

Open-World Object Manipulation using Pre-Trained Vision-Language Models

Anonymous Author(s)

Affiliation

Address

email

1 **Abstract:** For robots to follow instructions from people, they must be able to
2 connect the rich semantic information in human vocabulary, e.g. “can you get me
3 the pink stuffed whale?” to their sensory observations and actions. This brings
4 up a notably difficult challenge for robots: while robot learning approaches allow
5 robots to learn many different behaviors from first-hand experience, it is imprac-
6 tical for robots to have first-hand experiences that span all of this semantic infor-
7 mation. We would like a robot’s policy to be able to perceive and pick up the pink
8 stuffed whale, even if it has never seen any data interacting with a stuffed whale
9 before. Fortunately, static data on the internet has vast semantic information, and
10 this information is captured in pre-trained vision-language models. In this paper,
11 we study whether we can interface robot policies with these pre-trained models,
12 with the aim of allowing robots to complete instructions involving object cate-
13 gories that the robot has never seen first-hand. We develop a simple approach,
14 which we call Manipulation of Open-World Objects (MOO), which leverages a
15 pre-trained vision-language model to extract object-identifying information from
16 the language command and image, and conditions the robot policy on the current
17 image, the instruction, and the extracted object information. In a variety of exper-
18 iments on a real mobile manipulator, we find that MOO generalizes zero-shot to
19 a wide range of novel object categories and environments. In addition, we show
20 how MOO generalizes to other, non-language-based input modalities to specify
21 the object of interest such as finger pointing, and how it can be further extended to
22 enable open-world navigation and manipulation. The project’s website and evalua-
23 tion videos can be found at <https://robot-moo-anon.github.io/>.

24 1 Introduction

25 For a robot to be able to follow instructions from humans, it must cope with the vast variety of
26 language vocabulary, much of which may refer to objects that the robot has never interacted with
27 first-hand. For example, consider the scenario where a robot has never seen or interacted with a plush
28 animal from its own camera, and it is asked, “can you get me the pink stuffed whale?” How can the
29 robot complete the task? While the robot has never interacted with this object category before, the
30 internet and other data sources cover a much wider set of objects and object attributes than the robot
31 has encountered in its own first-hand experience. In this paper, we study whether robots can tap into
32 the rich semantic knowledge captured in such static datasets, in combination with the robot’s own
33 experience, to be able to complete manipulation tasks involving novel object categories.

34 Computer vision models can capture the rich semantic information contained in static datasets.
35 Indeed, composing modules for perception, planning, and control in robotics pipelines is a long-
36 standing approach [1, 2, 3], allowing robots to perform tasks with a wide set of objects [4]. How-
37 ever, these pipelines are notably brittle, since the success of latter motor control modules relies on
38 precise object localization. On the other hand, several prior works have trained neural network
39 policies with pre-trained image representations [5, 6, 7, 8] and pre-trained language instruction em-
40 beddings [9, 10, 11, 12]. While this form of vanilla pre-training can improve efficiency and gen-
41 eralization, it does not provide a mechanism for robots to ground and manipulate novel semantic
42 concepts, such as unseen object categories referenced in the language instruction. This leads to a
43 crossroads — some approaches can conceivably generalize to many object categories but rely on
44 fragile pipelines; others are less brittle but cannot generalize to new semantic object categories.

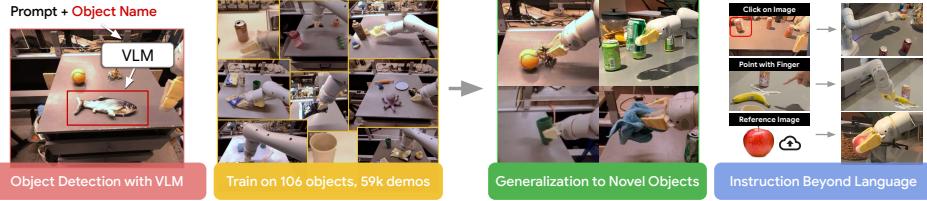


Figure 1: Overview of MOO. We train a language-conditioned policy conditioned on object locations from a frozen VLM. The policy is trained on demonstrations spanning a set of 106 objects using VLM-based object-centric representations, enabling generalization to novel objects, locations produced from new modalities.

45 To allow robots to generalize to new semantic concepts, we specifically choose to leverage open-
 46 vocabulary pre-trained vision-language models (VLMs), rather than models pre-trained on one
 47 modality alone. Such models capture the rich information contained in diverse static datasets, while
 48 grounding the linguistic concepts into a perceptual representation that can be connected to the robot’s
 49 observations. Crucially, rather than using the pre-trained model for precise state estimation in its en-
 50 tirety (akin to pipelined approaches), we only use the VLM to identify the relevant objects in the
 51 image by coarsely localizing them, while allowing an end-to-end trained policy to use this informa-
 52 tion along with the original observation to perform the task. More specifically, our system receives a
 53 language instruction and uses a VLM to identify the 2D image coordinates of objects in the instruc-
 54 tion. Along with the image and the instruction, the 2D coordinates of the objects are fed into our
 55 manipulation policy allowing it to ground the natural language to objects and know which objects
 56 to act upon without seeing any demonstrations with those objects. The VLM is frozen throughout
 57 all of our training, and the policy is trained with the real VLM detector in the loop to prevent the
 58 brittleness that can plague prior pipelined approaches.

59 The main contribution of this paper is a flexible approach for open-world object manipulation that
 60 interfaces policy learning with pre-trained vision-language models. An overview is given in Fig. 1.
 61 The pre-trained models are trained on massive static image and language data that far exceeds the
 62 robot’s own experience. The robot’s policy is trained to perform skills using demonstration data
 63 covering a more modest yet still physically diverse set of 106 training objects. The composition
 64 of the pre-trained vision-language model and the control policy leads to an overarching language-
 65 conditioned policy that can complete commands that refer to novel semantic categories.

66 We study the performance of our method across 1,472 evaluations on a real robotic manipulator,
 67 where we find that our approach is significantly more successful than recent robot learning methods.
 68 Beyond verbal object descriptions, we also find that the trained policy can be easily combined with
 69 other means of communicating intent, e.g., pointing at an object and inferring the object description
 70 using a VLM, showing a generic image of the object of interest, or using a simple GUI. Finally,
 71 our experiments further show that our method can be integrated with an open-vocabulary object
 72 navigation model called Clip-on-Wheels (CoW), to complete mobile manipulation tasks involving
 73 novel objects. Throughout this paper, we refer to our approach as Manipulation of Open-vocabulary
 74 Objects (MOO) and the integrated mobile manipulation system as CoW-MOO.

75 2 Related Work

76 **Leveraging Pre-Trained Models in Robotic Learning.** Using off-the-shelf vision, speech, or lan-
 77 guage models is a long-standing approach in robotics [13, 14, 10]. Modern pre-trained vision and
 78 language models have improved substantially over older models, and have played an increasing role
 79 in robotics research. A large body of prior work has trained policies on top of frozen or fine-tuned
 80 visual representations [5, 15, 6, 16, 17, 18, 19, 7, 8, 20, 21], while other works have leveraged pre-
 81 trained language models [22, 23, 9, 10, 11, 24, 25, 12]. Unlike these prior works, we aim to leverage
 82 vision-language models that ground language in visual observations. Our use of vision-language
 83 models enables generalization to novel semantic object categories, which cannot be achieved by
 84 using vision models or language models individually.

85 **Generalization in Robotic Learning.** A number of recent works have studied how robots can com-
 86 plete novel language instructions [26, 22, 23, 9, 10, 11, 27, 28, 24], typically focusing on instructions
 87 with novel combinations of words, i.e. compositional generalization, or instructions with novel ways

88 to describe previously-seen objects and behaviors. Our work focuses on how robots can complete
89 instructions with entirely new words that refer to objects that were not seen in the robot’s demonstra-
90 tion dataset. Other research has studied how robot behaviors like grasping and pick-and-place can
91 be applied to unseen objects [29, 30, 31, 32, 33, 34, 35, 36, 37], focusing on generalization to visual
92 or physical attributes. Our experimental settings require visual and physical object generalization
93 but also require semantic object generalization. That is, unlike these prior works, the robot must be
94 able to ground a description of a previously-unseen object category.

