Physically Grounded Vision-Language Models for Robotic Manipulation

Jensen Gao[1], Bidipta Sarkar[1], Fei Xia[2], Ted Xiao[2], Jiajun Wu[1],
Brian Ichter[2], Anirudha Majumdar[2,3], Dorsa Sadigh[1,2]

***Abstract*— Recent advances in vision-language models (VLMs) have led to improved performance on tasks such as visual question answering and image captioning. Consequently, these models are now well-positioned to reason about the physical world, particularly within domains such as robotic manipulation. However, current VLMs are limited in their understanding of the physical concepts (e.g., material, fragility) of common objects, which restricts their usefulness for robotic manipulation tasks that involve interaction and physical reasoning about such objects. To address this limitation, we propose PhysObjects, an object-centric dataset of 39.6K crowd-sourced and 417K automated physical concept annotations of common household objects. We demonstrate that fine-tuning a VLM on PhysObjects improves its understanding of physical object concepts, including generalization to held-out concepts, by capturing human priors of these concepts from visual appearance. We incorporate this physically grounded VLM in an interactive framework with a large language model-based robotic planner, and show improved planning performance on tasks that require reasoning about physical object concepts, compared to baselines that do not leverage physically grounded VLMs. We additionally illustrate the benefits of our physically grounded VLM on a real robot, where it improves task success rates. We release our dataset and provide further details and visualizations of our results at https://iliad.stanford.edu/pg-vlm/.**

I. Introduction

Large language models (LLMs) have shown great promise for converting language instructions into task plans for embodied agents [1], [2]. The fundamental challenge in applying LLMs for this is grounding them to the physical world, through sensory input such as vision. Prior work has made progress towards grounding LLMs by using vision-language models (VLMs) to indicate the presence of objects in a scene, or to provide feedback about occurrences in a scene [3]–[7]. However, vision could be used to further improve grounding by extracting more detailed scene information. For robotic manipulation, understanding physical concepts of objects, such as their material composition or their fragility, would help planners identify relevant objects to interact with, and affordances based on physical or safety constraints. For example, if a human wants a robot to get a cup of water, the robot should be able to determine if a cup already has water or something else in it. Also, the robot should handle the cup with greater caution if it is more fragile.

How can we use vision to reason about physical object concepts? Prior work has studied this problem using more traditional vision techniques, such as self-supervised learning on object interaction data. However, object interaction data can be challenging to collect when scaling up beyond a small set of objects in well-defined settings. While precise estimation of physical properties may sometimes be impossible without interaction data, humans can use their visual perception to reason at a high level about physical concepts without object interactions. For example, humans can reason that a glass cup is more fragile than a plastic bottle, and that it would be easier to use a bowl to hold water than a shallow plate. This reasoning is often based on prior semantic knowledge of visually similar objects, and can be done from static visual appearance alone.

Similarly, VLMs pre-trained using large-scale data have demonstrated broad visual reasoning abilities and generalization [8]–[13], and thus have the potential to physically reason about objects in a similar fashion as humans. Therefore, we propose to leverage VLMs as a scalable way of providing the kind of high-level physical reasoning that humans use to interact with the world, which can benefit a robotic planner, without the need for interaction data. The general and flexible nature of VLMs also removes the need to use separate task-specific vision models for physical reasoning. VLMs have already been commonly incorporated into robotic planning systems [3]–[7], [13], making them a natural solution for endowing physical reasoning into robotic planning.

However, while modern VLMs have improved significantly on tasks such as visual question answering (VQA), and there has been evidence of their potential for object-centric physical reasoning [14], we show in this work that their out-of-the-box performance for this still leaves much to be desired. Although VLMs have been trained on broad internet-scale data, this data does not contain many examples of object-centric physical reasoning. This motivates incorporating a greater variety and amount of such data when training VLMs. Unfortunately, prior visual datasets for physical reasoning are not well-suited for understanding common real-world objects, which is desirable for robotics. To address this, we propose PhysObjects, an object-centric dataset with human physical concept annotations of common household objects. Our annotations include categorical labels (e.g., object X is made of plastic) and preference pairs (e.g., object X is heavier than object Y).

Our main contributions are PhysObjects, a dataset of 39.6K crowd-sourced and 417K automated physical concept annotations of real household objects, and demonstrating that using it to fine-tune a VLM significantly improves physical reasoning. We show that our physically grounded VLM achieves improved test accuracy on our dataset, including on held-out physical concepts. Furthermore, to illustrate

[1]Stanford University, [2]Google DeepMind, [3]Princeton University. Contact: jenseng@stanford.edu.

