Affordance Perception by a Knowledge-Guided Vision-Language Model with Efficient Error Correction*

G.J. Burghouts, M. Schaaphok, M. van Bekkum, W. Meijer, F. Hillerström, J. van Mil

TNO, 2597 AK The Hague, The Netherlands

Abstract. Mobile robot platforms will increasingly be tasked with activities that involve grasping and manipulating objects in open world environments. Affordance understanding provides a robot with means to realise its goals and execute its tasks, e.g. to achieve autonomous navigation in unknown buildings where it has to find doors and ways to open these. In order to get actionable suggestions, robots need to be able to distinguish subtle differences between objects, as they may result in different action sequences: doorknobs require grasp and twist, while handlebars require grasp and push. In this paper, we improve affordance perception for a robot in an open-world setting. Our contribution is threefold: (1) We provide an affordance representation with precise, actionable affordances; (2) We connect this knowledge base to a foundational vision-language models (VLM) and prompt the VLM for a wider variety of new and unseen objects; (3) We apply a human-in-the-loop for corrections on the output of the VLM. The mix of affordance representation, image detection and a human-in-the-loop is effective for a robot to search for objects to achieve its goals. We have demonstrated this in a scenario of finding various doors and the many different ways to open them.

Keywords: Open World Robotics · Perception · Affordance · Vision-language model · Knowledge.

1 Introduction

Mobile robot platforms will increasingly be tasked with activities that involve grasping and manipulating objects in open world environments in order to reach a goal [35]. Interacting in such an open world poses challenges for a robot. In order to pursue its goal, the robot needs to take advantage of the actionable possibilities of objects that it encounters, e.g. lift it to get it out of the way, open it to see what is inside, etc. In contrast to a closed world or rigidly structured environment, in an open world the robot needs to be able to adapt to unforeseen events and interact with unknown objects [4]. Effective interaction with objects

* Supported by TNO ERP APPL.AI program.

is based on the perception of their affordances [10,3], what the object offers or provides to an user [11]. A button on a door affords pushing, while a handle may also afford pushing [26]. Understanding of affordance allows reasoning about object uses: what possibilities for interaction does the object offer and how can an object be handled and put to use? Understanding affordance comes from the combination of perception and prior knowledge of the world [11]. Perception provides clues about possible affordances: what object does the robot see, what properties does it have and what does the object seemingly allow? Combined with prior knowledge this leads to affordance understanding. Potential actions can then be deduced from models that describe how a robot may make use of some property (take action) based on those perceived affordances [3,4]. However, such models do not scale well, because a lot of manual engineering is required to extend them with new object classes and with new contexts for existing object classes.

In order to generate actionable suggestions, the robot needs to distinguish subtle differences between objects, as they may result in different action sequences: doorknobs require grasp and turn, while door handles require grasp and push. Sources that provide this knowledge about actions may include explicitly structured (semantic) knowledge bases [31] or e.g. foundational language models like ChatGPT [28]. Approaches that rely on explicit semantic affordance-action models are precise [3,4]: when one determines an affordance, one can deduce an associated action. This makes for actionable affordances, i.e. affordances with perceivable action possibilities. Similar to explicit knowledge models for perception however, this requires manual engineering and does not scale well. On the other hand, foundational language models scale well and embed a wealth of fine-grained knowledge. They can provide precise information on object affordances, as is demonstrated by ChatGPT when prompted with "Give me a visual description of a door handle and a door knob and give a step-by-step action list on how to open a door". The ChatGPT-generated text clearly outlined their visual properties and the steps for using both: "grasp and push or pull" for door handles and "grasp and twist" for door knobs [27].

Surprisingly, we discover in this paper that foundational vision-language models (VLMs) do not exhibit the same fine-grained distinctions as the language models. VLMs such as GLIP [19,37] are very flexible and are therefore promising also for affordance perception. They embed knowledge about a wide variety of objects and allow open vocabulary object prompting, which makes their use scalable for new objects and new contexts. However, we find that VLMs models lack the necessary fine-grained semantic differences between objects, such as a doorknob vs. a handlebar, where each requires a different action. This lack of discrimination, makes it difficult to deduce the right action for the robot. To enable the required fine-grained perception, the VLM needs to be fine-tuned. This should require little additional annotations and retraining, because the robot needs to be deployed again quickly to continue its task. We propose a solution by correcting the VLM efficiently for confused objects by a human-in-the-loop.

In this paper, we bring together knowledge representation and foundation models, to achieve a best of both worlds, complemented with a human-in-the-loop that refines missing knowledge. We improve affordance understanding in an open-world setting for a robot whose training and world model includes some, but does not have complete world knowledge. Our contribution is threefold: (1) We provide an affordance representation in a knowledge base that represents precise, actionable affordances for a limited set of relevant objects; (2) We connect this knowledge base to a VLM and prompt it for a wider variety of new and unseen objects similar to those in the knowledge base; (3) We apply a human-in-the-loop to correct the VLM where its finegrained discrimination is lacking.