95 **Vision-Language Models for Robotic Manipulation.** Two closest related works to our approach
96 are CLIPort [38] and PerAct [12] that use the CLIP vision-language model as a backbone of their
97 policy. Both of these approaches have demonstrated impressive level of generalization to unseen
98 semantic categories and attributes. Inspired by these works, we aim to expand them to more general
99 manipulation settings by: i) removing the need for depth cameras or camera calibration, ii) expand-
100 ing and demonstrating that the hereby introduced representation works with other modalities such
101 as pointing to the object of interest, iii) moving beyond 2D manipulation tasks, e.g. demonstrating
102 the approach on tasks such as reorienting objects upright as well as mobile manipulation tasks.

103 **Open-World Object Detection in Computer Vision.** Historically, object-detection methods have
104 been restricted to a fixed category map covering a limited set of objects [39, 40, 41, 42]. These
105 methods work well for the object categories on which they are trained, but have no knowledge of
106 objects outside their limited vocabulary. Recently, a new wave of object detectors have emerged that
107 aim to go beyond simple closed-vocabulary tasks by replacing the fixed one-hot category prediction
108 with a shared image-language embedding space that can be used to answer open-vocabulary object
109 queries [43, 44, 45, 46]. Typically these methods rely on internet-scale data in the form of pairs of
110 image and associated descriptive text to learn the grounding of natural language to objects. Various
111 methods have been employed to then extract object localization information in the form of bound-
112 ing boxes and segmentation masks. In our work, we use the OWL-ViT detector due to it’s strong
113 performance in the wild and publicly available implementation [43].

114 3 Manipulation of Open-World Objects (MOO)

115 The key goal of MOO is to develop a policy that can leverage the visually-grounded semantic in-
116 formation captured by pre-trained vision-language models for generalization to object types not in
117 the policy training set. More specifically, we aim to use the VLM to localize objects described in
118 a given instruction, while allowing the policy to complete the task using both the instruction and
119 the object localization information from the VLM. MOO accomplishes this in two stages. First,
120 the current observation and the words in the instruction corresponding to object(s) are passed to
121 the VLM to localize the objects. Then, the object localization information and the instruction sans
122 object information are passed to the policy, along with the original observation, to predict actions.

123 The key design choice of MOO lies in how to represent object information encoded in VLMs and
124 how to feed that information to the instruction-conditioned policy. In the remainder of this section,
125 we first overview the set-up, then describe the design of these crucial aspects of the method, and
126 finally provide an overview of the model architecture and the training procedure as well as describe
127 practical implementation details that allows us to deploy MOO on real robots.

128 3.1 Problem Set-Up

129 Formally, we assume that the robot, with image observations $o \in \mathcal{O}$ and actions $a \in \mathcal{A}$, is pro-
130 vided with a set of expert demonstrations $\mathcal{D}_{\text{robot}}$ collected via teleoperation. Each demonstra-
131 tion corresponds to a sequence of observation-action pairs $\{(o_j, a_j)\}_{j=1}^T$ collected over a time horizon
132 T , and is annotated with a structured language instruction ℓ for the task being performed in the
133 demonstration. To help facilitate object generalization, we assume that these language instructions
134 are structured as a combination of a template and a list of object descriptions within that template.
135 For example, for the instruction $\ell = \text{"move yellow banana near cup,"}$, the template is “move X near
136 Y,” and the object descriptions are $X = \text{"yellow banana"}$ and $Y = \text{"cup."}$ Inspired by RT-1 [24], in
137 this work, we focus on five different types of skills defining the templates: “pick X,” “move X near
138 Y,” “knock X over,” “place X upright,” and “place X into Y.”

139 All of the objects seen in the demonstrations are drawn from a set $\mathcal{S}_{\text{robot}}$, and our objective is to
140 complete new structured language instructions with a seen template but novel objects that are not in

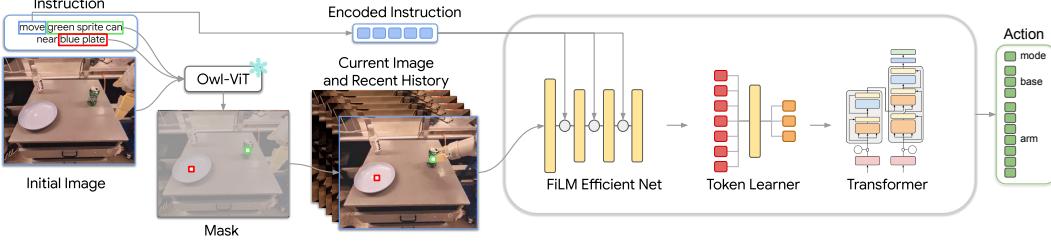


Figure 2: MOO architecture: We extract object location (represented as the center of the bounding box) on the first frame of an episode. The segmentation mask is concatenated channel-wise to the input image for each frame. We remove the language embedding for everything except the task so that the object specific information is only provided through the object instance mask.

141 S_{robot} , which also have novel object descriptions. In aiming to complete this goal, our approach will
 142 leverage imitation learning and vision-language models, which we briefly review in the Appendix.

143 3.2 Representing Object Information

144 To utilize the object knowledge encoded in the VLMs, we need to pick a representation that can
 145 be easily transferred to the text-conditioned control policy. We start by identifying the instruction
 146 template (represented by verb v) and object X (or list of objects X, Y, \dots) from the instruction
 147 ℓ . Equipped with an object description X , we query a VLM to produce a bounding box of the
 148 object of interest with the prompt $q = \text{"An image of an } X\text{"}$, and use the resulting detection (if
 149 any) as conditioning of our policy. To reduce the reliance of the exact segmentation of the object
 150 dimensions, we select a single pixel that is at the center of the predicted bounding box as the object
 151 representation. In the case of one object description, we use a single-channel object mask with the
 152 value set to 1.0 at the pixel of the object’s predicted location. In the case of two object descriptions,
 153 we set the pixel value of the first to be 1.0 and the second to be 0.5.

154 This design has two main advantages: first, it is a generic representation that works with objects of
 155 any size as long as they are visible, and second, it is compatible with a large selection of vision meth-
 156 ods such as bounding boxes or segmentation masks as these can be easily transformed into a single,
 157 object-centric pixel location. We ablate other object representation choices in the experiments.

158 Importantly, this approach can handle object descriptions that are not previously seen in the robot’s
 159 demonstration data, as long as it is sufficiently represented in the static large-scale training data
 160 of the VLM. For any unseen objects, we simply include a description in the task command, e.g.,
 161 “pick *stuffed toy whale*.” Once the object description is translated into a pixel location by the VLM,
 162 the robot’s policy trained on demonstration data only needs to be capable of interpreting the mask
 163 location and how to physically manipulate the novel object’s shape, rather than needing to also
 164 ground the semantic object description.

165 3.3 Architecture and Training of MOO

166 We present the model architecture and information flow of MOO in Fig. 2. As described above, we
 167 extract the object descriptions from the language instruction and together with the initial image feed
 168 them into the VLM to output object locations in the image. This information is then represented as
 169 an object mask with dots at the center of the objects of interest.

170 Once we obtain the mask, we append it channel-wise to the current image together with the recent
 171 image history, which is passed into the RT-1 policy architecture [24]. We use a language model
 172 to encode the previously extracted verb v part of the language instruction in an embedding space
 173 of an LLM. The images are processed by an EfficientNet [47] conditioned on the text embedding
 174 via FiLM [48]. This is followed by a Token Learner [49] to compute a small set of tokens, and
 175 finally a Transformer [50] to attend over these tokens and produce discretized action tokens. We
 176 refer the reader to the RT-1 paper for details regarding the later part of the architecture. The action
 177 space corresponds to the 7-DoF delta end-effector pose of the arm (including x, y, z, roll, pitch, yaw
 178 and gripper opening). The entire policy network is trained end-to-end using the imitation learning
 179 objective and we specify the details of the objective in the Appendix (Equation 1). Importantly, the
 180 VLM used to detect the objects is frozen during training, so that it does not overfit to the objects in
 181 the robot demonstration data. The policy is trained with this frozen VLM in the loop, so that the
 182 policy can learn to be robust to errors made by the VLM given other information in the image.