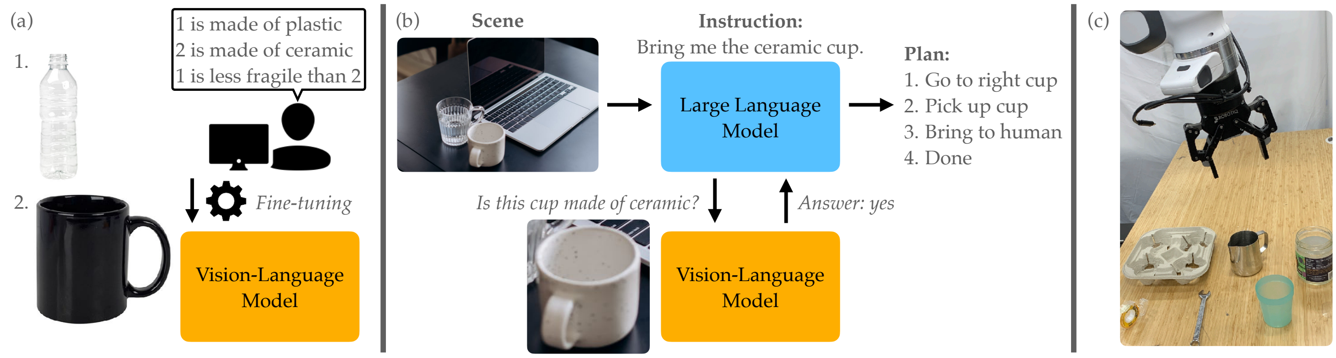


Fig. 1: (a) We collect physical concept annotations of common household objects for fine-tuning VLMs. (b) We use the fine-tuned VLM in an LLM-based robotic planning framework, where the LLM queries the VLM about physical concepts of objects in the scene, before producing a plan. (c) We evaluate LLM-generated plans on a real Franka Emika Panda robot.

the utility of improved physical reasoning for robotics, we incorporate our physically grounded VLM with an LLM-based robotic planner, where the LLM queries the VLM about physical concepts of objects in its scene. Our system achieves improved planning performance on tasks that require physical reasoning, compared to baselines that do not use physically grounded VLMs. Finally, we demonstrate the benefits of our physically grounded VLM for planning with a real robot, where its usage improves task success rates.

II. Related Work

We review prior work on physical reasoning, object attribute datasets, VLMs, using LLMs for robotic planning, and using LLMs and VLMs together in an interactive system.

Physical Reasoning. Prior works have studied estimating physical object properties from vision by learning from interaction data [15]–[17]. Other works focus on learning representations that capture physical concepts, rather than direct estimation [18], [19]. Unlike these works, we use pre-trained VLMs and human annotations as a more scalable alternative to learning from interaction. Mind's Eye investigates physical reasoning using LLMs [20], but relies on grounding using a simulator, which would be difficult to scale to the real world. VEC investigates physical reasoning with LLMs and VLMs [21], but reasons from text descriptions, while we reason from real images. OpenScene uses CLIP [22] to identify objects in scenes using properties such as material and fragility, but these results are only qualitative in nature [14]. In our work, we propose PhysObjects to better quantify and improve object-centric physical reasoning, and leverage this reasoning for robotic manipulation.

Object Attribute Datasets. There have been prior visual object attribute datasets with concepts included in PhysObjects, such as material and transparency [23]–[26]. However, they focus more on visual attributes such as color, while we focus on physical concepts. Physics 101 provides a dataset of object interaction videos and property measurements [16], but PhysObjects includes a greater variety of objects that are more relevant for household robotics.

Vision-Language Models. VLMs have made large improvements on multi-modal tasks such as VQA, by leveraging internet-scale image and text data [8]–[10], [12]. In our experiments, we use InstructBLIP [11] as our base VLM for fine-tuning and comparison, as it was the state-of-the-art open-source VLM at the time of our experiments. PaLM-E has shown strong performance on general visual-language tasks and robotic planning [13], but there has not been focused evaluation of it for physical reasoning. SuccessVQA fine-tunes VLMs on human data for success detection by treating it as a VQA task, and achieves better generalization than models designed specifically for success detection [27]. We similarly fine-tune VLMs on human data for physical reasoning by casting it as a VQA problem, to benefit from the generalization abilities and versatility of VLMs.

LLMs for Robotic Planning. Many recent works have used LLMs as robotic planners. SayCan uses visual value functions to provide affordances for grounding [2], but does not benefit from VLMs. Follow-up works have used VLMs for grounding LLM planners through object detection, or providing feedback about what has happened (e.g., success detection) [3]–[7]. Our work focuses on expanding the use of VLMs for grounding through physical reasoning, to let LLM-based planners perform tasks that require a deeper physical understanding of the world.

LLM/VLM Interaction. Our planning evaluation falls in the framework of Socratic Models [28], where large models interact with each other through text to perform tasks such as VQA [29], [30] and image captioning [31]. Most similar to our evaluation is Matcha, where an LLM receives a task instruction, obtains object-centric feedback from its environment, and uses this for task planning [32]. However, this work does not focus on visual feedback, as their evaluation is in a simulated environment where physical concepts are not visually observable. In contrast, we focus on physical reasoning from vision in real-world scenes.