2 Related Work

2.1 Affordance modelling

There are multiple proposed formalisations of affordances for robotics [38] [22] [9], all relating to some extent objects, actions and effects to an agent and its action capability. Functional representation of affordances should have the ability for recognition, cognitive and conceptual understanding, how to use and operate an object, invention of new objects/tools for a function [16]. Features/characteristics of objects can be taken into account when detecting affordances, but background knowledge is required [24], while affordances may be hidden from current perspective [24]. The most common used formalization of affordances is that a potential action exists that could generate the effect, if applied to a specific object in the environment. A single interaction will then produce an instance of this triplet (object, action, effect). The surveys by [2], [36], [3], [24] provide overviews of the work on affordances for robotics. Affordances can be used for different purposes, such as planning when one knows the desired effects and the objects and one needs to predict the action or for effect prediction when one knows both the object and the action [12]. A practical approach for affordances in robotics focuses on affordance templating[15], [14] for more complex manipulation tasks, but this can only handle situation envisioned at design time. Other approaches include deep learning [8], reinforcement learning and value functions [1] [12].

2.2 Affordance Perception

Most of the affordance perception research, focuses on task-driven object detection [20,30,32], i.e., finding objects that can cut the tape around a closed box. A context-based Gated Graph Neural Network uses the detected objects in an image and selects the ones that are suitable for a task, given the appearance of an object and the global context of all objects in the scene [30]. The category-based approach TOIST extracts the objects from a language description of the task [20]. The methods in [20,30] leveraged only a limited amount of external knowledge. Therefore, a recent trend is to use large language models (LLMs)

to extend the knowledge by a huge amount. In [34] a joint language and vision model was used for object detection. They modeled the detection via language embeddings from Wiktionary object descriptions. The state-of-the-art for affordance perception is the end-to-end learning approach presented in [32]. The approach uses a LLM with chain of thought (CoT) for extracting knowledge about which objects and properties can afford to solve a task. The extracted objects and properties are used to fine-tune an object detection model. This knowledge-conditioned model learns to both detect objects and to recognize visual attributes that provide affordances, e.g., a sharp object can cut. Instead of a comprehensive learning scheme during preparation, our goal is to be flexible in an open world, with new tasks that involve new objects. Therefore, instead of pre-learning for affordance perception, we focus on efficient learning on the job.

2.3 Open-Vocabulary Object Detection

For open world robotics, it is key to have open-vocabulary perception models. In computer vision, such methods have rapidly advanced since CLIP [29]. Here it was shown that contrastive pretraining on a massive number of noisy image-text pairs can result in models with impressive zero-shot generalization capabilities. CLIP is able to classify images, where we are interested in localizing objects in images, a task known as object detection. For object detection, ViLD [13] and OWL-ViT [25] extended CLIP with localization capabilities. GLIP [19] and GLIPv2 [37] followed a different strategy and proposed an architecture and pretraining strategy specifically designed for object detection. Recently GLIPv2 set a new state of the art performance on MS-COCO [21] for zero-shot tasks, i.e. unseen objects. GLIPv2 performs well at detecting almost any everyday object, therefore we take GLIP as a starting point. To search for the right objects that can provide specific affordances to solve the task at hand (e.g. opening a door via a push bar), we leverage a knowledge base that captures information about affordances and objects. Such knowledge feeds into the model via task-specific textual prompts. Inspired by [7,6], we improve the discrimination by validating known spatial relations between the objects in a scene. One problem with GLIPv2 and similar recent models is their struggle with fine-grained or less common objects or viewpoints [5]. We improve the model's ability to distinguish fine-grained objects by efficient learning on the job.

3 Method

3.1 Architecture

We propose a modular system as shown in 1, consisting of (a) a knowledge base in TypeDB [33] for affordance representation, (b) a neuro-symbolic program in Scallop [17] for logical constraints, (c) a GLIP module for localizing objects and (d) a human-in-the-loop for label refinement.

