigates to find an occluded or out-of-view target object [1, 2], is a challenging robotics problem. Prior work has shown success in revealing the desired object by manipulating the occluding objects [3, 4, 5], obtaining new observations through rotating the camera [6], or navigating to new locations [7, 8]. However, generalization to unseen environments remains challenging due to numerous long-tail objects present in the real world.

A key to this problem is semantic scene understanding to reason about object relationships and possible target object locations. As real-world environments are often organized semantically, when looking for a band-aid, knowing that it is more likely to be near other personal care items inside a bathroom cabinet than on an office desk, can significantly accelerate search.

Large vision and language models (VLMs and LLMs) show promise, as they are pretrained on internet-scale data and capture semantic information. A large body of prior work has shown that these models can provide good visual representations [9, 10, 11, 12, 13], ground language instructions [14, 15, 16, 17, 18, 19, 20], and serve as planners out of the box [21, 22, 23, 24, 25, 26]. CLIP [27] is a commonly-used interface to associate vision and language, and many works [7, 8, 28, 29] use it to build semantic scene representations and show improved performance on object query and navigation tasks. However, while informative, the dot product of CLIP text and image

Submitted to the 7th Conference on Robot Learning (CoRL 2023). Do not distribute.

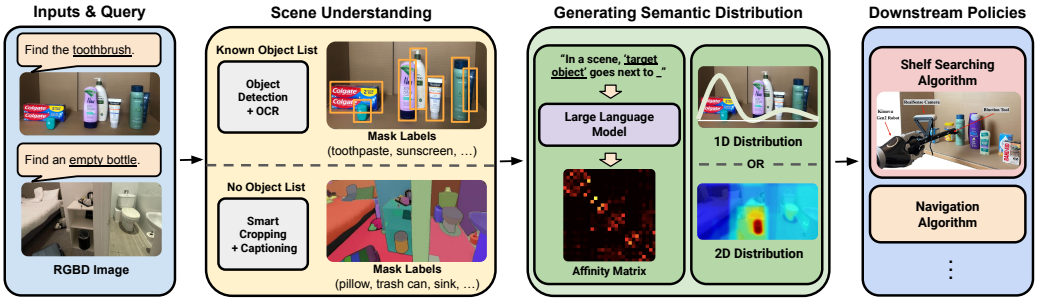

Figure 1: **Overview of Open-World Semantic Mechanical Search (OWSMS).** OWSMS first receives a scene image and a desired target object. It then applies an object detection, or segmentation algorithm combined with captioning as necessary when object lists are unavailable. OWSMS then uses a large language model to compute affinities between detected objects to the target object, and it uses these affinities to output a semantic occupancy distribution of the appropriate dimension for the downstream problem. This distribution indicates the likelihood of the physical presence of objects which can be used for downstream policies such as shelf searching or navigation in pursuit of the target object.

embeddings lacks deep reasoning capabilities and sometimes behaves as a bag-of-words [30]. As such, CLIP is most useful for localizing objects that are already visible somewhere in the scene or map [13, 8], a property that many open-vocabulary object detectors build on [31, 32, 33].

LLMs demonstrate advanced reasoning and planning capabilities [34]. Many prior works [25, 35, 7] 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 [3, 4, 5]. Likewise, for object navigation, prior research has explored exploration and navigation strategies that are independent of semantic understanding [36, 37, 38, 39]. As such, decoupling semantic reasoning and geometric planning may allow flexible integration with task-specific modules for various downstream settings.

In this work, we seek the best of both worlds. We propose Open-World Semantic Mechanical Search (OWSMS), which generates an explicit intermediate representation, a *semantic occupancy distribution*, that can serve as a plug-in semantic module for existing search algorithms. Given a scene, OWSMS uses object detection and image captioning to obtain mask labels, explicitly extracts semantic relationships between objects in the scene and the target object using an LLM, and uses these derived "affinities" to build a semantic distribution. We essentially perform mechanical search in the LLM feature space so as to not burden VLMs with both object detection and search as current works do. This decoupling leads to better results as the language feature space is better able to capture semantic relations. We show that OWSMS outperforms CLIP-based methods and can be easily integrated with a geometric shelf searching algorithm [3] to improve performance.