III. PhysObjects Dataset

To benchmark and improve VLMs for object-centric physical reasoning, we propose PhysObjects, a dataset of 39.6K crowd-sourced and 417K automated physical concept annotations for images of real household objects.

Image Source. We use the publicly released challenge version of the EgoObjects dataset [33] as our image source. To our knowledge, this was the largest object-centric dataset of real images that was publicly released when constructing PhysObjects. The dataset consists of frames from egocentric videos in realistic household settings, which makes it particularly relevant for household robotics. It includes 117,424 images, 225,466 object bounding boxes with corresponding category labels from 277 object categories, and 4,203 object instance IDs. PhysObjects consists of physical concept annotations for a large subset of this image data. [1]

We construct random training, validation, and test sets based on object instance IDs. We split the dataset per object category to ensure each object category is represented in each set when possible. Our training, validation, and test sets consist of 73.0%, 14.8%, and 12.2% of objects, respectively.

Concept	Description
Mass	how heavy an object is
Fragility	how easily an object can be broken/damaged
Deformability	how easily an object can change shape without breaking
Material	what an object is primarily made of
Transparency	how much can be seen through an object
Contents	what is inside a container
Can Contain Liquid	if a container can be used to easily carry liquid
Is Sealed	if a container will not spill if rotated
Density (*held-out*)	how much mass per unit of volume of an object
Liquid Capacity (*held-out*)	how much liquid a container can contain

TABLE I: Our physical concepts and brief descriptions

Physical Concepts. We collect annotations for eight main physical concepts and two additional concepts reserved for held-out evaluation. We select concepts based on prior work and what we believe to be useful for robotic manipulation, but do not consider all such concepts. For example, we do not include *friction* because this can be challenging to estimate without interaction, and we do not include *volume* because this requires geometric reasoning, which we do not focus on.

Of our main concepts, three are continuous-valued and applicable to all objects: *mass*, *fragility*, and *deformability*. Two are also applicable to all objects, but are categorical: *material* and *transparency*. *Transparency* could be considered continuous, but we use discrete values of *transparent*, *translucent*, and *opaque*. The other three are categorical and applicable only to container objects: *contents*, *can contain liquid*, and *is sealed*. We define which object categories are containers, resulting in 956 container object instances.

Our two held-out concepts are *density*, which is continuous and applicable to all objects, and *liquid capacity*, which is continuous and applicable only to containers. We only collect test data for these held-out concepts. We list all concepts and their brief descriptions in Table I.

For categorical concepts, we define a set of labels for each concept. Annotations consist of a label specified for a given object and concept. For the concepts *material* and *contents*, when crowd-sourcing, we allow for open-ended labels if none of the pre-defined labels are applicable.

[1] We publicly release our dataset on our website. Because the EgoObjects license does not permit incorporating it into another dataset, we release our annotations separately from the image data.

For continuous concepts, annotations are preference pairs, where given two objects, an annotation indicates that either one object has a higher level of a concept, the objects have roughly *equal* levels, or the relationship is *unclear*. We use preferences because it is generally more intuitive for humans to provide comparisons than continuous values [34], [35]. This is especially true when annotating static images with physical concepts, where it is difficult to specify precise grounded values. For example, it would be difficult to specify the *deformability* of a sponge as a value out of 10. Comparisons have also been used to evaluate LLMs and VLMs for physical reasoning in prior work [21]. Therefore, the kind of grounding studied in PhysObjects for continuous concepts is only relational in nature.

Automatic Annotations. Before crowd-sourcing, we first attempt to automate as many annotations as possible, so that crowd-workers only annotate examples that cannot be easily automated. For categorical concepts, we assign concept values to some of the defined object categories in EgoObjects, such that all objects in a category are labeled with that value. For continuous concepts, we define *high* and *low* tiers for each concept, such that all objects from a *high* tier category have a higher level of that concept than all objects from a *low* tier category. Then, we automate preference annotations for all object pairs between the two tiers.

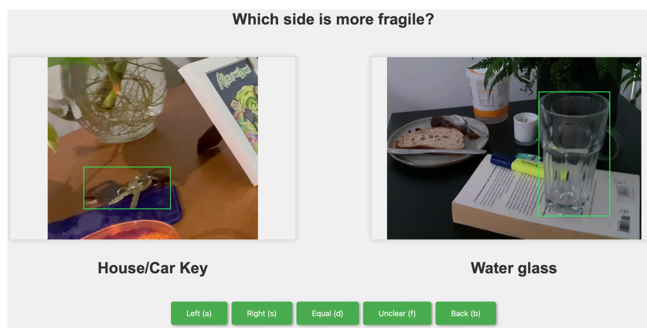


Fig. 2: Annotation UI for *fragility*. Here, the label is *right*, i.e., the *water glass* is more fragile than the *house/car key*.