The knowledge base provides labels for relevant object classes to the object detection module, the neuro-symbolic program provides (spatial) constraints on

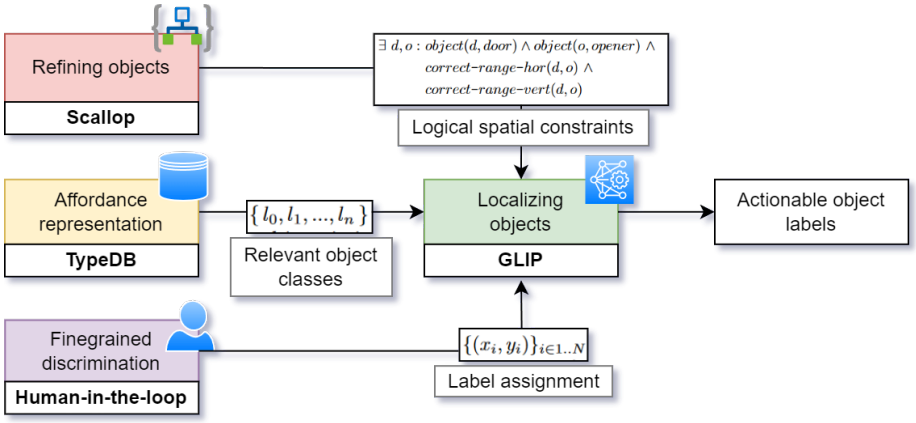

Fig. 1: Affordance detection architecture

localization of objects and the human-in-the-loop will provide label refinement and correction on identified objects. The output of the object detection is a set of actionable object labels.

3.2 Affordance representation

To find objects that offer the desired affordance to reach the specified goal, a proper representation of affordances is required. We present a representation of affordances using three relations: the effect relation, the affordance relation and the action relation. The effect relation connects a property to an object. The affordance relation connects an action to an effect and the action relation relates to a direct object. Figure 2 shows an example of this representation, where the effect is a relation between the object 'door' and the property 'accessibility', the affordance is the relation between the action 'push down' and the effect and the action relation 'push down' is related to the direct object 'handle'.

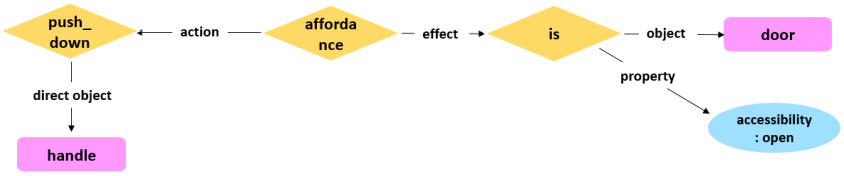

Fig. 2: Basic affordance representation using three relations, effect relation, affordance relation and action relation

We designed this representation to provide more flexibility in modelling complex affordances and to achieve more flexible, actionable instructions. The representation allows us to (1) explicitly model whether action and effect are on the same or on different objects, (2) represent more detailed and complex affordance structures, including indirect object and other agents (see Figure 3). The model structure also allows for (3) different affordances leading to the same effect, which supports quick identification of different affordances, (4) one action to have multiple effects, where it can distinguish between (a) both effects happening jointly or (b) one of the effects taking place. Finally, we can model (5) effects as a verb (e.g. the effect is that person A "holds" a cup), (6) probabilities of an effect taking place, (7) chains of actions to obtain the desired effect (where the "action" relation is replaced by an "action chain" relation). The affordance representations are modelled in a TypeDB knowledge graph. Manually engineering these graphs would be too labour-intensive and inefficient. We can however create baseline versions of these graphs by semi-automated generation (and then manual curation) based on common-sense knowledge available in LLMs ([18]). In order to avoid scalability problems and inference perfomance issues, we envision creating domain- and/or use-case-specific models (e.g. office environment, industrial site, etc.).

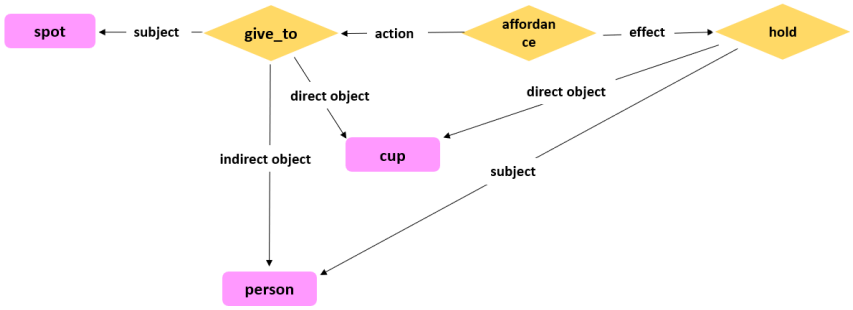

Fig. 3: Affordance representation, where the robot SPOT gives a cup to another person, with the effect that this person holds the cup.

3.3 Localizing Objects with Affordances

The goal is to find objects that offer the desired affordance. For instance, if the robot is tasked to open a door, to localize a push bar or handle in an image. There can be multiple means to open doors, e.g. a push bar, a handle, a knob, a button. The set of relevant objects $\{ l_0, l_1, ..., l_n \}$ is taken from a knowledge base, as proposed in [7]. To localize these objects in image, we prompt the GLIP model [19] in the string format as proposed in [19,37]:

$$<l_0>.\ <l_1>.\ \cdots <l_n> \tag{1}$$

In accordance with [5], the model is not able to distinguish between fine-grained object labels. In Figure 4, results are shown for three different scenes with various door and openers. for example, the door handle is mislabeled as a knob. For a robot, this mistake is crucial, because it will select the wrong action to open it. A knob should be turned, whereas the handle should be pushed downwards. There are many false positives too, respectively the handrail is mistaken for a push bar, the window on the side is mistaken for a knob, and the window frame in the background is mistaken for a door handle.

