Appendix

Anonymous Author(s)
Affiliation
Address
email

1 Scene Understanding

1.1 Object detection + OCR

Because ViLD is a general-purpose detector, it cannot easily distinguish between objects belonging to the same domain (e.g., Advil versus Ibuprofen). Because of this, we use OCR with Keras OCR[1] to improve the quality of the object detections. While OCR has been used in prior work to aid object detection [2], we use text embedding combined with OCR for better performance. For each object, we concatenate the text observed on it and compute the text embedding using OpenAI Embeddings. We compute the dot product between the embeddings of the concatenated text and every class label. We normalize this probability vector by subtracting the minimum value and then adjusting the vector with some temperature. We finally multiply this by the object detection probability vector.

Let C_i denote the class label of object $\mathcal{O}_i$ (e.g., "Tylenol" as opposed to the broader category "medication"); I_i represent the general shape, size, and color-related features of $\mathcal{O}_i$; and T_i be the detected text on $\mathcal{O}_i$. Recall that all objects belong to some class C_i. We calculate

$$
\begin{aligned}
&P(C_i|\ I_i, T_i) \\
&= \frac{P(I_i, T_i|C_i) \cdot P(C_i)}{P(I_i, T_i)} \\
&= \frac{P(T_i|I_i, C_i) \cdot P(I_i|C_i) \cdot P(C_i)}{P(I_i, T_i)} \\
&= \frac{P(T_i|C_i) \cdot P(I_i|C_i) \cdot P(C_i)}{P(I_i, T_i)} \\
&= \frac{P(C_i|T_i)P(T_i)}{P(C_i)} \cdot \frac{P(C_i|I_i)P(I_i)}{P(C_i)} \cdot \frac{P(C_i)}{P(I_i, T_i)} \\
&\propto P(C_i|T_i)P(C_i|I_i)
\end{aligned}
$$

as T_i is independent of I_i when conditioned on C_i, and $P(C_i)$ is uniform. This illustrates that the multiplication of the OCR probabilities and the object detection probabilities can give us a refined estimate of the category probabilities.

We test object detection performance on scenes generated through isolated perception experiments. We take RGB images of 100 scenes of the Pharmacy domain using a high-resolution camera and study the effect of having OCR.

Results for this experiment are in Table 1. As is standard in the computer vision literature, we report mAP (mean Average Precision) averaged over intersection-over-union (IOU) thresholds from 0.50 to 0.95 with a step size of 0.05, as well as top-k classification accuracy (i.e., if the ground truth label appears in the k labels with the highest probabilities). The results show that OCR leads to a significant improvement across all metrics, with mAP improving by a factor of 12 and top-1 accuracy improving by a factor of 3.

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

Table 1: Object Detection Refinement Results. We study the effect of OCR and report the mean average precision (mAP) of the predicted bounding boxes and top-K accuracy of the predicted labels.

Method	**mAP** (↑)	**Top-K Accuracy %** (↑)		
		k=1	k=3	k=5
ViLD	2.4	14.7	32.3	41.6
ViLD + OCR	28.9	45.0	62.0	69.5

2 Creating the Semantic Distribution

2.1 Offline Semantic Distribution Generation with Object List

We now generate a semantic distribution based on the affinity matrix and detected objects. The semantic occupancy distribution models the probability that the target object occupies a given location, given the classes of observed objects in the scene, i.e. $P(L_T = l \mid L_{1...n} = l_{1...n}, C_{1...n} = c_{1...n})$, where L_T is the location of the target object, $L_{1...n}$ are the positions of the *visible* objects, and $C_{1...n}$ are the inferred classes of the visible objects. We abbreviate this quantity as $P(L_T = l \,|L, C)$.

We interpret affinity values M_{ij} to be the probability of object j being the closest to object i in expectation across scenes. However, given the current scene, there may be more or less space that is nearest to a particular object, so we interpret these affinity values as being normalized per unit area. Thus, formally, given $N(l)$ representing the index of the object closest to location $l = (x_l, y_l)$, $P(L_T = l \,|L, C) \propto M_{target,N(l)}$.

In simulation experiments for constrained environments, $N(\cdot)$ is computed using the 3D coordinates of the visible objects obtained from the depth image. We compute the 2D semantic occupancy distribution (in the horizontal plane of the shelf) and reduce it to 1D by summing along camera rays. In physical experiments, to avoid errors due to noisy depth readings we compute the distribution directly in 2D, using pixel distance for $N(\cdot)$ instead of world coordinates.