Crowd-Sourcing Annotations. We obtain additional annotations via crowd-sourcing, using 573 crowd-workers on the Prolific platform. Crowd-workers use a web-based user interface (example for *fragility* shown in Fig. 2) where they are presented with object bounding boxes in the context of their overall image, and provide annotations using on-screen buttons or their keyboard. For categorical concepts, we collect annotations for the majority of objects that were not automatically annotated. For continuous concepts, because it is impractical to annotate every pair of objects in the dataset, we randomly sample pairs to annotate. We enforce that 20% of the sampled pairs are between objects of the same category, to prioritize understanding differences between objects of the same category. We collect annotations from three crowd-workers for each example. To promote high-quality data, we include attention checks as 10% of provided examples, which have known labels, and only keep data from annotators that achieve 80% accuracy on these.

	Most Common	Text Only	InstructBLIP	Single Concept FT (ours)	PG-InstructBLIP (ours)
Mass	42.2	73.3	62.2	**80.0**	**80.0**
Fragility	64.9	64.9	78.4	91.2	**94.6**
Deformability	46.5	62.8	67.4	**95.3**	93.0
Material	37.1	73.9	67.1	83.7	**84.6**
Transparency	77.6	82.2	85.8	89.4	**90.1**
Contents	39.5	50.9	35.1	81.6	**83.3**
Can Contain Liquid	56.3	**92.2**	59.4	84.4	87.5
Is Sealed	80.6	80.6	74.2	80.6	**87.1**
Average	55.6	72.6	66.2	85.8	**87.5**

TABLE II: Test accuracy for main concepts on crowd-sourced PHYSOBJECTS

Dataset Statistics. We crowd-source 39.6K annotations for 13.2K examples, and automate annotations for 417K additional examples. For crowd-sourced annotations, 93.7% of examples have at least 2/3 annotator label agreement, and 58.1% have unanimous agreement.

IV. PHYSICALLY GROUNDING VISION-LANGUAGE MODELS

Fine-Tuning VLMs. We work with the FlanT5-XXL [36] version of InstructBLIP [11]. InstructBLIP takes as input a single RGB image and text prompt, and predicts text as output. In our setup, we choose the model inputs to be a single bounding box of an object, and a question text prompt corresponding to each concept.

Learning From Preferences. Learning for categorical concepts amounts to maximum likelihood of annotated labels. However, it is not as straightforward to train a VLM on preferences for continuous concepts, because preference learning requires a continuous score. To do this with VLMs, which naturally have discrete text outputs, we prompt the VLM with questions that can be answered with *yes* or *no* for continuous concepts. Then, we extract the following score function:

$$s(o, c) = \frac{p(\text{yes} \mid o, c)}{p(\text{no} \mid o, c)}$$

where o is an object bounding box image, c is a concept, and $p(\cdot|o, c)$ is the likelihood under the VLM of text, conditioned on the object image and concept. We use this as our score function because it can take any non-negative value, and $\log s(o, c)$ has the intuitive interpretation as the difference of log-likelihoods between *yes* and *no*.[2] We then use the Bradley-Terry model [37] to estimate the probability of a human indicating that object o_1 has a higher value than object o_2 for concept c as:

$$P(o_1 > o_2 \mid c) = \frac{s(o_1, c)}{s(o_1, c) + s(o_2, c)}.$$

We assume a dataset $\mathcal{D}$ of preference annotations (o_1, o_2, c, y), where $y \in \{[1, 0], [0, 1], [0.5, 0.5]\}$ corresponds to if o_1 is preferred, o_2 is preferred, or if they are indicated to be equal. We then fine-tune the VLM by minimizing the following objective:

$$\begin{aligned}\mathcal{L}(\mathcal{D}) = -\,&\mathbb{E}_{(o_1, o_2, c, y)\sim\mathcal{D}}[y_1 \log P(o_1 > o_2 \mid c)\\ &+ y_2 \log(1 - P(o_1 > o_2 \mid c)].\end{aligned}$$

In practice, this is the binary cross-entropy objective where the logits for each object image o is the difference of log-likelihoods $\log s(o, c) = \log p(\text{yes} \mid o, c) - \log p(\text{no} \mid o, c)$.

V. EXPERIMENTAL RESULTS

We evaluate VLMs for physical reasoning using 1) test accuracy on PHYSOBJECTS, 2) planning accuracy on real scenes for physical reasoning tasks, and 3) task success rate on a real robot.