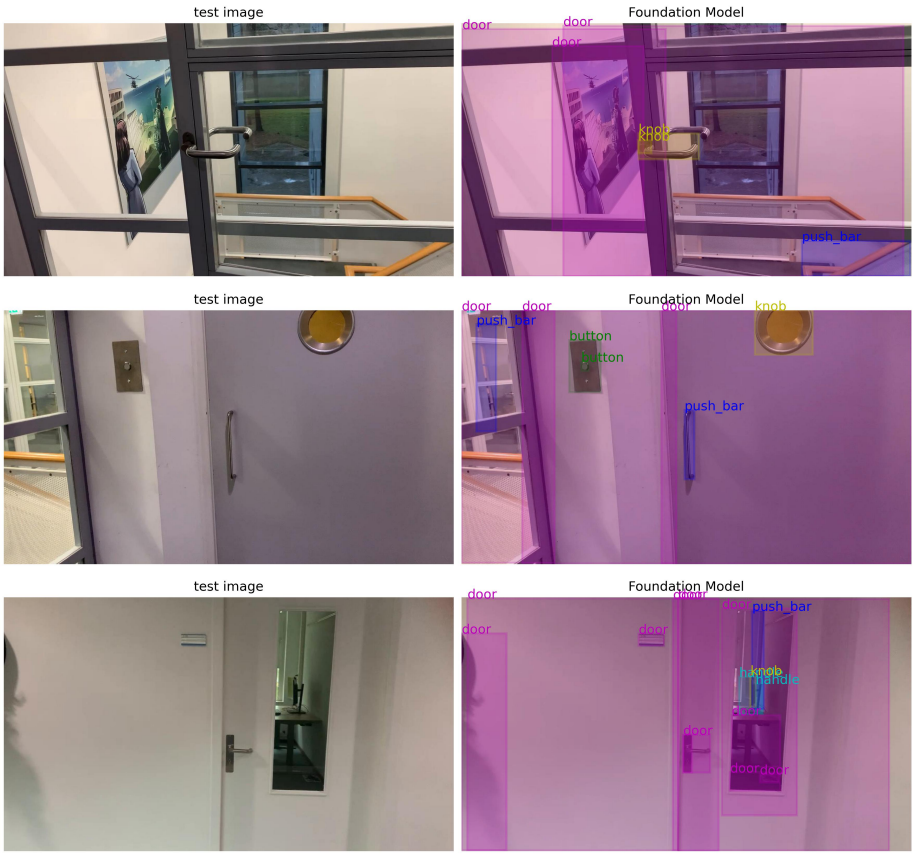

Fig. 4: Standard GLIP is incapable of fine-grained discrimination of various door openers.

3.4 Finegrained Discrimination by Sparse Human Feedback

When using an open-vocabulary detection model, our objective is to increase the model's discrimination between fine-grained objects. Given that most of the relevant objects can be detected (see Figure 4), the recall is promising already. The problem is mostly in the classification of the objects, i.e. the labels assigned to them (e.g. a handle that is labeled as a knob). To improve the label assignment, our proposal is to involve a human in the process, to reassign labels in an efficient manner. In our scenario, the robot is tasked to navigate through an unknown building. During a first run, it is able to find the relevant objects (recall), but with wrong labels assigned to them. Detected objects are visualized and grouped on a 2D canvas to create an overview, where the human can quickly identify the classes of objects and label a few characteristic instances of each class. In our approach, each object is represented by a feature vector by a visual encoder, we have used CLIP [29]. The objects can now be distributed on the 2D canvas by a dimensionality reduction, we have used t-SNE [23]. This canvas enables the human to quickly assign labels to the encountered object classes.

