Appendix

Anonymous Author(s)
Affiliation
Address
email

1 Method Details

1.1 Spatial Relation Heuristics

We implement a total of 8 object spatial relation heuristics given the aggregated semantic point cloud and 3D object bounding boxes. We consider two in-contact relations, when the minimum distance between the object point clouds are smaller than 5cm:

1. **Inside.** Object A is considered inside object B if over 50% percent of object A's point cloud is inside the convex hull of object B.
2. **On top of.** Object A is considered on top of object B if it satisfies the following three conditions: a) over 70% of object A's point cloud lies within a XY-plane projected from object B's 3D bounding box. b) over 70% of object A's points are above the upper Z bound of object B's bounding box.

If the object point cloud distance is larger than 5cm but smaller than 40cm, we subtract the center points of object A and B's 3D bounding box, transform it back to camera coordinates (Y-up) and normalize to get a 3 dimensional unit vector. The following relations are considered:

1. **Above & Below.** If the y-component of the vector is over 0.9, then object A is above object B. If the y-component is smaller than -0.9, then object A is below object B.
2. **On the left & On the right.** If the x-component of the vector is over 0.8, then object A is on the right of object B, if the x-component is smaller than -0.8, then object A is on the left of object B.
3. **Occluding.** If 90% of object A's point has smaller depth than object B's minimum depth and object A's 2D bounding box (projected in image space from its aggregated 3D point cloud) overlaps more than 25% with that of object B's projected bounding box, then object B is considered occluding object A.
4. **Near.** If none of the above relations satisfy but the object distance is smaller than 10cm, then object A is near object B.

We acknowledge that the spatial heuristics method requires some tuning on the thresholds, yet we find it to be an efficient and reliable method to obtain object spatial relations on both our simulation and real world data, and one set of parameters is sufficient for most scenarios. The biggest advantage of the method is zero-shot generalization to novel scenes, comparing to learning-based methods that suffer from domain shift. The point clouds are also more robust against occlusion by taking advantage of the aggregated point cloud across frames.

1.2 Scene Graph Aggregation Heuristics

We aggregate the 3D point cloud over time frames with a similar approach as Li et al. [1]. Consider the point cloud observed in the current time step p_t and the accumulated point cloud from all previous time steps P_{t-1}, we can obtain the accumulated point cloud P_t up to time step t with 4 operations: ADD, UPDATE, REPLACE, and DELETE.

1. **ADD.** If an object is observed in p_t but not in P_{t-1}, we consider it a newly appeared object and add it as a new node. If the node is task-relevant, then its relations with all existing objects will

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

be computed and added as edges. If the node is not task-relevant, then its relations with existing task-relevant objects will be computed and added as edges.

2. **UPDATE.** If an object is in both p_t and P_{t-1}, and the object point cloud is aligned, we update the object point cloud by concatenating the newly observed points and updating its existing edges with other objects by recomputing the spatial relations.

3. **REPLACE.** If an object is in both p_t and P_{t-1}, but the misalignment of the object point clouds is larger than a threshold d, indicating the object has moved. We remove the old node in P_{t-1} and add the object as a new node.

4. **DELETE.** If an object is in P_{t-1} but not in p_t, and the robot is interacting with the object, then we remove the object from the accumulated point cloud since its location becomes unknown.

1.3 Example Prompts

1.3.1 Subgoal Verification

system prompt:

> You are a success verifier that outputs 'Yes' or 'No' to indicate whether the robot goal is satisfied given the robot observations.

user prompt:

> The robot goal is to [SUBGOAL]. Here are the robot observations after execution: [OBSERVATION]
> Q: Is the goal satisfied?
> A: Yes

The [SUBGOAL] and [OBSERVATION] entry will be filled in when prompting for each subgoal. Here's a complete example of a failed execution during the task "boil water". The robot accidentally dropped the pot when navigating to the stove burner and then attempted to put the pot on the stove burner:

> The robot goal is to pick up pot. Here are the robot observations after execution:
> Visual observation: pot (empty and clean). pot is inside robot gripper.
> Q: Is the goal satisfied?
> A: Yes

> The robot goal is to put pot in sink. Here are the robot observations after execution:
> Visual observation: faucet (turned off), pot (empty and clean), sink. pot (empty and clean) is inside sink. pot (empty and clean) is on the right of soap bottle. nothing is inside robot gripper.
> Q: Is the goal satisfied?
> A: Yes

> The robot goal is to toggle on faucet. Here are the robot observations after execution:
> Visual observation: pot (filled with water and clean), faucet (turned on), sink. pot (filled with water and clean) is inside sink. pot (filled with water and clean) is on the right of soap bottle. nothing is inside robot gripper. Auditory observation: water runs in sink.
> Q: Is the goal satisfied?
> A: Yes

> The robot goal is to toggle off faucet. Here are the robot observations after execution:
> Visual observation: pot (filled with water and clean), faucet (turned off), sink. pot (filled with water and clean) is inside sink. pot (filled with water and clean) is on the right of soap bottle. nothing is inside robot gripper.
> Q: Is the goal satisfied?
> A: Yes

> The robot goal is to pick up pot. Here are the robot observations after execution:
> Visual observation: pot (filled with water and clean), faucet (turned off), sink. pot is inside robot gripper.
> Q: Is the goal satisfied?
> A: Yes

The robot goal is to put pot on fourth stove burner. Here are the robot observations after execution:
Visual observation: second stove burner (turned off), first stove burner (turned off), third stove burner (turned off), fourth stove burner (turned off). nothing is inside robot gripper.
Q: Is the goal satisfied?
A: No

Here a subgoal is not satisfied, the program enters the **execution analysis** mode: history observations stored in the event-based summary are retrieved to query LLM for failure explanation.