A. Dataset Evaluation

We refer to InstructBLIP fine-tuned on all main concepts in PHYSOBJECTS as Physically Grounded InstructBLIP, or PG-InstructBLIP. [3] We focus our evaluation on crowd-sourced examples, because as described in Section III, these were collected with the intent for their labels to not be discernible from object category information alone, and thus they are generally more challenging. We report test accuracy on these examples in Table II. Our baselines include *Most Common*, where the most common label in the training data is predicted, *Text Only*, where an LLM makes predictions using in-context examples from PHYSOBJECTS, but using object category labels instead of images, and InstructBLIP. We also compare to versions of InstructBLIP fine-tuned on single concept data. We find that PG-InstructBLIP outperforms InstructBLIP on all concepts, with the largest improvement on *contents*, which InstructBLIP has the most difficulty with. We also find that PG-InstructBLIP performs slightly better than the single concept models, suggesting possible positive transfer from using a single general-purpose model compared to separate task-specific models, although we acknowledge the improvement here is not extremely significant. PG-InstructBLIP also generally outperforms *Most Common* and *Text Only*, suggesting that our evaluation benefits from reasoning beyond dataset statistics, and from using vision.

[2] We experimented with other choices of score functions, and found that while all performed similarly with respect to test accuracy on PHYSOBJECTS, we found this score function to produce the most interpretable range of likelihoods for different responses, which we hypothesize to be beneficial for downstream planning.

[3] We release the model weights for PG-InstructBLIP on our website.

	Instruct-BLIP	PG-InstructBLIP (ours)
Density	54.2	**70.3**
Liquid Capacity	65.4	**73.0**
Average	59.8	**71.7**

TABLE III: Test accuracy for held-out concepts on crowd-sourced PHYSOBJECTS

Generalization Results. We additionally evaluate both InstructBLIP and PG-InstructBLIP on test data for our held-out concepts, which we report in Table III. We find that PG-InstructBLIP improves upon InstructBLIP by **11.9%**, despite having never seen these evaluated concepts nor object instances during fine-tuning. We believe this suggests that fine-tuning VLMs can offer possible generalization benefits to concepts that are related to those seen during fine-tuning.

	Instruct-BLIP	PG-InstructBLIP (ours)
Mass	55.6	**82.2**
Fragility	70.3	**83.8**
Deformability	76.7	**88.4**
Material	67.7	**83.4**
Transparency	81.5	**83.8**
Contents	32.5	**81.6**
Can Contain Liquid	56.3	**89.1**
Is Sealed	71.0	**80.6**
Average	64.0	**84.1**

TABLE IV: Test accuracy for main concepts with paraphrased prompts

In Table IV, we report results for main concepts on unseen paraphrased question prompts. We find that PG-InstructBLIP still outperforms InstructBLIP, with limited degradation from the original prompts, suggesting robustness to question variety from using a large pre-trained VLM.

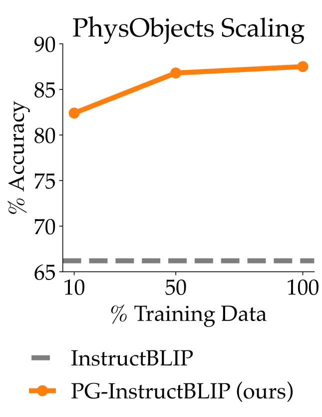


Fig. 3: Performance scaling with dataset size

Dataset Scaling. In Fig. 3, we illustrate how average performance scales with dataset size, by fine-tuning on different fractions of data from PHYSOBJECTS. Performance scales positively, but the models still benefit significantly from only 10% of our dataset, suggesting that the physical reasoning of VLMs can be improved with relatively small amounts of annotated data.

Additional Results. We include additional results in our Appendix (found on our website). These include showing that PG-InstructBLIP has limited degradation on general VQA benchmarks compared to InstructBLIP, suggesting that existing systems using VLMs can benefit from PHYSOBJECTS for physical reasoning, without sacrificing other reasoning abilities. We also include results using different question prompts, using a smaller version of InstructBLIP, evaluating on automatically annotated data, transfer to held-out concepts, and ablations on our fine-tuning process.

B. Real Scene Planning Evaluation

Next, we evaluate the efficacy of PG-InstructBLIP for robotic planning on unseen images of real scenes. We provide an example scene in Fig. 4. We evaluate on tasks with language instructions, and assume a library of primitive robotic operations with language descriptions.

Planning Framework. The LLM used in our planning framework is GPT-4 [38]. It is first given object detections in the scene, a list of primitives, and the task instruction, and then asks a VLM questions about objects in the scene. There are no constraints on the questions. Afterwards, the LLM either indicates the task is not possible, or produces a plan consisting of primitives to execute.