Being provided with a few labels, a task-specific model can now be deployed for the task and environment at hand. For simplicity we leverage a nearest neighbor model, on top of the VLM, to relabel the outputs of the VLM. More specifically, only the labels and confidences are changed by this second model; the boxes remain unchanged. The relabelling model is constructed as follows. The N labels are defined as $D = \{(x_i, y_i)\}_{i \in 1..N}$, where x_i denotes the feature vector of object i and y_i is the provided label for object i. An object seen during testing is encoded by feature vector x_j^{test}. It gets a label assigned, y_j^{test}, with a confidence value c_j^{test}. The confidence value c is important for the robot's planning, because it can go to the most confident object first. The assignment of label y and confidence c is inferred the minimal cosine distance to labeled objects:

$$c(x_j^{test}, x_i) = \frac{x_j^{test} \cdot x_i}{\|x_j^{test}\| \cdot \|x_i\|} \tag{2}$$

$$y_j^{test} = \arg\max_{y_i} c(x_j^{test}, x_i), \;\; (x_i, y_i) \in D \tag{3}$$

3.5 Refining Objects by Spatial Reasoning

As can be seen from Figure 4, there are many false positives when using GLIP. The precision of affordance objects can be enhanced by verifying known spatial relations. For instance, the opener of a door is typically nearby the door. We follow the approach from [7]. A neuro-symbolic program [17] takes the often uncertain objects and their labels and confidences $\{(y_j^{test}, c_j^{test})\}_{j \in 1..M}$ and verifies spatial relations that are predefined by the user before the robot starts the mission. The neuro-symbolic program takes the spatial relations in the form of first-order logic:

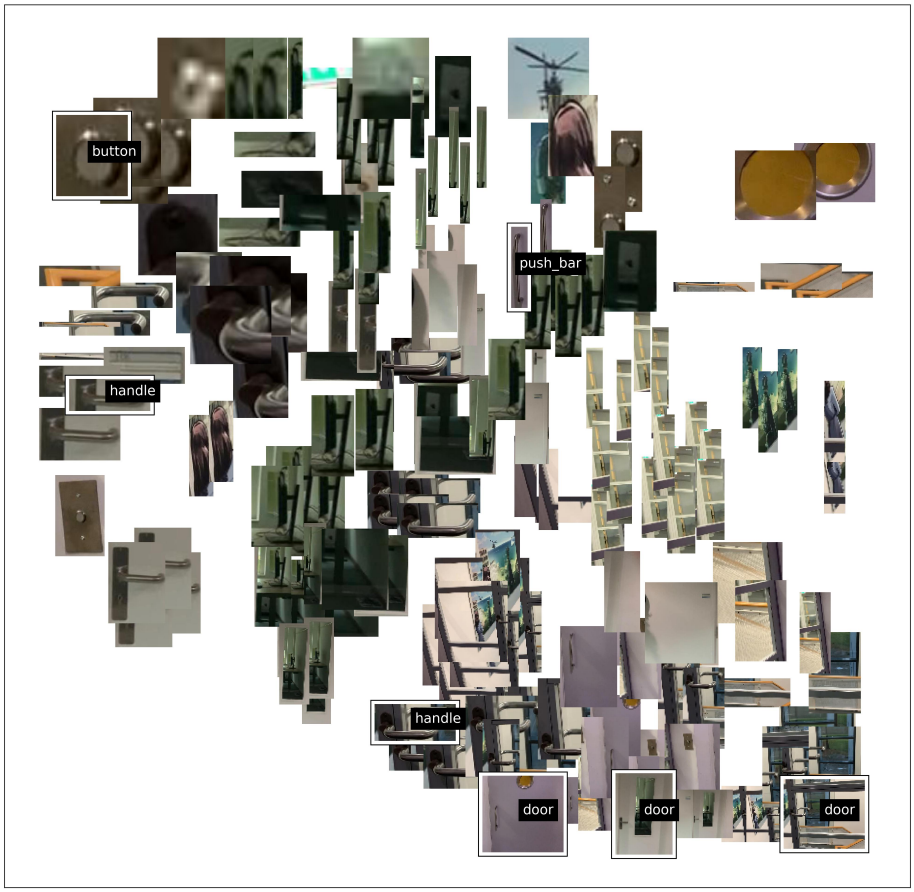

Fig. 5: In the overview of all detected objects, the main classes can be identified quickly, as shown by the new labels that were assigned to them by a user.

$$\begin{aligned} \exists\, d, o : \; & object(d, door) \wedge object(o, opener) \wedge \\ & correct\text{-}range\text{-}hor(d, o) \wedge \\ & correct\text{-}range\text{-}vert(d, o) \end{aligned} \tag{4}$$

This specifies that a door opener o is expected to be close to the horizontal side of the door d and vertically close to the middle of the door d:

$$correct\text{-}range\text{-}hor(d, o) = max(1 - \frac{hor\text{-}dist\text{-}from\text{-}side(o, d)}{width(d)}, 0) \tag{5}$$

$$correct\text{-}range\text{-}vert(d,o) = max(1 - \frac{vert\text{-}dist\text{-}from\text{-}middle(o,d)}{height(d)}, 0) \quad (6)$$

The spatial reasoning approach can be a solution for classes of object affordances if the specification expressed in first-order logic can be a fairly accurate representation of real world conditions. In case there is a higher degree of uncertainty in spatial relations, the approach may be less effective and other formalisms (e.g. possibilistic logic) may yield better results.