3 Combining with Downstream Policies

3.1 Constrained Environments

We consider the problem of robotic mechanical search for a target object $\mathcal{O}_T$ in a cluttered, semantically organized shelf containing the target and N other rigid objects $\{\mathcal{O}_1, ..., \mathcal{O}_N\}$ of cuboidal shapes in stable poses. We build on the problem statement and assumptions in Huang et al. [3]. We model the setup as a finite-horizon Partially Observable Markov Decision Process (POMDP). States $s_t \in \mathcal{S}$ consist of the full geometries and poses of the objects in the shelf at timestep t and observations $y_t \in \mathcal{Y} = \mathcal{R}^{H \times W \times 4}$ are RGBD images from a robot-mounted depth camera at timestep t. Actions $a_t \in \mathcal{A} = \mathcal{A}_p \cup \mathcal{A}_s$ are either *pushing* or *suction* actions, where the former are horizontal linear translations of an object along the shelf and the latter pick up an object with a suction gripper and translate it to an empty location on the shelf with no other objects in front of it. We make the following assumptions:

- The dimensions of the shelf are known.
- Each dimension of each object is between size $S_{\min} = 5\,\text{cm}$ and size $S_{\max} = 25\,\text{cm}$.
- The shelf is semantically organized.
- The names of all objects in the shelf are a subset of a known list of object names.
- Actions cannot inadvertently topple objects or move multiple objects simultaneously.

Table 2: 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
OWSMS-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
OWSMS-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
OWSMS-E	**186/192**	$\mathbf{1.56 \pm 0.08}$	**188/194**	2.15 ± 0.15	**184/193**	3.00 ± 0.27	**167/191**	$\mathbf{3.07 \pm 0.25}$
OWSMS-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
OWSMS-E	**173/194**	3.01 ± 0.22	152/188	3.80 ± 0.31	140/190	4.78 ± 0.44	115/181	$\mathbf{5.33 \pm 0.50}$
OWSMS-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

4 Experiments

4.1 Scene Generation

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.

4.2 Simulation Object Retrieval Experiments

We run an extensive suite of experiments using the same simulator as prior work in mechanical search on shelves [4] and study the benefit brought by OWSMS. We use a grid search on the average number of actions required in the pharmacy domain with 15 objects to tune the Gaussian smoothing σ to be 50 pixels and γ for PaLM to be 1 and for OpenAI Embeddings to be 0.004. We use the same parameters for the other two domains.

We generate scenes with various numbers of objects: $N = 12$, 15, 18, and 21. We generate 200 scenes for each value of N. We discard scenes where the target object starts out visible, resulting in