We also study the different design considerations required for constrained environments such as pharmacy shelves and kitchen cabinets with known object lists and for open-world unconstrained environments without object lists. When object lists are available, OWSMS uses an open vocabulary object detection model [40] refined with Optical Character Recognition (OCR), to identify objects. When object lists are unavailable, OWSMS combines segmentation [41] and image captioning [42] to generate object mask descriptions. We find that this gives the best performance.

Together, our contributions are:

1. OWSMS, a novel zero-shot approach using pretrained VLMs and LLMs for synthesizing semantic occupancy distributions for use in downstream search policies;

2. An empirically evaluated pipeline for generating object labels in open-world scenes without access to any object lists;

3. Experiments for mechanical search on shelves show that OWSMS improves a geometric-based planner by 24% across three domains in simulation and 47% in physical experiments;
4. For unconstrained open-world environments, we show that we outperform CLIP in creating semantic distributions for sampled environments in real homes, offices, and buildings.

2 Related Work and Preliminaries

2.1 Mechanical Search

Mechanical search [1, 2] 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 [1] and shelf environments [43, 44, 45, 46, 47] 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 [1, 4, 5, 3, 48]. Kurenkov et al. [2] study mechanical search through mobile manipulation in larger home environments, proposing 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, we harness large pretrained models to extract open-vocabulary semantic information zero-shot.

There are many types of navigation tasks, such as point goals [49, 50, 51], image goals [52, 53], and object goals [54, 55]. Finding out-of-view objects is an object goal navigation task. Classical geometry-based methods typically first build a map [56, 57] and then perform planning [58, 59]. Learning-based methods typically use reinforcement learning trained in simulation [54, 60, 50, 61, 62, 63, 64, 65] or YouTube videos [55] to learn semantics and efficient exploration strategies. Recently, many works have explored using LLMs and VLMs out-of-the-box for semantic scene understanding and zero-shot object navigation, which this work belongs to. The most common strategy is to use CLIP features [27] obtained from pretrained open-vocab detectors [66, 67] as in VL-Maps [7] and OpenScene [68] or from region proposals models as in CLIP-Fields [69], ConceptFusion [70], and NLMaps-SayCan [71] and fuse them into 3D point clouds or implicit representations [30]. The constructed representations can then be used for open-vocabulary target queries to locate the object and perform navigation. Gadre et al. [8] 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 [72] works best, followed by patchifying images to obtain separate CLIP embeddings and compute similarity with text embeddings. In this work, instead of using the similarity of CLIP embeddings for object localization, we use the LLM feature space to explicitly reason about the object's semantic relationships and show that OWSMS outperforms these two methods.

2.2 Natural Language in Robotics

Natural language is increasingly used as an interface for robot task specification [73]. Examples include language-conditioned imitation learning [74, 19, 16, 18, 20, 75], language-conditioned reinforcement learning [76, 77, 14], and online correction of robot policies through language feedback [78, 79]. In particular, pretrained image encoders and open-vocabulary object detectors have enabled generalization to novel object queries at test time [25, 24, 13]. In this work, we also take in novel object targets specified using natural language, but 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. Our method shares similarity with HOLM [6], 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 [7, 68, 6], propose a pipeline for generating object labels without access to any object lists, and evaluate our method in real-world environments.

Many studies have also used LLMs as a planner by letting them break down tasks through step-by-step reasoning [22, 23, 24, 25, 35] or directly write code [21, 26]. These end-to-end planning paradigms benefit from the deep reasoning abilities of LLMs, but incorporating additional non-

language information and integration with domain-specific policies is not straightforward. 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. We propose decoupling semantic reasoning and geometric planning; rather than directly output primitive instructions from image observations, OWSMS distills LLMs' semantic reasoning from its feature space into a semantic distribution that specialized planning and manipulation policies can use.

2.3 Occupancy Distribution

Occupancy distributions indicates the probability of each pixel in the image containing the target object's amodal segmentation mask [1]. Prior work [1, 3, 4, 5] 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. Huang et al. [4] propose the LAX-RAY system, which uses a neural network to predict the spatial occupancy distribution. A greedy policy called Distribution Area Reduction (DAR) uses this distribution to greedily reduce the overlap between objects and the distribution the most. OWSMS generates the occupancy distribution using semantic information, which can then be combined with the LAX-RAY spatial distribution for downstream search.