Fig. 4: Example scene in our planning evaluation

Task Category	No VLM	Instruct-BLIP	PG-InstructBLIP (ours)
Single Concept	36.8	68.4	**84.1**
Multi-Concept	27.8	27.8	**94.4**
Common Knowledge	35.7	78.6	**85.7**
Overall	33.3	56.9	**88.2**

TABLE V: Task plan accuracy on 51 real scenarios

Results. We report task planning accuracy using InstructBLIP and PG-InstructBLIP in Table V. We also compare to a planner that does not use VLM interaction for grounding. We evaluate on 51 task scenarios across 8 scenes, using a non-author human to evaluate task plans. We divide our task scenarios into three categories. *Single Concept* requires identifying objects using one physical concept, e.g., finding the heaviest object. *Multi-Concept* requires reasoning about multiple physical concepts, e.g., asking for a metal container that can hold water. This may include concepts outside of PHYSOBJECTS. *Common Knowledge* requires additional reasoning about common knowledge of objects, e.g., understanding the label of a container. While our tasks focus on physical concepts in PHYSOBJECTS, the LLM can ask questions about other concepts that may also be useful, particularly for *Common Knowledge* tasks.

PG-InstructBLIP outperforms InstructBLIP on all task categories, especially *Multi-Concept*. It does slightly better on *Common Knowledge*, suggesting that it can reason about non-PHYSOBJECTS concepts at a similar level as InstructBLIP. Using no VLM performs substantially worse than using VLM interaction, indicating that our tasks require additional grounding beyond object detection. We provide further details of results on our website.

C. Real Robot Evaluation

Lastly, we evaluate plans on real scenes using a Franka Emika Panda robot. We use a similar planner as in the previous section, but with different prompts and primitives. We assume a library of primitives for pick-and-place tasks. We evaluate on two scenes, with five tasks per scene, which we provide in Table VI. We report success rates using InstructBLIP and PG-InstructBLIP in Table VII. We ensure the primitives execute successfully, so our success rates only reflect plan quality.

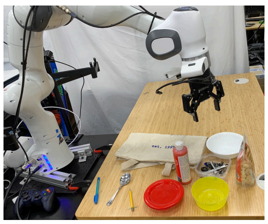
Scene Image	**Task Instructions**
	1) Move all objects that are not plastic to the side. 2) Find a container that has metals. Move all metal objects into that container. 3) Move all containers that can be used to carry water to the side. 4) Put the two objects with the least mass into the least deformable container. 5) Move the most fragile object to the side.
	1) Put all containers that can hold water to the side. 2) Put all objects that are not plastic to the side. 3) Put all objects that are translucent to the side. 4) Put the three heaviest objects to the side. 5) Put a plastic object that is not a container into a plastic container. Choose the container that you are most certain is plastic.

TABLE VI: Scene images and task instructions for our real robot evaluation

We find that using PG-InstructBLIP leads to successful robot executions more often than InstructBLIP. For example, when asked "Is this object not plastic?" about the ceramic bowl in Fig. 5a, InstructBLIP incorrectly assigns a likelihood of 0.89 to *yes*, while PG-InstructBLIP only assigns 0.18. However, when asked "Is this object translucent?" about the glass jar in Fig. 5b, both InstructBLIP and PG-InstructBLIP incorrectly assign likelihoods of 0.95 and 0.91 to *yes*, respectively. We note that while these questions relate to physical concepts in PhysObjects, neither are formatted like the training questions for PG-InstructBLIP. For example, the training prompt for *transparency* was "Is this object transparent, translucent, or opaque?". This suggests that despite using a large pre-trained VLM, PG-InstructBLIP may sometimes still fail due to out-of-distribution questions. We provide more results and visualizations on our website.

	Instruct-BLIP	PG-InstructBLIP (ours)
Scene 1	2/5	**5/5**
Scene 2	2/5	**4/5**
Overall	4/10	**9/10**

TABLE VII: Success rates for real robot evaluation

(a) Ceramic bowl (b) Glass jar

Fig. 5: Objects from our real robot evaluation

VI. Discussion

Summary. In this work, we propose PhysObjects, the first large-scale dataset of physical concept annotations of real household object images, and demonstrate that fine-tuning a VLM on it significantly improves its physical reasoning abilities, including on held-out physical concepts. We find that using the fine-tuned VLM for real-world robotic planning improves performance on tasks that require physical reasoning. We believe our work makes progress toward expanding the applicability of VLMs for robotics.

Limitations and Future Work. While we show PhysObjects can improve the physical reasoning of a VLM, it still makes errors relative to human judgment. Also, while our proposed methodology for continuous concepts improves relational grounding, which we show can be useful for robotic planning, the model outputs are not grounded in real physical quantities, which would be needed for some applications, e.g., identifying if an object is too heavy to be picked up. Future work can investigate incorporating data with real physical measurements to improve grounding.

While we believe the physical concepts in this work to have broad relevance for robotics, future work can expand on these for greater downstream applications. This could include expanding beyond physical reasoning, such as geometric reasoning (e.g., whether an object can fit inside a container), or social reasoning (e.g., what is acceptable to move off a table for cleaning). We believe our dataset is a first step towards this direction of using VLMs for more sophisticated reasoning in robotics.