4 Experiment

4.1 Setup and Dataset

For experimentation, we consider the scenario where a robot navigates through an office building, and the robot needs to localize the openers such that it can open doors to move through them. In order to test the validity of our approach, we have conducted a limited experiment in a single environment. Videos of three different scenes with various doors and openers were collected for training, comprising 1553 frames. Only 3 frames were labeled by a human user. These frames result from the human feedback as shown in Figure 5. This resulted in 3 labeled doors, 2 labeled door handles, 1 labeled push bar and 1 labeled button. For testing, another video was recorded with similar doors and openers, but under different circumstances such as changed viewpoint and camera distance. This yielded 1370 test frames.

4.2 Visualization

We already established that GLIP's object predictions are not actionable due to many wrong labels and false positives (see Figure 4). With our relabeling method from Section 3.4, the predictions improve significantly. Figure 6 shows the improved results. In the top row, it shows that the handle was initially labeled as a knob, but it is corrected, and now labeled as a handle. This enables a robot to select the correct action (push down) instead of the wrong action (rotate). In the bottom row, a similar improvement is observed, for relabeling a knob (error) to a handle (correct). In the middle row, the button was corrected from knob to button, and the false positive (push bar) is removed. Most improvement is due to the relabeling method, whereas the spatial reasoning adds minor improvements. For instance, in the middle row, a false positive (handle) is removed, because its spatial relation to the door is not conforming to the expectation (Equation 4). A demo clip of our method's predictions on the full test recordings can be found online.

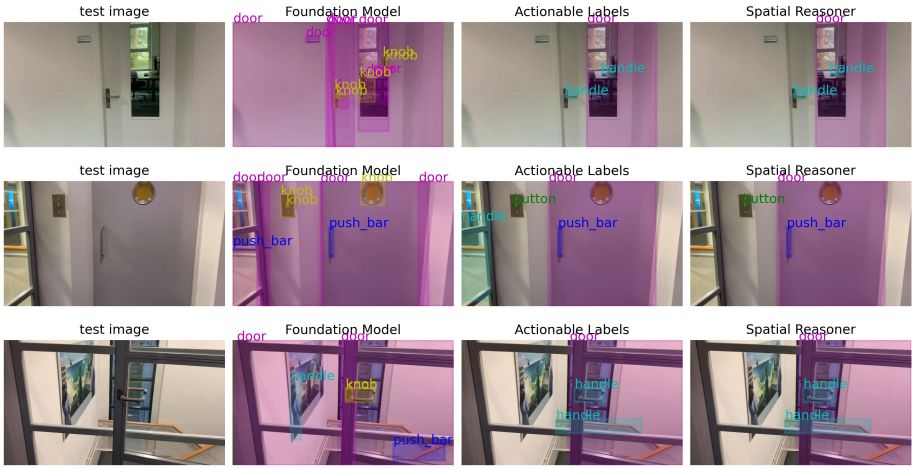

Fig. 6: Our relabeling method yields actionable labels.

4.3 Performance Evaluation

A quantitative analysis shows that the performance is indeed improved significantly. The metric for object detection is mAP which measures both the localization and classification accuracy. For the localization, a predicted object should have an overlap with the ground truth of at least IoU ≥ 0.5, where IoU is the intersection over union. From the 1370 test frames, 11 images were labeled for validation. The predictions of GLIP are not actionable, as can be observed in Figure 7 from the low scores for the green bars (left bar in each group), for the door openers, i.e. the handle (almost zero), push bar and button. All scores are below 0.15, which means that less than 1 out of 7 predicted objects is correct. For application in robotics, this is insufficient because the robot would make mostly mistakes while it operates. The performance of our relabeling method is indicated in blue (middle bar in each group). The scores are much higher, showing its effectiveness for finding objects that are related to the affordances of interest. The spatial reasoner is effective for the push bar (orange bar). For this class, GLIP produced many false positives in the background, as there are many elongated shapes that have some resemblance with a push bar. These false positives do not conform to expected spatial relations, i.e. being close to the door. For the other door openers, the spatial reasoning does not impact the predictions much.

5 Conclusion

We have investigated the possibility of affordance analysis by a robot without closed-world assumptions: its training and world model does not include complete world knowledge. Affordances, objects and actions can be precisely modeled