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. 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 $y_t \in \mathcal{Y} = \mathcal{R}^{H \times W \times 4}$ are RGBD images from a robot-mounted RGBD camera at timestep t. Given the name of the target object and the observation y_t, the goal is to generate a dense occupancy distribution that encodes semantic affinities (with respect to the target object) and is useful for downstream tasks.

4 Method

We propose OWSMS, 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. 1 visualizes the pipeline. OWSMS 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. By and large, OWSMS can be applied to two common situations: 1) unconstrained environments where objects in the scene are previously unseen and 2) constrained environments where all objects in the scene are a subset of a known list. A known object list is helpful, but the assumption must be relaxed to feasibly do mechanical search in an unconstrained environment.

4.1 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 [40], to obtain object segmentation masks and labels from an RGB image. We also find using Keras OCR [80] can improve the quality of the ViLD object detection, by increasing the mean average precision (mAP) on 100 pharmacy scene images from 2.4 to 28.9. Details are in Appendix.

Crop Generation + Image Captioning When an object list is not available, many open vocabulary detectors such as ViLD cannot be used. We instead create smart crops and use an image captioning model, specifically BLIP-2 [42], to convert object crops to their text descriptions, which are analogous to having object class labels. Since BLIP-2 can do VQA, we specifically ask for the dominant objects in each crop as it generates less noisy captions. We generate crops that are both object-centric (using Segment-Anything (SAM) [41]) for better object boundaries in the semantic distribution and general multiscale, overlapping crops which help encode large-scale semantic information.

4.2 Creating the Semantic Distribution

We consider two ways to use a language model to generate affinity scores for the semantic distribution. **(1) OWSMS-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*}". The structure of this prompt directly represents the probability of the target object given we see the label. Since the label is also 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 a given location. We approximate the probability of the target object at each location to be proportional to the affinity score between the target and the nearest object at this location. To account for noise, we apply spatial smoothing using a Gaussian kernel with std σ. More details are in Appendix. **(2) OWSMS-E:** An alternative method we explore is to use a language embedding model (e.g. OpenAI Embedding Model [81]) and get embeddings for all labels and the target object, then 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 4.1 can contain many incorrect or hallucinated labels, making the distribution noisy. To mitigate this, we propose to use CLIP to verify the captions. 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.

4.3 Combining with Downstream Policies

Constrained Environments We consider semantically organized shelves as the constrained environment. 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. Additional details are in the appendix. We use OWSMS as the plug-in semantic module for the existing search algorithm LAX-RAY from [4] by multiplying the semantic occupancy distribution with the learned spatial distribution which indicates where a target object could be hiding based on geometry. We then use the DAR policy [4] to perform mechanical search. Since the search in constrained environments requires the manipulating other objects, once the search begins, the shelf may become semantically disorganized. As such, at each step in a rollout, OWSMS uses only the object locations where it was first discovered.

Unconstrained Environments We consider large room spaces, with semantic diversity (rooms of a house, 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 spaced-out 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 result from not having an object list. Then, after selecting the view, we again select the highest affinity point and move to that location. The agent is teleoperated for this experiment alone.

5 Experiments

5.1 Experiments for Constrained Environments

5.1.1 Semantic Scene Generation

For constrained environments, we evaluate both in simulation and on real shelves. We consider three object domains in simulation: a pharmacy, an industrial kitchen, and an office, and the pharmacy

domain in real. We select 27, 24, and 40 representative objects respectively in these domains from the Google Product Taxonomy [82]. Details about scene generation from Google Product Taxonomy and the full lists of objects are in the Appendix. Both simulation and real experiments take place in a $0.8\,\text{m} \times 0.35\,\text{m} \times 0.57\,\text{m}$ shelf. We note that, while we use Google Taxonomy as an approximation of the ground truth semantic distribution for producing realistic semantically-arranged scenes and for evaluation purposes, the taxonomy has a limited category and cannot be used directly for open-world scenes. In contrast, OWSMS can handle long-tail objects and arbitrary visual modifiers.