183 **3.4 Practical Implementation**

184 To detect objects in our robot images, we use the Owl-ViT open-vocabulary object detector [43]. In
185 practice, we find that it is capable of detecting most clearly visible objects without any fine-tuning,
186 given a descriptive natural language phrase. The interface to the detector requires a natural language
187 phrase describing what to detect (e.g., “An image of a small blue elephant toy.”) along with an image
188 to run the detection on. The output from the model is a score map indicating which locations are
189 most likely to correspond to the natural language description and their associated bounding boxes.
190 We select a universal score threshold to filter detections. To detect our objects, we rely on some
191 prompt-engineering using descriptive phrases including the color, size, and shape of objects, though
192 most of our prompts worked well on the first attempt. We share the prompts in the Appendix.

193 To make the inference time practical on real robots, we extract the object information via VLM only
194 in the first episode frame. By doing so, most of the heavy computation is executed only once at
195 the beginning and we can perform real-time control for the entire episode. Since the information is
196 appended to the current observation, we rely on the policy to find the corresponding object in the
197 current image if the object has moved since the first timestep.

198 **3.5 Training Data**

199 We start with the demonstration data used by RT-1 [24] covering 16 unique objects. Despite the use of the VLM for
200 semantic generalization, we expect that the policy will require more physical object diversity to generalize to novel objects.
201 Therefore, we expand the dataset with additional diverse “pick” data across a set of 90 diverse objects, for a total of
202 106 objects, as shown in Figure 3. We choose to only expand the set of objects for the picking
203 skill, since it is the fastest skill to execute and therefore allows for the greatest amount of diverse
204 data collection within a limited budget of demonstrator time. Our additional set of 90 diverse objects
205 only appear in “pick” episodes. All other tasks, such as “move near” or “place into”, must be learned
206 from the original 16 objects in the RT-1 dataset. Detailed statistics are in Fig. 9 in the Appendix.
207



Figure 3: (Left) RT-1 objects that account for $\approx 70\%$ of training data covering all skills. (Middle) Diverse training objects that appear only in “pick” demonstrations. (Right) Unseen objects used only for evaluation.

218 **4 Experiments**

219 Our experiments answer the following questions: 1) Does MOO generalize across objects for different
220 skills including unseen objects? 2) Does MOO generalize beyond new objects – Is MOO robust
221 to distractors, backgrounds and environments? 3) Can the intermediate representation used support
222 non-linguistic modalities to specify the task? 4) Does the object generalization performance scale
223 with the (a) number of training episodes, (b) number of unique objects in the training episodes and
224 (c) size of the model? 5) Can MOO be used for open-world navigation and manipulation?

225 **4.1 Experimental Setup**

226 **Seen and unseen objects.** The training data is collected with teleoperation on table-top environments
227 across a set of 106 different object types. We evaluate performance on 49 objects “seen” in
228 training and report the performance as “seen”. We hold out 47 objects not present in training and
229 report performance on these as “unseen”. Note that previous works often focus on unseen combinations
230 of previously seen commands and objects (e.g. “pick an apple” even though the training data
231 contains “move an apple into a bowl” and “pick a bowl”); we adopt a more strict definition of unseen
232 objects, where our unseen object categories were not seen in the robot’s training demonstration data
233 at any point for any task, therefore making our unseen performance a zero-shot object generalization
234 problem. Furthermore, we report results across different environments that introduce novel textures,
235 backgrounds, locations, and additional open-world objects not present in the training data.

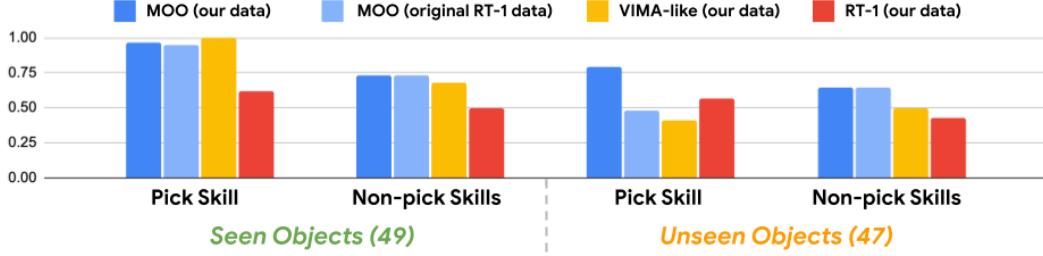


Figure 4: **Main Results.** While baseline methods perform competitively on in-distribution combinations of objects and skills seen during training, they fail to generalize to novel objects. MOO substantially improves generalization to novel object categories unseen during training, especially for the “pick” skill.

236 **Evaluation details.** We evaluate on a set of tabletop tasks involving manipulating a diverse set
 237 of objects. We use mobile manipulators with 7 degree-of-freedom arm and two-fingered gripper
 238 (Fig 10). Our experiments evaluate the percent of successfully completed manipulation commands
 239 which include five skills: “pick”, “move near”, “knock,” “place upright,” and “place into” across
 240 a set of evaluation episodes (definition and success criteria follow RT-1 [24] and are described in
 241 the Appendix). To study object specificity and robustness, for all evaluation episodes, we include
 242 between two to four distractor objects in the scene which the robot should not interact with. For
 243 each evaluation episode, we randomly scatter the evaluation object(s) and the distractor objects onto
 244 the table. There is no consistent placement of the target object relative to the distractors. We repeat
 245 this process 21 times and report the performance. We present the experimental setup in Figure 10.

246 **Baselines.** We compare two prior methods: RT-1 [24] and a modified version of VIMA [25], referred
 247 to as “VIMA-like”. VIMA-like preserves the cross-attention mechanism, but uses the mask image
 248 as the prompt token and the current image as state token. These modifications are necessary because
 249 VIMA uses Transporter-based action space and is not applicable to our task, i.e., our robot arm
 250 moves in 6D and has a gripper that can open and close continuously. These two baselines correspond
 251 to common alternatives where the computer vision data is used as a pre-training mechanism (as in
 252 RT-1) or object-centric information is fed to the network through cross attention (as in VIMA-like).

253 4.2 Experimental Results

254 Generalization to Novel Objects.

255 We investigate the question: *Does*
 256 *MOO generalize across objects for*
 257 *different manipulation skills includ-*
 258 *ing objects never seen at training*
 259 *time?* Experiments are presented in
 260 Figure 4 and example trajectories are
 261 shown in Figure 12 in the Appendix.
 262 Relative to the baselines on the pick
 263 tasks, MOO exhibits substantial im-
 264 provement over the seen object per-
 265 formance as well as the unseen ob-
 266 jects, which in both cases reaches

267 $\sim 50\%$ improvement. MOO can cor-
 268 rectly utilize a VLM to find novel ob-
 269 jects and incorporate that informa-
 270 tion more effectively than the VIMA-like
 271 baseline. When comparing the perfor-
 272 mance on seen objects for the skills other than *pick*, we ob-
 273 serve a slightly worse performance than for the *pick* tasks. This is understandable since the “Seen
 274 objects” for “Non-pick skills” have only been seen during the *pick* episodes as shown in Fig. 9. This
 275 demonstrates MOO’s ability to transfer the learned object generalization across the skills so that the
 276 objects that have only been picked can now be also used for other skills. In addition, MOO exhibit
 277 generalization to unseen objects (i.e. unseen in any previous tasks, including pick) that is at the same
 278 level as for unseen objects for the *pick* skill, and 50% better than baseline.

279 **Robustness Beyond New Objects.** To further test the robustness of MOO, we analyze novel evalua-
 280 tion settings with significantly increased difficulty and visual variation, which are shown in Fig. 6.

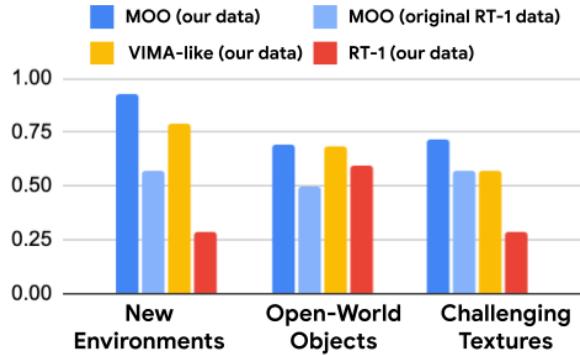


Figure 5: MOO is able to generalize to new objects, textures, and environments with greater success than prior methods. Visualiza-
 tions are shown in Fig. 6.