Acknowledgments

This work was supported by NSF Awards 2132847, 1941722, and 2338203, ONR N00014-23-1-2355 and YIP, DARPA YFA, and Ford. We thank Minae Kwon, Siddharth Karamcheti, Suvir Mirchandani, and other ILIAD lab members for helpful discussions and feedback, and Siddharth Karamcheti for helping to set up the real robot evaluation.

REFERENCES

[1] Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In *International Conference on Machine Learning*, pages 9118–9147. PMLR, 2022.

[2] Brian Ichter, Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, Dmitry Kalashnikov, Sergey Levine, Yao Lu, Carolina Parada, Kanishka Rao, Pierre Sermanet, Alexander T Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Mengyuan Yan, Noah Brown, Michael Ahn, Omar Cortes, Nicolas Sievers, Clayton Tan, Sichun Xu, Diego Reyes, Jarek Rettinghouse, Jornell Quiambao, Peter Pastor, Linda Luu, Kuang-Huei Lee, Yuheng Kuang, Sally Jesmonth, Kyle Jeffrey, Rosario Jauregui Ruano, Jasmine Hsu, Keerthana Gopalakrishnan, Byron David, Andy Zeng, and Chuyuan Kelly Fu. Do as i can, not as i say: Grounding language in robotic affordances. In *6th Annual Conference on Robot Learning*, 2022.

[3] Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, Pierre Sermanet, Tomas Jackson, Noah Brown, Linda Luu, Sergey Levine, Karol Hausman, and Brian Ichter. Inner monologue: Embodied reasoning through planning with language models. In *6th Annual Conference on Robot Learning*, 2022.

[4] Boyuan Chen, Fei Xia, Brian Ichter, Kanishka Rao, Keerthana Gopalakrishnan, Michael S Ryoo, Austin Stone, and Daniel Kappler. Open-vocabulary queryable scene representations for real world planning. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11509–11522. IEEE, 2023.

[5] Wenlong Huang, Fei Xia, Dhruv Shah, Danny Driess, Andy Zeng, Yao Lu, Pete Florence, Igor Mordatch, Sergey Levine, Karol Hausman, et al. Grounded decoding: Guiding text generation with grounded models for embodied agents. *Advances in Neural Information Processing Systems*, 36, 2023.

[6] Satvik Sharma, Huang Huang, Kaushik Shivakumar, Lawrence Yunliang Chen, Ryan Hoque, brian ichter, and Ken Goldberg. Semantic mechanical search with large vision and language models. In *7th Annual Conference on Robot Learning*, 2023.

[7] Jimmy Wu, Rika Antonova, Adam Kan, Marion Lepert, Andy Zeng, Shuran Song, Jeannette Bohg, Szymon Rusinkiewicz, and Thomas Funkhouser. Tidybot: Personalized robot assistance with large language models. *Autonomous Robots*, 2023.

[8] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022.

[9] Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, Alexander Kolesnikov, Joan Puigcerver, Nan Ding, Keran Rong, Hassan Akbari, Gaurav Mishra, Linting Xue, Ashish V Thapliyal, James Bradbury, Weicheng Kuo, Mojtaba Seyedhosseini, Chao Jia, Burcu Karagol Ayan, Carlos Riquelme Ruiz, Andreas Peter Steiner, Anelia Angelova, Xiaohua Zhai, Neil Houlsby, and Radu Soricut. PaLI: A jointly-scaled multilingual language-image model. In *The Eleventh International Conference on Learning Representations*, 2023.

[10] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*, pages 19730–19742, 2023.

[11] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[12] Shikun Liu, Linxi Fan, Edward Johns, Zhiding Yu, Chaowei Xiao, and Anima Anandkumar. Prismer: A vision-language model with multi-task experts. *Transactions on Machine Learning Research*, 2024.

[13] Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. PaLM-e: An embodied multimodal language model. In *Proceedings of the 40th International Conference on Machine Learning*, pages 8469–8488, 2023.

[14] Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. Openscene: 3d scene understanding with open vocabularies. In *CVPR*, 2023.

[15] Jiajun Wu, Ilker Yildirim, Joseph J Lim, Bill Freeman, and Josh Tenenbaum. Galileo: Perceiving physical object properties by integrating a physics engine with deep learning. *Advances in neural information processing systems*, 28, 2015.

[16] Jiajun Wu, Joseph J Lim, Hongyi Zhang, Joshua B Tenenbaum, and William T Freeman. Physics 101: Learning physical object properties from unlabeled videos. In *BMVC*, volume 2, page 7, 2016.

[17] Yunzhu Li, Toru Lin, Kexin Yi, Daniel Bear, Daniel L.K. Yamins, Jiajun Wu, Joshua B. Tenenbaum, and Antonio Torralba. Visual grounding of learned physical models. In *ICML*, 2020.

[18] Michael Janner, Sergey Levine, William T. Freeman, Joshua B. Tenenbaum, Chelsea Finn, and Jiajun Wu. Reasoning about physical interactions with object-oriented prediction and planning. In *International Conference on Learning Representations*, 2019.