5.1.2 Evaluation of Semantic Distributions

Metric	CLIP	OWSMS-E	OWSMS-LLM
$\Delta\%$ JSD ↑	20.0%	33.8%	44.6%

Table 1: Percentage improvement of the generated semantic distributions for various methods compared with a uniform prior, measured based on the Jensen-Shannon Distance (JSD) from the ground truth distribution. OWSMS outperforms CLIP.

We evaluate the quality of the generated semantic distribution for OWSMS compared with a CLIP baseline proposed by CoW [8]. Specifically, we create the semantic distribution by taking the dot product between the CLIP image embeddings for each mask and the CLIP text embedding of the target object. To evaluate the quality of the generated semantic distribution, we compute the reduction of Jensen-Shannon Distance (JSD) [83] between the generated distribution and the ground truth distribution compared to a uniform prior. From Table 1, OWSMS significantly outperforms CLIP embeddings, with the OWSMS-LLM variant outperforming OWSMS-E. This suggests the reasoning capability of LLMs is valuable for capturing semantics beyond CLIP-embedding similarities.

5.1.3 Simulation Object Retrieval Experiments

We run an extensive suite of experiments using the First Order Shelf Simulator (FOSS) from [3]. In simulation experiments, we assume perfect object detection but consider geometry for occlusion. Details can be found in the Appendix. For each domain, we generate scenes with the procedure described above with various numbers of objects N=12, 15, 18, 21 with 200 scenes for each. Termination occurs when the target object becomes visible or reaching maximum action number $2N$.

For each scene, we evaluate whether OWSMS improves the performance of LAX-RAY [3]. We consider both OWSMS-E and OWSMS-LLM for augmenting the geometric distribution from LAX-RAY. We report the following metrics: **Successes:** 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	$\Delta\%$	Successes	# Actions	$\Delta\%$	Successes	# Actions	$\Delta\%$
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
OWSMS-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
OWSMS-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 2: Simulation Experiment Results averaged over 12, 15, 18, 21 number of objects, also reported with $\Delta\%$, the percentage reduction in actions from Spatial NN.

We report results for all numbers of objects N in the Appendix and the results averaged across all values of N in Table 2. In all domains, OWSMS-LLM and OWSMS-E improve LAX-RAY performances with higher success rates and fewer search actions. In the pharmacy and office domain, OWSMS-LLM outperforms OWSMS-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 which do not have a clear semantic categorization, making semantic prior less effective.

Overall, the results suggest that OWSMS-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.

5.1.4 Physical Object Retrieval Experiments

We use the Kinova Gen2 robot fitted with a 3D-printed blade and suction tool [4] (see Fig. 1). We use an Intel RealSense depth camera mounted on the tool to provide RGBD observations. We conduct physical experiments on the pharmacy domain. 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. More details are in Appendix. As simulation results from Table 2 suggest OWSMS-LLM outperforms OWSMS-E, we evaluate LAX-RAY augmented with OWSMS-LLM in physical experiments. An identical set of 12 semantically arranged scenes (starting configurations) is used for each method.

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

Table 3: 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.

Results are shown in Table 3. We observe that OWSMS significantly accelerates mechanical search on shelves, reducing the average number of actions by 47.1%. In physical experiments, the noise in depth image results in worse spatial distribution than in simulation, making the semantic distribution more critical in identifying where a target object may lie in physical experiments. Thus, OWSMS-LLM outperforms the spatial distribution by a larger margin, 47.1%, in physical experiments compared to 32.5% in the simulated pharmacy domain.

5.1.5 Ablation on Object Lists

Metric	Uniform Dist.	OWSMS-LLM w/o Object List	OWSMS-LLM w/ Object List
JSD $\downarrow$	0.554 ± 0.006	0.421 ± 0.032	0.382 ± 0.036

Table 4: We measure the deviation in the semantic distribution generated by these methods and the ground truth using JSD. OWSMS with the object list outperforms OWSMS without the object list.

To evaluate the benefit of object lists, we compare the performance of our method on shelves with and without access to the object list by computing the Jensen-Shannon Distance (JSD) [83] between the generated distribution and the ground truth distribution on the 12 physical shelves as in Section 5.1.4. From Table 4, we see that OWSMS achieves a better semantic distribution compared to a uniform prior in both cases, but knowing the object list improves results. This is not surprising since knowing an object list reduces the noise in the scene understanding and leads to a higher quality of the semantic distribution.