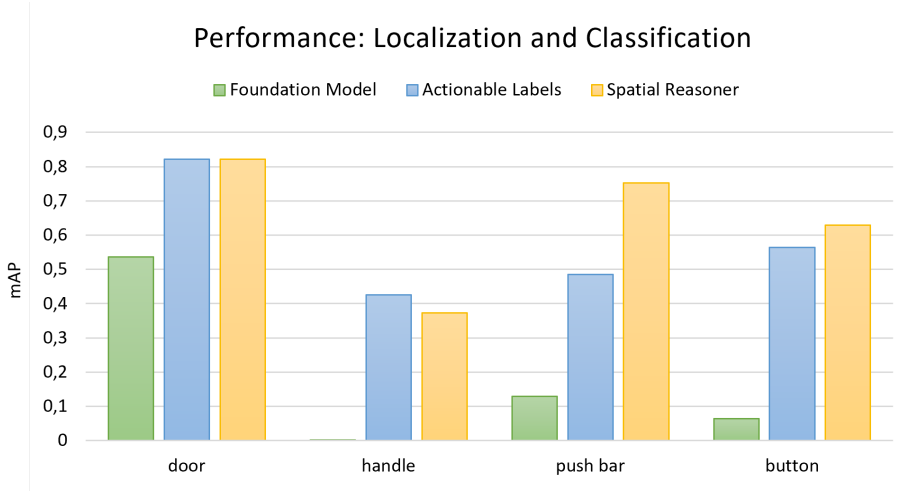

Fig. 7: For the localization and classification of affordance objects, the performance of GLIP (green bars) is increased significantly by our method (blue and orange bars).

in a relational model. A future option could be to automatically extract those from a knowledge base or LLM. Objects from these relations can be visually localized with a VLM. Label assignment quality varies but is overall semantically imprecise (false positives include mix-ups of handrails and push bars) and not actionable (e.g. knob is identified as a handle). Furthermore, detection provides a good starting point for determination of relevant objects in terms of affordances: the recall is good but precision needs improvement. An effective way to determine actionable labels and improve the model, is involvement of sparse feedback from a human who labels characteristic objects from a 2D plot. This drastically improves the mAP (combination of precision and recall for localization and labels) with only a few labels. Our current, limited experiment is a first step towards a more elaborate investigation of our approach in a diverse set of environments and a more open-world setting.

References

1. Ahn, M., Brohan, A., Brown, N., Chebotar, Y., Cortes, O., David, B., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Ho, D., Hsu, J., Ibarz, J., Ichter, B., Irpan, A., Jang, E., Ruano, R., Jeffrey, K., Jesmonth, S., Yan, M.: Do as i can, not as i say: Grounding language in robotic affordances. Conference on Robotic Learning (04 2022)
2. Ardon, P., Pairet, E., Lohan, K.S., Ramamoorthy, S., Petrick, R.: Affordances in robotic tasks – a survey. arXiv:2004.07400 [cs] (2022)

3. Ardon, P., Pairet, E., Lohan, K.S., Ramamoorthy, S., Petrick, R.P.: Building affordance relations for robotic agents - a review. Proceedings of the Thirtieth Joint Conference on Artificial Intelligence IJCAI-21 - Survey track (2021)
4. Beßler, D., Porzel, R., Pomarlan, M., Beetz, M., Malaka, R., Bateman, J.: A formal model of affordances for flexible robotic task execution. Frontiers in Artificial Intelligence and Applications **325**, 2425–2432 (2020). https://doi.org/10.3233/FAIA200374
5. Bugliarello, E., Sartran, L., Agrawal, A., Hendricks, L.A., Nematzadeh, A.: Measuring progress in fine-grained vision-and-language understanding (2023)
6. Burghouts, G.J., Hillerström, F., Walraven, E., van Bekkum, M., Ruis, F., Sijs, J.: Anomaly detection in an open world by a neuro-symbolic program on zero-shot symbols. In: IROS 2022 Workshop Probabilistic Robotics in the Age of Deep Learning (2022)
7. Burghouts, G.J., Meijer, W., Hillerström, F., van Mil, J., van Bekkum, M., Schaaphok, M., Ruis, F.: Improved zero-shot object localization using contextualized prompts and objects in context. In: ICRA2023 Workshop on Pretraining for Robotics (PT4R) (2023)
8. Chen, D., Kong, D., Li, J., Wang, S., Yin, B.: A survey of visual affordance recognition based on deep learning. IEEE Transactions on Big Data **9**(6), 1458–1476 (2023). https://doi.org/10.1109/TBDATA.2023.3291558
9. Francisco Cruz, Sven Magg, C.W., Wermter., S.: Training agents with interactive reinforcement learning and contextual affordances. IEEE Transactions on Cognitive and Developmental Systems **8**, 271–284 (2016)
10. Gaver, W.W.: Technology affordances. Conference on Human Factors in Computing Systems - Proceedings pp. 79–84 (1991). https://doi.org/10.1145/108844.108856
11. Gibson, J.J.: The Ecological Approach to Visual Perception. Psychology Press Classic Editions (1979)
12. Graves, D., Günther, J., Luo, J.: Affordance as general value function: a computational model. Adaptive Behavior **4**, 307–327 (2022)
13. Gu, X., Lin, T.Y., Kuo, W., Cui, Y.: Open-vocabulary detection via vision and language knowledge distillation. arXiv preprint arXiv:2104.13921 (2021)
14. Hart, S., Quispe, A.H., Lanighan, M.W., Gee, S.: Generalized affordance templates for mobile manipulation. 2022 International Conference on Robotics and Automation (ICRA) pp. 6240–6246 (2022)
15. Hart, S., Dinh, P., Hambuchen, K.: The affordance template ros package for robot task programming. In: 2015 IEEE International Conference on Robotics and Automation (ICRA). pp. 6227–6234 (2015). https://doi.org/10.1109/ICRA.2015.7140073
16. Ho, S.B.: A general framework for the representation of function and affordance: A cognitive, causal, and grounded approach, and a step toward agi (2022)
17. Huang, J., Li, Z., Chen, B., Samel, K., Naik, M., Song, L., Si, X.: Scallop: From probabilistic deductive databases to scalable differentiable reasoning. In: Advances in Neural Information Processing Systems. vol. 34, pp. 25134–25145 (2021)
18. Kommineni, V.K., König-Ries, B., Samuel, S.: From human experts to machines: An llm supported approach to ontology and knowledge graph construction (2024)
19. Li*, L.H., Zhang*, P., Zhang*, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J.N., Chang, K.W., Gao, J.: Grounded language-image pre-training. In: CVPR (2022)