[19] Zhenjia Xu, Jiajun Wu, Andy Zeng, Joshua B Tenenbaum, and Shuran Song. Densephysnet: Learning dense physical object representations via multi-step dynamic interactions. In *Robotics: Science and Systems (RSS)*, 2019.

[20] Ruibo Liu, Jason Wei, Shixiang Shane Gu, Te-Yen Wu, Soroush Vosoughi, Claire Cui, Denny Zhou, and Andrew M. Dai. Mind's eye: Grounded language model reasoning through simulation. In *The Eleventh International Conference on Learning Representations*, 2023.

[21] Lei Li, Jingjing Xu, Qingxiu Dong, Ce Zheng, Xu Sun, Lingpeng Kong, and Qi Liu. Can language models understand physical concepts? In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

[22] Alec Radford, Jong Wook Kim, Chris Hallacy, A. Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[23] Genevieve Patterson and James Hays. Coco attributes: Attributes for people, animals, and objects. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI 14*, pages 85–100. Springer, 2016.

[24] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123:32–73, 2017.

[25] Khoi Pham, Kushal Kafle, Zhe Lin, Zhihong Ding, Scott Cohen, Quan Tran, and Abhinav Shrivastava. Learning to predict visual attributes in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13018–13028, 2021.

[26] Vignesh Ramanathan, Anmol Kalia, Vladan Petrovic, Yi Wen, Baixue Zheng, Baishan Guo, Rui Wang, Aaron Marquez, Rama Kovvuri, Abhishek Kadian, Amir Mousavi, Yiwen Song, Abhimanyu Dubey, and Dhruv Mahajan. Paco: Parts and attributes of common objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7141–7151, June 2023.

[27] Yuqing Du, Ksenia Konyushkova, Misha Denil, Akhil Raju, Jessica Landon, Felix Hill, Nando de Freitas, and Serkan Cabi. Vision-language models as success detectors. In *Proceedings of The 2nd Conference on Lifelong Learning Agents*, pages 120–136, 2023.

[28] Andy Zeng, Maria Attarian, Brian Ichter, Krzysztof Marcin Choromanski, Adrian Wong, Stefan Welker, Federico Tombari, Aveek Purohit, Michael S Ryoo, Vikas Sindhwani, Johnny Lee, Vincent Vanhoucke, and Pete Florence. Socratic models: Composing zero-shot multimodal reasoning with language. In *The Eleventh International Conference on Learning Representations*, 2023.

[29] Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Yumao Lu, Zicheng Liu, and Lijuan Wang. An empirical study of gpt-3 for few-shot knowledge-based vqa. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 3081–3089, 2022.

[30] Zhenwei Shao, Zhou Yu, Meng Wang, and Jun Yu. Prompting large language models with answer heuristics for knowledge-based visual question answering. In *Computer Vision and Pattern Recognition (CVPR)*, pages 14974–14983, 2023.

[31] Deyao Zhu, Jun Chen, Kilichbek Haydarov, Xiaoqian Shen, Wenxuan Zhang, and Mohamed Elhoseiny. Chatgpt asks, blip-2 answers:

Automatic questioning towards enriched visual descriptions. *arXiv preprint arXiv:2303.06594*, 2023.

[32] Xufeng Zhao, Mengdi Li, Cornelius Weber, Muhammad Burhan Hafez, and Stefan Wermter. Chat with the environment: Interactive multimodal perception using large language models. *arXiv preprint arXiv:2303.08268*, 2023.

[33] Meta. Egoobjects dataset. https://ai.facebook.com/datasets/egoobjects-dataset/, Last accessed on 2023-05-28.

[34] Dorsa Sadigh, Anca D. Dragan, S. Shankar Sastry, and Sanjit A. Seshia. Active preference-based learning of reward functions. In *Proceedings of Robotics: Science and Systems (RSS)*, July 2017.

[35] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[36] Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*, 2022.

[37] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[38] OpenAI. Gpt-4 technical report, 2023.

[39] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip. July 2021.

[40] Dongxu Li, Junnan Li, Hung Le, Guangsen Wang, Silvio Savarese, and Steven C. H. Hoi. Lavis: A library for language-vision intelligence, 2022.

[41] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Gray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[42] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913, 2017.

[43] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pages 3195–3204, 2019.

[44] Matthias Minderer, Alexey Gritsenko, Maxim Neumann Austin Stone, Dirk Weissenborn, Aravindh Mahendran Alexey Dosovitskiy, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, Xiao Wang, Xiaohua Zhai, Thomas Kipf, and Neil Houlsby. Simple open-vocabulary object detection with vision transformers. *ECCV*, 2022.

[45] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[46] Yixin Lin, Austin S. Wang, Giovanni Sutanto, Akshara Rai, and Franziska Meier. Polymetis. https://facebookresearch.github.io/fairo/polymetis/, 2021.