5.2 Experiments for Unconstrained Environments

Method	OWL-ViT	CLIP	OWSMS-LLM	OWSMS-E
IoU	0.106 ± 0.031	0.218 ± 0.030	0.385 ± 0.032	$\mathbf{0.431} \pm 0.032$

Table 5: IoU results for different methods. The object detector, OWL-ViT, performed poorly as the target object are semantically related but have very little visual similarity. CLIP performs worse than OWSMS because OWSMS is getting semantic similarity in text-only latent space which can capture more nuance than the visual-language embedding space.

For the unconstrained environment, we perform two experiments. We first evaluate the semantic distribution generation on a static image dataset consisting of 10 real scenes taken from a hotel, an office building, and two local grocery stores. For each scene, we sample 5 target objects. We annotate the ground truth search area based on the real scene to quantitatively evaluate the accuracy of each method as measured by IOU. We evaluate the following baselines: CLIP and OWL-ViT, which are the two best-performing methods found by Gadre et al. [8]. For CLIP, it uses the same crop-label pairs as OWSMS-LLM, and OWSMS-E to generate a semantic distribution as described in Section 5.1.2. We then threshold this distribution and create a mask so as to calculate IOU with the ground truth. On the other hand, OWL-ViT gives bounding boxes for its labels, so 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.

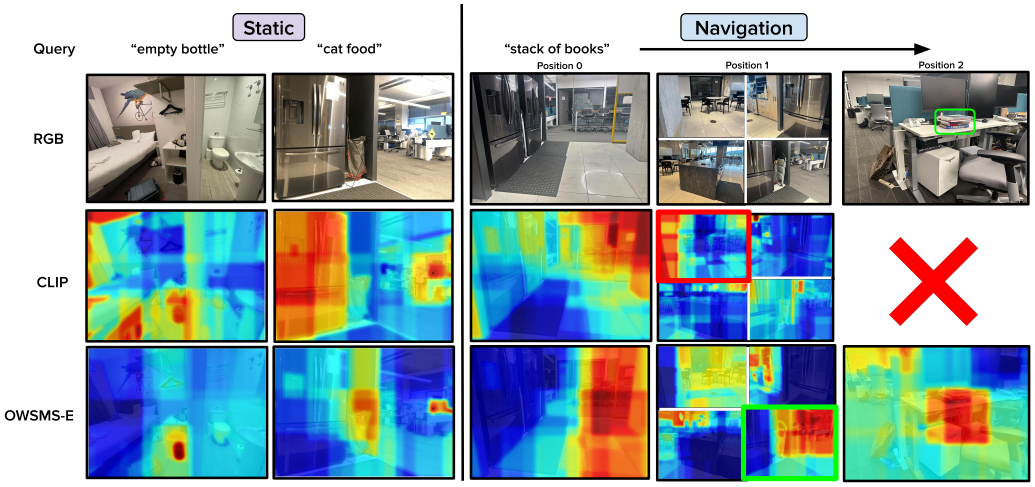

Figure 2: **Object navigation experiments in real. Left:** Two examples from the static dataset evaluation with the 2D probability distributions generated for OWSMS and CLIP. Red corresponds to high probability while blue corresponds to low probability. **Right:** A short horizon navigation example where we start at position 0 and end at position 2. OWSMS 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."

Table 5 shows the results. We see that OWSMS outperforms both CLIP and OWL-ViT. More examples are in the Appendix. We hypothesize 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 would bring trouble for searching occluded objects. For example, CLIP would associate a bottle full of water with an empty bottle while LLMs will associate the empty bottle with the trash can. In addition, CLIP has a "bag of words" behaviour [30], causing it to relate "cat food" with a fridge instead of a cat sign. Language models show they have better semantic reasoning as in Figure 2 "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 suited for searching occluded objects. OWSMS-E slightly outperforms OWSMS-LLM as they are both bottle-necked by the quality of labels from BLIP-2.