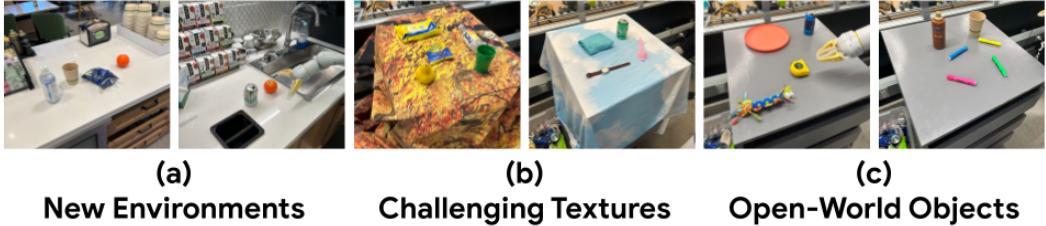


Figure 6: To study the robustness of MOO, we evaluate on (a) new environments, (b) challenging texture backgrounds which are visually similar to unseen objects in the scene, and (c) additional open-world objects.

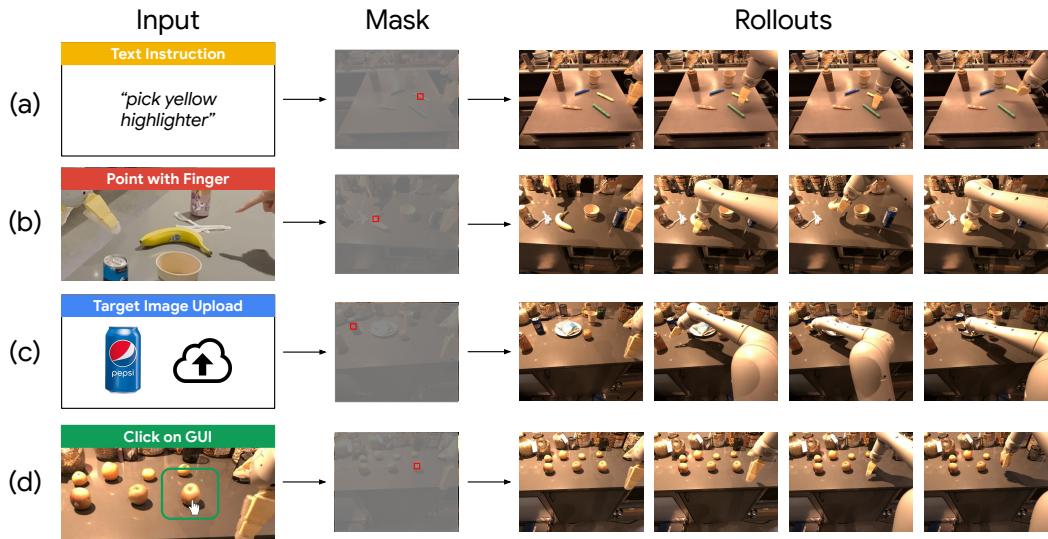


Figure 7: We explore using various input modalities to generate the single-pixel object representations used by MOO. (a) shows the standard mask generation process using OWL-ViT with a text instruction. (b) shows using a VLM to generate a text caption, then fed to OWL-ViT. (c) shows an uploaded image to prompt OWL-ViT. (d) shows a user providing a ground-truth mask via a GUI.

280 To reduce the number of real robot evaluations, we focus this comparison on the picking skill. The
 281 results are presented in Fig. 5. Across these challenging evaluation scenes, MOO is significantly
 282 more robust compared to VIMA-like [25] and RT-1 [24]. This indicates that the use of VLMs in
 283 MOO not only improves generalization to new objects that the robot has not interacted with, but also
 284 significantly improves generalization to new backgrounds and environments.

285 **Input Modality Experiments.** To answer our third question, we perform a number of qualitative
 286 experiments testing different input modalities (detailed description in the Appendix). We find that
 287 MOO is able to generalize to masks generated from a variety of upstream input modalities, even
 288 under scenarios outside the training distribution including scenes with duplicate objects and clutter.

289 As the first qualitative example, Fig. 7(b) illustrates that VLM such as PaLI [51] can infer what
 290 object a human is pointing at, allowing OWL-ViT to generate an accurate mask of the object of interest.
 291 Secondly, OWL-ViT can also use visual query features instead of textual features to generate
 292 a mask, enabling images of target objects to act as conditioning for MOO, as shown in Figure 7(c).
 293 This modality is useful in cases where text-based mask generation due to ambiguity in natural lan-
 294 guage, or when target images are found in other scene contexts. We explore both the setting where
 295 target images are sourced from similar scenes or from diverse internet images. Finally, we show that
 296 MOO can interpret masks directly provided by humans via a GUI, as shown in Figure 7(d). This
 297 is useful in cases where both text-based and image-based mask generation is difficult, such as with
 298 duplicate or cluttered objects. MOO is robust to how upstream input masks were generated, and our
 299 preliminary results suggest interesting future avenues in the space of human-robot interaction.

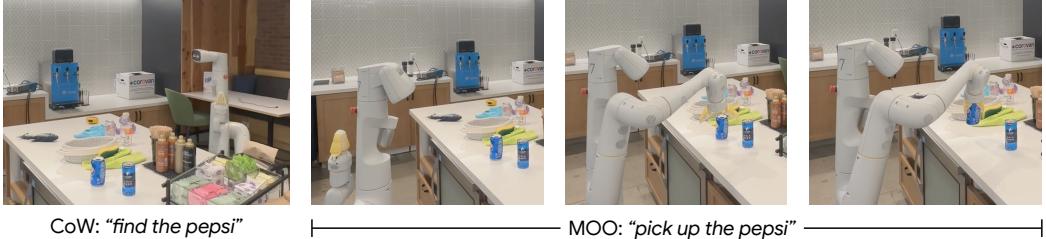


Figure 8: We present CoW-MOO, a system that combines an open-vocabulary object navigation by CoW [52] with open-world manipulation by MOO. Full videos are shown on the project’s website.

300 **MOO Ablations.** We conduct a number of ablations to assess the impact of the size and diversity of
 301 our dataset and the scale (in terms of number of parameters) of our model. In Table 1 we vary both
 302 the number of unique objects in the training set (reducing it from 106 to 53 to 16 unique training
 303 objects) and the number of total training episodes (reducing it by half – from 59051 training episodes
 304 to 29525) while keeping all objects in the dataset. We aim to vary these two axes independently to
 305 determine the impact of the overall size of the dataset vs its object diversity on the final results.
 306 Interestingly, we find that seen object performance is not affected by reducing object diversity, but
 307 generalization to unseen objects is very sensitive to object diversity.

308 Additionally, we investigate the impact of scaling model size. We train two smaller versions of MOO
 309 where we scale down the total number of layers and the layer width by a constant factor. The version
 310 of MOO that we use in our main experiments has 111M parameters, which, for the purpose of this
 311 ablation, we then reduce by an order of magnitude down to 10.2M and then by 5X again down to
 312 2.37M. Comparing different sizes of the model, we find significant drop offs in both “seen” (from
 313 92% to 54% and 39% respectively) and “unseen” object performance (from 75% to 50% and 13%;
 314 see Fig. 11 in the Appendix for a graph of the results). We also note that we could not make MOO
 315 larger than 111M parameters without increasing the latency on robot to an unacceptable level, but
 316 we expect continued performance gains with bigger models if latency requirements can be relaxed.

317 **Open-World Navigation and Manipulation.** Finally, we consider how such a system can be integrated
 318 with open-vocabulary object-based navigation. Coincidentally, there is an open-vocabulary
 319 object navigation algorithm called Clip on Wheels (CoW) [52]; we implement a variant of CoW and
 320 combine it with MOO, which we refer to as CoW-MOO. CoW handles open-vocabulary navigation
 321 to an object of interest, upon which MOO continues with manipulating the target object. This com-
 322 bination enables a truly open-world task execution, where the robot is able to first find an object
 323 it has never interacted with, and then successfully manipulate it to accomplish the task. We show
 324 example qualitative experiments in Fig. 8 and in the video of this system on the project’s website¹.

325 5 Conclusion and Limitations

326 In this paper we presented MOO, an approach for leveraging the rich semantic knowledge captured
 327 by vision-language models in robotic manipulation policies. We conduct 1,472 real world evalua-
 328 tions to show that MOO allows robots to generalize to novel instructions involving novel objects,
 329 enables greater robustness to visually challenging table textures and new environments, is amenable
 330 to multiple input modalities, and can be combined with open-vocabulary semantic navigation.