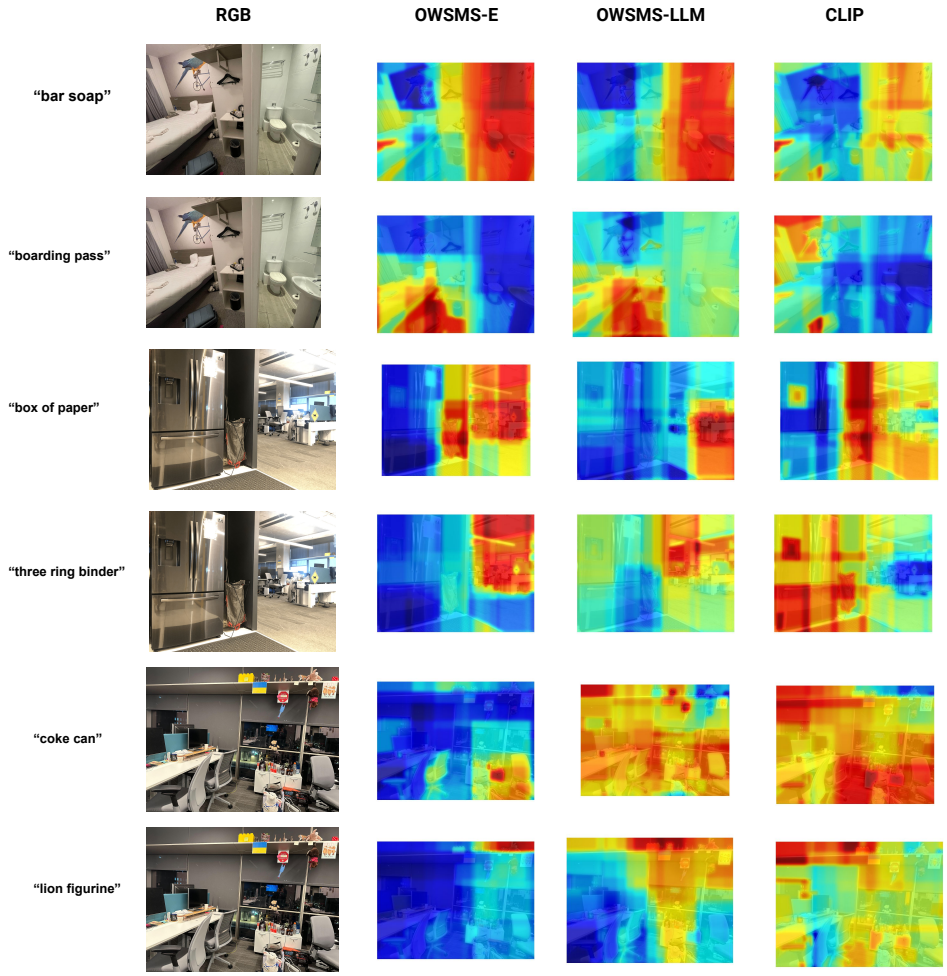

Figure 1: More examples of the semantic distributions generated by different methods from the static dataset.

just under 200 scenes for each value of N. Termination occurs when at least $X = 1\%$ of the target object becomes visible or reaching maximum action number $2N$. The reason for the low threshold is that the DAR policy has trouble making progress on a partially revealed target object [4], which may dilute the comparison between different methods for generating semantic distributions.

We report results for all numbers of objects N in Table 2, OWSMS-LLM outperforms both OWSMS-E, while also beating LAX-RAY across various values of N in terms of success rate (by an additional 30/741 scenes) and average number of actions required (by 32.4%). A point of note is that the action differential percentage grows as the number of objects increases. At 21 objects, LAX-RAY requires 8.24 actions on average, whereas OWSMS requires just 5.47. This trend agrees with intuition that it is unscalable to search large environments with no semantic intuition.

4.3 Physical Object Retrieval Experiment

Because the RealSense camera is not able to capture the fine details of the text on the objects when observing the entire scene at resolution 640×480 pixels, we perform a three-stage scan of the scene by moving the end-effector to 3 adjacent positions, all of which are closer to the shelf, where the text is more easily readable. At each of these poses, we take a picture of the scene, project the known world position of the objects to the new camera frame, identify text with OCR, and assign

each text detection to the object it is contained in. If there are detections on the same object from multiple scan locations, we use the OCR that has the lowest entropy for its distribution, a measure of confidence. During the physical experiments rollouts, when the action given by the policy causes unintentional toppling or a missed grasp due to depth sensor noise, we reset the object to undo the action and run the policy again.

4.4 Experiments for Unconstrained Environments

More examples of the semantic distribution comparison between OWSMS and CLIP based models are shown above.

5 Object Lists in Constrained Environments

Pharmacy Domain : vitamins , fish oil , omega-3 , calcium , probiotics , protein powder , COQ10 , anthocyanin , shampoo , conditioner , toothpaste , toothbrush , dental floss , face wash , sunscreen , lotion , hand cream , body wash , aspirin , tylenol , ibuprofen , advil , pain relief , shaving cream , eye drops , deodorant , band-aid

Kitchen Domain : spoon , ladle , spatula , tongs , whisk , fork , peeler , grater , saucepan , frying pan , salt , pepper , cumin , coriander , basil , turmeric , parsley , oregano , sugar , flour , cornstarch , oats , quinoa , rice

Office Domain :pen , pencil , highlighter , sticky note , binder paper , printer paper , index card , paper CLIP , rubber band , stapler , staples , tape dispenser , 3-hole punch , dry erase marker , sharpie , label maker , notebook , eraser , white-out , calculator , thumbtack , pencil sharpener , bubble wrap , styrofoam , packing tape , shipping boxes , ethernet cable , modem , router , network card , network bridge , headphones , speakers , aux cable , microphone , keyboard , mouse , USB adapter , hard drive , flash drive

References

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

[2] S. Karaoglu, J. Gemert, and T. Gevers. Object reading: Text recognition for object recognition. volume 7585, 10 2012. ISBN 978-3-642-33884-7. doi:10.1007/978-3-642-33885-4_46.

[3] 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.

[4] 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.