In the second experiment, we have an agent that follows the downstream navigation policy described in Section 4.3 and selects action based on the semantic distribution from CLIP and OWSMS-E. As shown in Figure 2, 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 pillars in the scene rather than the area with office desks and chairs. OWSMS-E continues towards the office (green box in position 1) and finds a stack of books on a desk successfully.

6 Limitations and Future Work

We present OWSMS, a framework for generating a semantic occupancy distribution, and show that it facilitates mechanical search in both unconstrained and constrained settings. The method has the following limitations, which open up possibilities for future work: (1) We only evaluate the navigation task with a heuristic navigation policy combined with the generated distribution without large-scale evaluations of other navigation policies; (2) The performance of OWSMS without object lists is bottlenecked by the image captioning model, and even the best open-source models have significant amounts of noise; (3) OWSMS considers pairwise affinities between the objects and the target, but not among more objects. Future work might explore how to condition the LLM on the presence of more objects.

References

[1] 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.

[2] 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.

[3] 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)*, pages 2045–2052, 2021.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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*, pages 17359–17371. PMLR, 2022.

[10] 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.

[11] 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*, pages 416–426. PMLR, 2023.

[12] 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.

[13] 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.

[14] 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.

[15] 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*, pages 991–1002. PMLR, 2022.

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

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

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

[19] 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.

[20] 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.

[21] 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.

[22] 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.

[23] 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*, pages 9118–9147. PMLR, 2022.

[24] 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.

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

[26] 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.

[27] 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.

[28] 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.

[29] 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. URL https://openreview.net/forum?id=UW5A3SweAH.

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

[31] X. Gu, T.-Y. Lin, W. Kuo, and Y. Cui. Open-vocabulary object detection via vision and language knowledge distillation, 2022.

[32] 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.

[33] X. Zhou, R. Girdhar, A. Joulin, P. Krähenbühl, and I. Misra. Detecting twenty-thousand classes using image-level supervision, 2022.

[34] 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.

[35] 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.

[36] 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'*, pages 146–151. IEEE, 1997.

[37] J. Maja. Integration of representation into goal-driven behavior-based robots. *IEEE transactions on robotics and automation*, 8(3):304–312, 1992.

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

[39] B. Yamauchi and R. Beer. Spatial learning for navigation in dynamic environments. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 26(3):496–505, 1996.

[40] 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.

[41] 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.

[42] 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*, abs/2301.12597, 2023.

[43] M. Gupta, T. Rühr, M. Beetz, and G. S. Sukhatme. Interactive environment exploration in clutter. pages 5265–5272, 2013.

[44] M. R. Dogar, M. C. Koval, A. Tallavajhula, and S. S. Srinivasa. Object search by manipulation. *Autonomous Robots*, 36(1):153–167, 2014.

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

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

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

[48] 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.

[49] 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.

[50] 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*, pages 15691–15700, 2021.

[51] 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*, pages 1022–1031, 2019.

[52] 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)*, pages 3316–3323. IEEE, 2022.

[53] 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)*, pages 3357–3364. IEEE, 2017.

[54] 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*, pages 17031–17041, 2022.

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

[56] S. Thrun. Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 99(1):21–71, 1998.

[57] H. J. S. Feder, J. J. Leonard, and C. M. Smith. Adaptive mobile robot navigation and mapping. *The International Journal of Robotics Research*, 18(7):650–668, 1999.

[58] B. Kuipers and Y.-T. Byun. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Robotics and autonomous systems*, 8(1-2):47–63, 1991.

[59] B. H. Wilcox. Robotic vehicles for planetary exploration. *Applied Intelligence*, 2:181–193, 1992.

[60] 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*, 33: 4247–4258, 2020.

[61] 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*, 35:5982–5994, 2022.

[62] 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)*, pages 13194–13200. IEEE, 2021.

[63] 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*, 33:9700–9712, 2020.

[64] 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*, pages 6750–6759, 2019.

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

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

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

[68] 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.

[69] 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.

[70] 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.

[71] 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.

[72] 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.

[73] O. Mees and W. Burgard. Composing pick-and-place tasks by grounding language, 2021.

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

[75] 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.

[76] 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.

[77] 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.

[78] 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.

[79] 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.

[80] Keras ocr. https://support.google.com/merchants/answer/6324436?hl=en.

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

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

[83] J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991. doi:10.1109/18.61115.