331 Despite the promising results, MOO has multiple important limitations. First, the object mask rep-
 332 resentation used by MOO may struggle in visually ambiguous cases, such as where objects are
 333 overlapping or occluded. Second, we expect the generalization of the policy to still be limited by the
 334 motion diversity of training data. For example, we expect that the robot may struggle to grasp novel
 335 objects with drastically different shapes or sizes than those seen in the training demonstration data,
 336 even with successful object localization. Third, instructions are currently expected to conform to a
 337 set of templates from which target objects and verbs can be easily separated. We expect this limita-
 338 tion could be lifted by leveraging an LLM to extract relevant properties from freeform instructions.
 339 Finally, MOO cannot currently handle complex object descriptions involving spatial relations, such
 340 as “the small object to the left of the plate.” Fortunately, we expect performance on tasks such as
 341 these to improve significantly as vision-language models continue to advance moving forward.

¹<https://robot-moo-anon.github.io/>

342 **References**

- 343 [1] N. NILSSON. Shakey the robot. *SRI International, Technical Note*, 323, 1984.
- 344 [2] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale,
345 M. Halpenny, G. Hoffmann, et al. Stanley: The robot that won the darpa grand challenge.
346 *Journal of field Robotics*, 23(9):661–692, 2006.
- 347 [3] P. Karkus, X. Ma, D. Hsu, L. P. Kaelbling, W. S. Lee, and T. Lozano-Pérez. Differentiable
348 algorithm networks for composable robot learning. *Robotics: Science and Systems (RSS)*,
349 2019.
- 350 [4] A. Curtis, X. Fang, L. P. Kaelbling, T. Lozano-Pérez, and C. R. Garrett. Long-horizon manip-
351 ulation of unknown objects via task and motion planning with estimated affordances. In *2022*
352 *International Conference on Robotics and Automation (ICRA)*, pages 1940–1946. IEEE, 2022.
- 353 [5] S. Levine, C. Finn, T. Darrell, and P. Abbeel. End-to-end training of deep visuomotor policies.
354 *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.
- 355 [6] A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, and T. Funkhouser. Learning synergies
356 between pushing and grasping with self-supervised deep reinforcement learning. In *2018*
357 *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4238–
358 4245. IEEE, 2018.
- 359 [7] S. Parisi, A. Rajeswaran, S. Purushwalkam, and A. Gupta. The unsurprising effectiveness of
360 pre-trained vision models for control. *arXiv preprint arXiv:2203.03580*, 2022.
- 361 [8] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta. R3m: A universal visual represen-
362 tation for robot manipulation. *arXiv preprint arXiv:2203.12601*, 2022.
- 363 [9] S. Nair, E. Mitchell, K. Chen, B. Ichter, S. Savarese, and C. Finn. Learning language-
364 conditioned robot behavior from offline data and crowd-sourced annotation. In *Conference
365 on Robot Learning*, pages 1303–1315. PMLR, 2021.
- 366 [10] E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, and C. Finn. Bc-z:
367 Zero-shot task generalization with robotic imitation learning. In *Conference on Robot Learn-
368 ing*, pages 991–1002. PMLR, 2021.
- 369 [11] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, K. Gopalakrishnan,
370 K. Hausman, A. Herzog, et al. Do as i can, not as i say: Grounding language in robotic
371 affordances. *Conference on Robot Learning (CoRL)*, 2022.
- 372 [12] M. Shridhar, L. Manuelli, and D. Fox. Perceiver-actor: A multi-task transformer for robotic
373 manipulation. *arXiv preprint arXiv:2209.05451*, 2022.
- 374 [13] S. Teller, M. R. Walter, M. Antone, A. Correa, R. Davis, L. Fletcher, E. Frazzoli, J. Glass,
375 J. P. How, A. S. Huang, et al. A voice-commandable robotic forklift working alongside hu-
376 mans in minimally-prepared outdoor environments. In *2010 IEEE International Conference
377 on Robotics and Automation*, pages 526–533. IEEE, 2010.
- 378 [14] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox. Posecnn: A convolutional neural network for
379 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017.
- 380 [15] S. Kumra and C. Kanan. Robotic grasp detection using deep convolutional neural networks.
381 In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages
382 769–776. IEEE, 2017.
- 383 [16] B. Zhou, P. Krähenbühl, and V. Koltun. Does computer vision matter for action? *Science
384 Robotics*, 4(30):eaaw6661, 2019.
- 385 [17] L. Yen-Chen, A. Zeng, S. Song, P. Isola, and T.-Y. Lin. Learning to see before learning to act:
386 Visual pre-training for manipulation. In *2020 IEEE International Conference on Robotics and
387 Automation (ICRA)*, pages 7286–7293. IEEE, 2020.

- 388 [18] B. Chen, A. Sax, G. Lewis, I. Armeni, S. Savarese, A. Zamir, J. Malik, and L. Pinto. Robust policies via mid-level visual representations: An experimental study in manipulation and
389 navigation. *arXiv preprint arXiv:2011.06698*, 2020.
- 390
- 391 [19] R. Shah and V. Kumar. Rrl: Resnet as representation for reinforcement learning. *arXiv preprint*
392 *arXiv:2107.03380*, 2021.
- 393
- 394 [20] I. Radosavovic, T. Xiao, S. James, P. Abbeel, J. Malik, and T. Darrell. Real-world robot
395 learning with masked visual pre-training. *arXiv preprint arXiv:2210.03109*, 2022.
- 396
- 397 [21] Y. J. Ma, S. Sodhani, D. Jayaraman, O. Bastani, V. Kumar, and A. Zhang. Vip: Towards
398 universal visual reward and representation via value-implicit pre-training. *arXiv preprint*
399 *arXiv:2210.00030*, 2022.
- 400
- 401 [22] F. Hill, S. Mokra, N. Wong, and T. Harley. Human instruction-following with deep reinforce-
402 ment learning via transfer-learning from text. *arXiv preprint arXiv:2005.09382*, 2020.
- 403
- 404 [23] C. Lynch and P. Sermanet. Grounding language in play. *arXiv preprint arXiv:2005.07648*,
405 2020.
- 406
- 407 [24] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Haus-
408 man, A. Herzog, J. Hsu, et al. RT-1: Robotics transformer for real-world control at scale. *arXiv*
409 *preprint arXiv:2212.06817*, 2022.
- 410
- 411 [25] Y. Jiang, A. Gupta, Z. Zhang, G. Wang, Y. Dou, Y. Chen, L. Fei-Fei, A. Anandkumar, Y. Zhu,
412 and L. Fan. Vima: General robot manipulation with multimodal prompts. *arXiv preprint*
413 *arXiv:2210.03094*, 2022.
- 414
- 415 [26] S. Stepputtis, J. Campbell, M. Phiellipp, S. Lee, C. Baral, and H. B. Amor. Language-
416 conditioned imitation learning for robot manipulation tasks. *ArXiv*, abs/2010.12083, 2020.
- 417
- 418 [27] O. Mees, L. Hermann, and W. Burgard. What matters in language conditioned robotic imitation
419 learning over unstructured data. *IEEE Robotics and Automation Letters*, 7(4):11205–11212,
420 2022.
- 421
- 422 [28] H. Liu, L. Lee, K. Lee, and P. Abbeel. Instruction-following agents with jointly pre-trained
423 vision-language models. *arXiv preprint arXiv:2210.13431*, 2022.
- 424
- 425 [29] L. Pinto and A. Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700
426 robot hours. In *IEEE international conference on robotics and automation (ICRA)*, 2016.
- 427
- 428 [30] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg. Dex-
429 net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp
430 metrics. *arXiv preprint arXiv:1703.09312*, 2017.
- 431
- 432 [31] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen. Learning hand-eye coordination
433 for robotic grasping with deep learning and large-scale data collection. *The International*
434 *Journal of Robotics Research*, 37(4-5), 2018.
- 435
- 436 [32] C. Finn and S. Levine. Deep visual foresight for planning robot motion. In *IEEE International*
437 *Conference on Robotics and Automation (ICRA)*, 2017.
- 438
- 439 [33] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly,
440 M. Kalakrishnan, V. Vanhoucke, et al. Scalable deep reinforcement learning for vision-based
441 robotic manipulation. In *Conference on Robot Learning*, pages 651–673. PMLR, 2018.
- 442
- 443 [34] S. Dasari, F. Ebert, S. Tian, S. Nair, B. Bucher, K. Schmeckpeper, S. Singh, S. Levine, and
444 C. Finn. Robonet: Large-scale multi-robot learning. In *Conference on Robot Learning*, 2019.
- 445
- 446 [35] S. Young, D. Gandhi, S. Tulsiani, A. Gupta, P. Abbeel, and L. Pinto. Visual imitation made
447 easy. In *CoRL*, 2020.
- 448
- 449 [36] Y. Chebotar, K. Hausman, Y. Lu, T. Xiao, D. Kalashnikov, J. Varley, A. Irpan, B. Eysenbach,
450 R. Julian, C. Finn, et al. Actionable models: Unsupervised offline reinforcement learning of
451 robotic skills. *arXiv preprint arXiv:2104.07749*, 2021.
- 452