20. Li, P., Tian, B., Shi, Y., Chen, X., Zhao, H., Zhou, G., Zhang, Y.Q.: Toist: Task oriented instance segmentation transformer with noun-pronoun distillation. Advances in Neural Information Processing Systems **35**, 17597–17611 (2022)
21. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. pp. 740–755. Springer (2014)
22. Luis Montesano, Manuel Lopes, A.B., Santos-Victor., J.: Modeling affordances using bayesian networks. IEEE/RSJ International Conference on Intelligent Robots and Systems pp. 4102–4107 (2007)
23. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(11) (2008)
24. Min, H., Yi, C., Luo, R., Zhu, J., Bi, S.: Affordance research in developmental robotics: A survey. IEEE Transactions on Cognitive and Developmental Systems **8**(4), 237–255 (2016). https://doi.org/10.1109/TCDS.2016.2614992
25. Minderer, M., Gritsenko, A., Stone, A., Neumann, M., Weissenborn, D., Dosovitskiy, A., Mahendran, A., Arnab, A., Dehghani, M., Shen, Z., Wang, X., Zhai, X., Kipf, T., Houlsby, N.: Simple open-vocabulary object detection with vision transformers. arXiv preprint arXiv:2205.06230 (2022)
26. Norman, D.A.: The Design of Everyday Things. Basic Books, New York, 2 edn. (2013)
27. OpenAI: ChatGPT (2023), accessed: 2023-04-21
28. OpenAI: Gpt-4 technical report (2023)
29. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: ICML (2021)
30. Sawatzky, J., Souri, Y., Grund, C., Gall, J.: What object should i use?-task driven object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7605–7614 (2019)
31. Speer, R., Havasi, C.: Representing general relational knowledge in concept net 5. Proceedings of the 8th International Conference on Language Resources and Evaluation, LREC 2012 pp. 3679–3686 (2012)
32. Tang, J., Zheng, G., Yu, J., Yang, S.: Cotdet: Affordance knowledge prompting for task driven object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3068–3078 (2023)
33. Vaticle: The polymorphic database powered by types (2023), `https://typedb.com/`
34. Vo, D.M., Chen, H., Sugimoto, A., Nakayama, H.: Noc-rek: novel object captioning with retrieved vocabulary from external knowledge. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 17979–17987. IEEE (2022)
35. Yamanobe, N., Wan, W., Ramirez-Alpizar, I.G., Petit, D., Tsuji, T., Akizuki, S., Hashimoto, M., Nagata, K., Harada, K.: A brief review of affordance in robotic manipulation research. Advanced Robotics **31**(19-20), 1086–1101 (2017). https://doi.org/10.1080/01691864.2017.1394912
36. Zech, P., Haller, S., Lakani, S.R., Ridge, B., Ugur, E., Piater, J.: Computational models of affordance in robotics: a taxonomy and systematic classification. Adaptive Behavior **25**(5), 235–271 (2017). https://doi.org/10.1177/1059712317726357, `https://doi.org/10.1177/1059712317726357`

37. Zhang, H., Zhang, P., Hu, X., Chen, Y.C., Li, L.H., Dai, X., Wang, L., Yuan, L., Hwang, J.N., Gao, J.: Glipv2: Unifying localization and vision-language understanding (2022)
38. Şahin, E., Çakmak, M., Doğar, M.R., Uğur, E., Üçoluk, G.: To afford or not to afford: A new formalization of affordances toward affordance-based robot control. Adaptive Behavior (2007). https://doi.org/10.1177/1059712307084689