- 435 [37] B.-H. Wu, S. Nair, R. Martín-Martín, L. Fei-Fei, and C. Finn. Greedy hierarchical variational
436 autoencoders for large-scale video prediction. *ArXiv*, abs/2103.04174, 2021.
- 437 [38] M. Shridhar, L. Manuelli, and D. Fox. Cliport: What and where pathways for robotic manipu-
438 lation. In *Conference on Robot Learning*, 2022.
- 439 [39] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object
440 detection and semantic segmentation. In *2014 IEEE Conference on Computer Vision and*
441 *Pattern Recognition*, pages 580–587, 2014. doi:10.1109/CVPR.2014.81.
- 442 [40] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time
443 object detection, 2015. URL <https://arxiv.org/abs/1506.02640>.
- 444 [41] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *2017 IEEE International*
445 *Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017. doi:10.1109/ICCV.2017.
446 322.
- 447 [42] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection.
448 In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007, 2017.
449 doi:10.1109/ICCV.2017.324.
- 450 [43] M. Minderer, A. A. Gritsenko, A. Stone, M. Neumann, D. Weissenborn, A. Dosovitskiy,
451 A. Mahendran, A. Arnab, M. Dehghani, Z. Shen, X. Wang, X. Zhai, T. Kipf, and N. Houlsby.
452 Simple open-vocabulary object detection with vision transformers. *ArXiv*, abs/2205.06230,
453 2022.
- 454 [44] X. Gu, T.-Y. Lin, W. Kuo, and Y. Cui. Open-vocabulary object detection via vision and lan-
455 guage knowledge distillation. In *International Conference on Learning Representations*, 2022.
456 URL <https://openreview.net/forum?id=1L3lnMbR4WU>.
- 457 [45] A. Kamath, M. Singh, Y. LeCun, G. Synnaeve, I. Misra, and N. Carion. Mdetr - modulated
458 detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF Inter-
459 national Conference on Computer Vision (ICCV)*, pages 1780–1790, October 2021.
- 460 [46] Y. Zhong, J. Yang, P. Zhang, C. Li, N. Codella, L. H. Li, L. Zhou, X. Dai, L. Yuan, Y. Li,
461 and J. Gao. Regionclip: Region-based language-image pretraining. In *Proceedings of the*
462 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16793–
463 16803, June 2022.
- 464 [47] M. Tan and Q. Le. Efficientnet: Rethinking model scaling for convolutional neural networks.
465 In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.
- 466 [48] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville. Film: Visual reasoning with a
467 general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*,
468 volume 32, 2018.
- 469 [49] M. S. Ryoo, A. Piergiovanni, A. Arnab, M. Dehghani, and A. Angelova. Tokenlearner: What
470 can 8 learned tokens do for images and videos? *arXiv preprint arXiv:2106.11297*, 2021.
- 471 [50] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polo-
472 sukhin. Attention is all you need. *Advances in neural information processing systems*, 30,
473 2017.
- 474 [51] X. Chen, X. Wang, S. Changpinyo, A. Piergiovanni, P. Padlewski, D. Salz, S. Goodman,
475 A. Grycner, B. Mustafa, L. Beyer, et al. Pali: A jointly-scaled multilingual language-image
476 model. *arXiv preprint arXiv:2209.06794*, 2022.
- 477 [52] S. Y. Gadre, M. Wortsman, G. Ilharco, L. Schmidt, and S. Song. Cows on pasture: Baselines
478 and benchmarks for language-driven zero-shot object navigation. *arXiv*, 2022.
- 479 [53] D. A. Pomerleau. Alvinn: An autonomous land vehicle in a neural network. *Advances in*
480 *neural information processing systems*, 1, 1988.
- 481 [54] M. Minderer, A. Gritsenko, A. Stone, M. Neumann, D. Weissenborn, A. Dosovitskiy, A. Ma-
482 hendran, A. Arnab, M. Dehghani, Z. Shen, et al. Simple open-vocabulary object detection with
483 vision transformers. *arXiv preprint arXiv:2205.06230*, 2022.

484 **Appendix**

485 **Imitation Learning and RT-1**

486 MOO builds upon a language-conditioned imitation learning setup. The goal of language-
 487 conditioned imitation learning is to learn a policy $\pi(a | \ell, o)$, where a is a robot action that should be
 488 applied given the current observation o and task instruction ℓ . To learn a language-conditioned policy
 489 π , we build on top of RT-1 [24], a recent robotics transformer-based model that achieves high lev-
 490 els of performance across a wide variety of manipulation tasks. RT-1 uses behavioral cloning [53],
 491 which optimizes π by minimizing the negative log-likelihood of an action a given the image ob-
 492 servations seen so far in the trajectory and the language instruction, using a demonstration dataset
 493 containing N demonstrations:

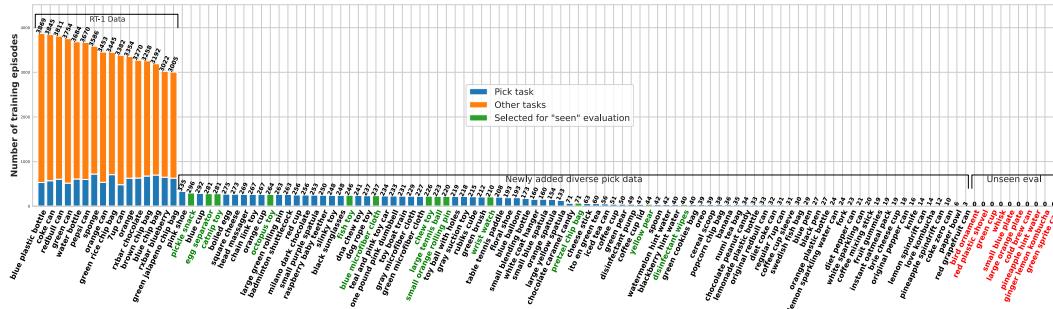
$$J(\pi) := \sum_{n=1}^N \sum_{t=1}^{T^{(n)}} \log \pi(a_t^{(n)} | \ell^{(n)}, \{o_j^{(n)}\}_{j=1}^t). \quad (1)$$

494 **Vision-Language Models**

495 In recent years, there has been a growing interest in developing models that can detect objects
 496 in images based on natural language queries. These models, known as vision-language models
 497 (VLMs), are enabling detectors to identify a wide range of objects based on natural language queries.
 498 Typically the text queries are tokenized and embedded in a high-dimensional space by a pre-trained
 499 language encoder, and the image is processed by a separate network to extract image features into
 500 the same embedding space as the text features. The language and image representations are then
 501 combined to make predictions of the bounding boxes and segmentation masks. Given a natural
 502 language query, q , and an image observation on which to run detection, o , these models aim to
 503 produce a set of embeddings for the image $f_i(o)$ with shape (height, width, feature dim) and an
 504 embedding of the language query $f_l(q)$ with shape feature dim such that $\text{logits} = f_i(o) \cdot f_l(q)$ gives
 505 a logit score map and is maximized at regions in o which correspond to the queries in q . Each
 506 image embedding location within $f_i(o)$ is also associated with a predicted bounding box or mask
 507 indicating the spatial extent of that object corresponding to $f_i(o)$. In this work, we use the Owl-ViT
 508 detector [54], which we discuss further in Sec. 3.4.

509 **Datasets**

510 We collect a focused collection of teleoperated demonstration data that focuses on increasing object
 511 diversity for the most efficient skill to collect data for, the picking task. Detailed dataset statistics
 512 across objects are shown in Figure 9.



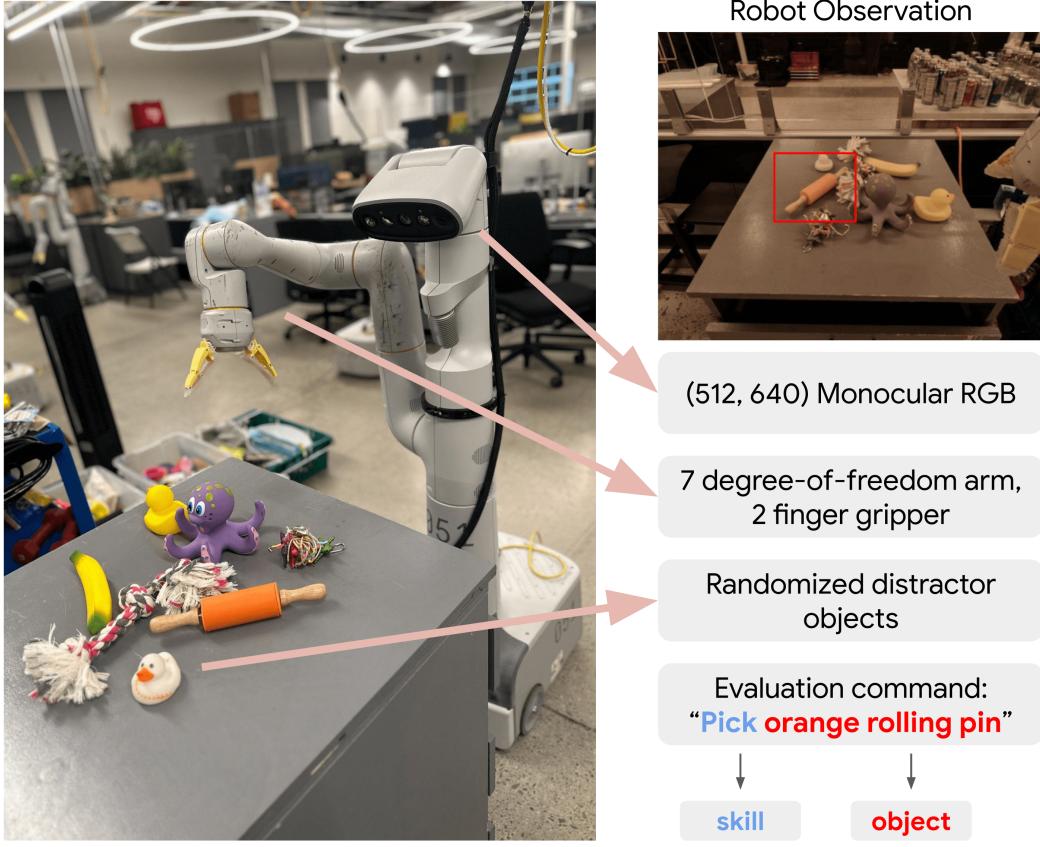


Figure 10: Image of our robot hardware and evaluation setting.

515 **Skills.** Our experiments evaluate the percent of successfully completed manipulation commands
 516 which include five skills: “pick”, “move near”, “knock,” “place upright,” and “place into” across
 517 a set of evaluation episodes. The definition of the tasks follows RT-1 [24]: For “pick”, success is
 518 defined as (1) grasping the specified object and (2) lifting the object at least 6 inches from the table
 519 top. For “move near”, success is defined as (1) grasping the specified object and (2) placing it within
 520 6 inches of the specified target object. For “knock”, success is defined as placing the specified object
 521 from an “upright” position onto its side. “Place upright” tasks are the inverse of “knock” and involve
 522 placing an object from its side into an upright position. Finally, “place into” tasks involve placing
 523 one object into another, such as an apple into a bowl.

524 **Robustness evaluation details.** We evaluate the robustness of MOO on a variety of visually chal-
 525 lenging scenarios with drastically different furniture and backgrounds, as shown in Figure 6; the
 526 results are reported in Figure 5. The first set of these difficult evaluation scenes introduces six eval-
 527 uations across five additional open-world objects that correspond to various household items that have
 528 not been seen at any point during training. The second set of difficult scenes introduces 14 eval-
 529 uations across two patterned tablecloths; these tablecloth textures are significantly more challenging
 530 than the plain gray counter-tops seen in the training demonstration dataset. Finally, the last set of
 531 difficult scenes include 14 evaluations across three new environments in natural kitchen and office
 532 spaces that were never present training. These new scenes simultaneously change the counter-top
 533 materials, backgrounds, lighting conditions, and distractor items.

534 **Input modality demonstration details.** We explore the ability of MOO to incorporate object-
 535 centric mask representations that are generated via different processes than the one used during
 536 training. During training, an OWL-ViT generates mask visual representations from textual prompts,
 537 as described in Section 3.2. We study whether MOO can successfully accomplish manipulation
 538 tasks given (1) a mask generated from a text caption from a generative VLM, (2) a mask generated
 539 from an image query instead of a text query, or (3) a mask directly provided by a human via a

Objects	Episodes per Object	Dataset Filtering		Pick	
		Seen objects	Unseen objects	Seen objects	Unseen objects
100%	100%	98	79		
50%	100%	92	75		
18%	100%	88	19		
100%	50%	46	38		
100%	10%	23	0		

Table 1: Performance of MOO relative to the amount of data used for training. Both data scale and data diversity are important.

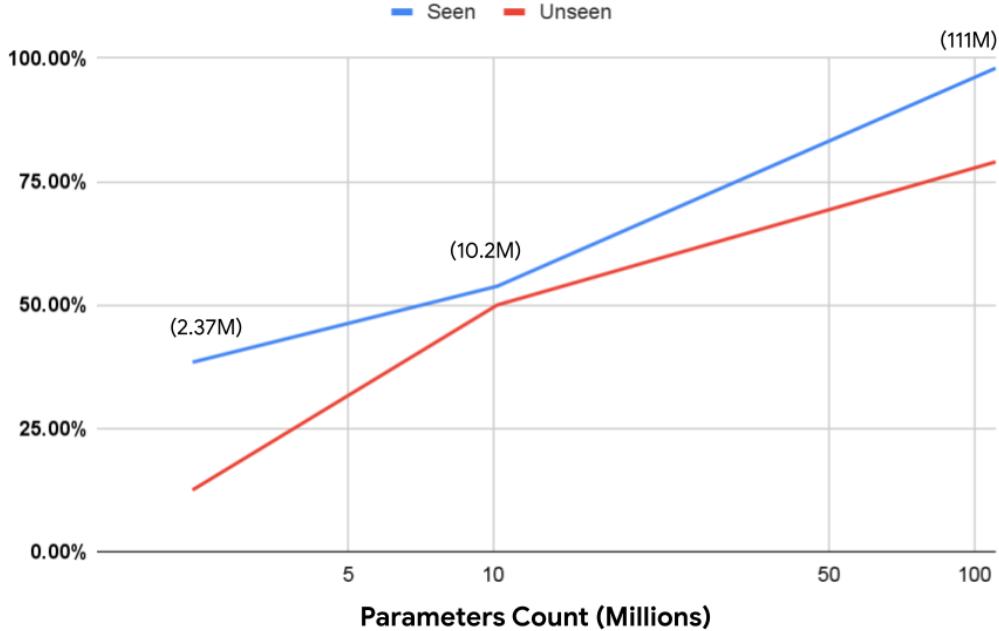


Figure 11: Pick success vs. model size. We see continuous improvements on both seen and unseen objects as we increase the number of parameters of our model architecture while keeping the data set size fixed. In comparison to our main model, we scaled down layer widths and depth by the same constant multiplier. We expect more performance gains at larger model capacity, yet are currently unable to scale further due to real time inference constraints on our robot.

540 GUI. For each of these cases, we implement different procedures for generating the object mask
 541 representation, which are then fed to the frozen MOO policy.

542 **Training data ablation.** We ablate the amount of data used to train MOO, and find that both data
 543 diversity and data scale are important, as shown in Table 1.

544 Prompts used

545 We use the following prompts to OWL-ViT detect our objects. All prompts were prefixed with the
 546 phrase “An image of a”.
 547 7up can → “white can of soda”
 548 banana → “banana”
 549 black pen → “black pen”
 550 blue chip bag → “blue bag of chips”
 551 blue pen → “blue pen”
 552 brown chip bag → “brown bag of chips”
 553 cereal scoop → “cereal scoop”
 554 chocolate peanut candy → “bag of candy snack”
 555 coffee cup → “coffee cup”
 556 coke can → “red can of soda”

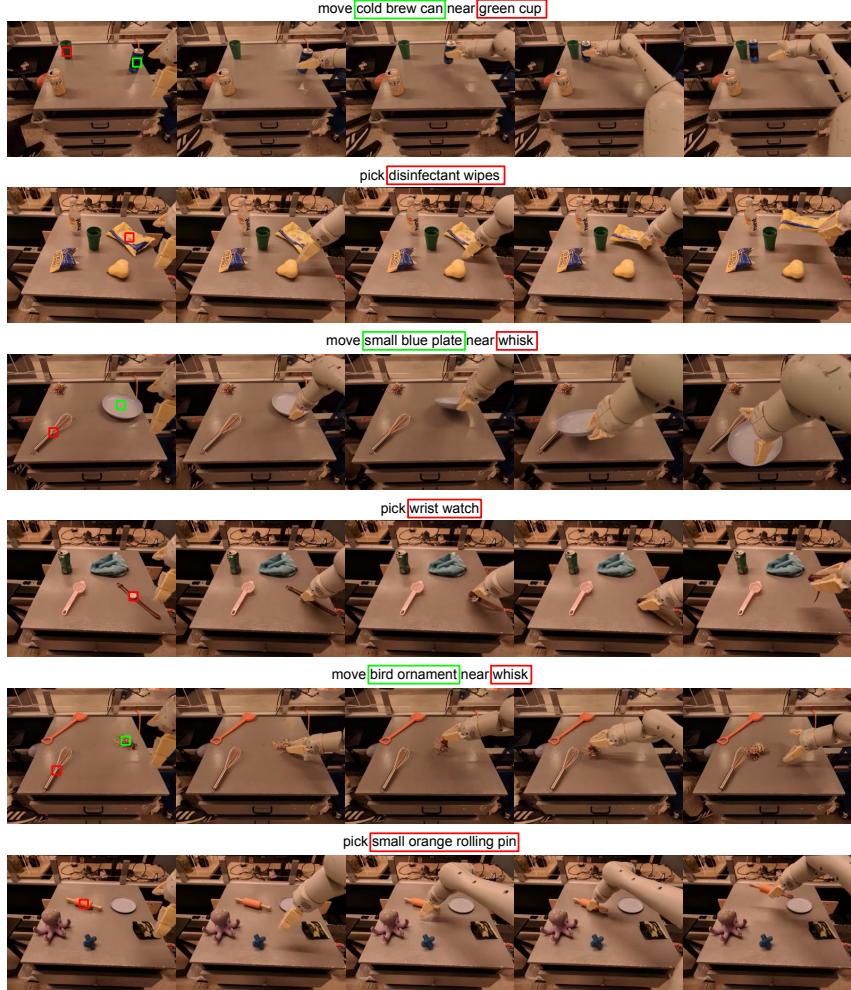


Figure 12: Example images of our policy detecting and grasping objects not seen during training time. The object detections are colored in correspondence to the text above the image, and the images are ordered left to right across time.

```

557 coke zero can → “can of soda”
558 disinfectant pump → “bottle”
559 fork → “fork”
560 green can → “green aluminum can”
561 green cookies bag → “green snack food bag”
562 green jalapeno chip bag → “green bag of chips”
563 green sprite can → “green soda can”
564 knife → “knife”
565 orange can → “orange aluminum can”
566 orange plastic bottle → “orange bottle”
567 oreo → “cookie snack food bag”
568 pepsi can → “blue soda can”
569 popcorn chip bag → “bag of chips”
570 pretzel chip bag → “bag of chips”
571 red grapefruit can → “red aluminum can”
572 redbull can → “skinny silver can of soda”
573 rxbar blueberry → “small blue rectangular snack food bar”
574 spoon → “spoon”
575 swedish fish bag → “bag of candy snack food”
576 water bottle → “clear plastic waterbottle with white cap”
577 white sparkling can → “aluminum can”

```

- 578 blue plastic bottle → “clear plastic waterbottle with white cap”
579 diet pepper can → “can of soda”
580 disinfectant wipes → “yellow and blue pack”
581 green rice chip bag → “green bag of chips”
582 orange → “round orange fruit”
583 paper bowl → “round bowl”
584 rxbar chocolate → “small black rectangular snack food bar”
585 sponge → “scrub sponge”
586 blackberry hint water → “clear plastic bottle with white cap”
587 pineapple hint water → “clear plastic bottle with white cap”
588 watermelon hint water → “clear plastic bottle with white cap”
589 regular 7up can → “can of soda”
590 lemonade plastic bottle → “clear plastic bottle with white cap”
591 diet coke can → “silver can of soda”
592 yellow pear → “yellow pear”
593 green pear → “green pear”
594 instant oatmeal pack → “flat brown pack of instant oatmeal”
595 coffee mixing stick → “small thin flat wooden popsicle stick”
596 coffee cup lid → “round disposable coffee cup lid”
597 coffee cup sleeve → “brown disposable coffee cup sleeve”
598 numi tea bag → “small flat packet of tea”
599 fruit gummies → “small blue bag of snacks”
600 chocolate caramel candy → “small navy bag of candy”
601 original redbull can → “can of energy drink with dark blue label”
602 cold brew can → “blue and black can”
603 ginger lemon kombucha → “yellow and tan aluminum can with brown writing”
604 large orange plate → “circular orange plate”
605 small blue plate → “circular blue plate”
606 love kombucha → “white and orange can of soda”
607 original pepper can → “dark red can of soda”
608 ito en green tea → “light green can of soda”
609 iced tea can → “black can of soda”
610 cheese stick → “yellow cheese stick in wrapper”
611 brie cheese cup → “small white cheese cup with wrapper”
612 pineapple spindrift can → “white and cyan can of soda”
613 lemon spindrift can → “white and brown can of soda”
614 lemon sparkling water can → “yellow can of soda”
615 milano dark chocolate → “white pack of snacks”
616 square cheese → “small orange rectangle packet ”
617 boiled egg → “small white egg in a plastic wrapper”
618 pickle snack → “small black and green snack bag”
619 red cup → “plastic red cup”
620 blue cup → “plastic blue cup”
621 orange cup → “plastic orange cup”
622 green cup → “plastic green cup”
623 head massager → “metal head massager with many wires”
624 chew toy → “blue and yellow toy with orange polka dots”
625 wrist watch → “wrist watch”
626 small orange rolling pin → “small orange rolling pin with wooden handles”
627 large green rolling pin → “large green rolling pin with wooden handles”
628 rubiks cube → “rubiks cube”
629 blue microfiber cloth → “blue cloth”
630 gray microfiber cloth → “gray cloth”
631 green microfiber cloth → “green cloth”
632 small blending bottle → “small turquoise and brown bottle”
633 large tennis ball → “large tennis ball”
634 table tennis paddle → “table tennis paddle”
635 octopus toy → “purple toy octopus”
636 pink shoe → “pink shoe”
637 floral shoe → “red and blue shoe”
638 whisk → “whisk”
639 orange spatula → “orange spatula”
640 small blue spatula → “small blue spatula”
641 large yellow spatula → “large yellow spatula”
642 egg separator → “large pink cooking spoon”

- 643 green brush → “green brush”
644 small purple spatula → “small purple spatula”
645 badminton shuttlecock → “shuttlecock”
646 black sunglasses → “black sunglasses”
647 toy ball with holes → “toy ball with holes”
648 red plastic shovel → “red plastic shovel”
649 bird ornament → “colorful ornament with blue and yellow confetti”
650 blue balloon → “blue balloon animal”
651 catnip toy → “small dark blue plastic cross toy”
652 raspberry baby teether → “red and green baby pacifier”
653 slinky toy → “gray metallic cylinder slinky”
654 dna chew toy → “big orange spring”
655 gray suction toy → “gray suction toy”
656 teal and pink toy car → “teal and pink toy car”
657 two pound purple dumbbell → “purple dumbbell”
658 one pound pink dumbbell → “pink dumbbell”
659 three pound brown dumbbell → “brown dumbbell”
660 dog rope toy → “white pink and gray rope with knot”
661 fish toy → “fish”
662 chain link toy → “skinny green rectangular toy”
663 toy boat train → “plastic toy boat”
664 white coat hanger → “white coat hanger”
665