1.3.2 Failure explanation: execution analysis

system prompt:

You are expected to provide explanation for a robot failure. You are given the robot actions and observations so far. Briefly explain the failure in 1-2 sentence. Mention relevant time steps if possible.

user prompt:

The robot task is to boil water. At 00:44, a failure was identified.

[Robot actions and observations before 00:44]
00:01. Action: Move to pot. Visual observation: nothing is inside robot gripper.
00:11. Action: Move to pot. Visual observation: faucet (turned off). nothing is inside robot gripper.
00:15. Action: Move to pot. Visual observation: pot (empty and clean). pot (empty and clean) is on the left of potato. pot (empty and clean) is on top of third countertop. nothing is inside robot gripper.
00:18. Action: Pick up pot. Visual observation: pot (empty and clean). pot is inside robot gripper.
00:21. Action: Move to sink. Visual observation: pot (empty and clean), faucet (turned off), sink. pot is inside robot gripper.
00:22. Action: Move to sink. Visual observation: pot (empty and clean), faucet (turned off), sink. pot is inside robot gripper.
00:25. Action: Put pot in sink. Visual observation: pot (empty and clean), faucet (turned off), sink. pot (empty and clean) is inside sink. pot (empty and clean) is on the right of soap bottle. nothing is inside robot gripper.
00:28. Action: Toggle on faucet. Visual observation: pot (filled with water and clean), faucet (turned on), sink. pot (filled with water and clean) is inside sink. pot (filled with water and clean) is on the right of soap bottle. nothing is inside robot gripper. Auditory observation: water runs in sink.
00:31. Action: Toggle off faucet. Visual observation: pot (filled with water and clean), faucet (turned off), sink. pot (filled with water and clean) is inside sink. pot (filled with water and clean) is on the right of soap bottle. nothing is inside robot gripper.
00:34. Action: Pick up pot. Visual observation: pot (filled with water and clean), faucet (turned off), sink. pot is inside robot gripper.
00:36. Action: Move to fourth stove burner. Visual observation: pot (empty and clean), faucet (turned off), sink. pot (empty and clean) is on the right of potato. pot (empty and clean) is on top of third countertop. nothing is inside robot gripper. Auditory observation: something drops.
00:38. Action: Move to fourth stove burner. Visual observation: pot (empty and clean), faucet (turned off), sink. nothing is inside robot gripper.
00:42. Action: Move to fourth stove burner. Visual observation: second stove burner (turned off), fourth stove burner (turned off), third stove burner (turned off), first stove burner (turned off). nothing is inside robot gripper.
00:43. Action: Move to fourth stove burner. Visual observation: second stove burner (turned off), fourth stove burner (turned off), third stove burner (turned off), first stove burner (turned off). nothing is inside robot gripper.

[Observation at the end of 00:44]
Action: Put pot on fourth stove burner. Visual observation: second stove burner (turned off), fourth stove burner (turned off), third stove burner (turned off), first stove burner (turned off). nothing is inside robot gripper.

Q: Infer from [Robot actions and observations before 00:44] or [Observation at the end of 00:44], briefly explain what happened at 00:44 and what caused the failure.
A: At 00:44, the robot attempted to put the pot on the fourth stove burner, but the pot was not in its gripper. The failure was caused by the robot dropping the pot filled with water at 00:36 while moving to the fourth stove burner.

The failure steps 00:36 and 00:44 can be extracted from the answer by prompting LLM to extract time steps from the output failure explanation.

1.3.3 Failure explanation: planning analysis

In case all subgoals are satisfied, then there's likely mistakes in the robot original plan. The program will enter **planning analysis** mode. Take the failure scenario when the robot plan is wrong during the task "boil water" so that the robot placed the pot on a stove burner but toggled on another:

system prompt:

> You are expected to provide explanation for a robot failure. You are given the current robot state, the goal condition, and the robot plan. Briefly explain what was wrong with the robot plan in 1-2 sentence.

user prompt:

> The robot task is to boil water. The task is considered successful if a pot is filled with water, the pot is on top of a stove burner that is turned on.
> Here's the robot observation at the end of the task execution:
> faucet (turned off), second stove burner (turned on), sink, pot (filled with water and clean), fourth stove burner (turned off), third stove burner (turned off), first stove burner (turned off). pot (filled with water and clean) is on top of fourth stove burner (turned off). nothing is inside robot gripper.
> The robot plan is:
> 00:18. Goal: Pick up pot.
> 00:25. Goal: Put pot in sink.
> 00:28. Goal: Toggle on faucet.
> 00:31. Goal: Toggle off faucet.
> 00:34. Goal: Pick up pot.
> 00:46. Goal: Put pot on stove burner.
> 00:49. Goal: Toggle on stove burner.
>
> Q: Known that all actions in the robot plan were executed successfully, what's wrong with the robot plan that caused the robot to fail?
> A: The robot placed the pot on the fourth stove burner but turned on the second stove burner, causing a mismatch between the pot's location and the active burner.

The failure time step can be obtained by a follow-up query to the LLM with the prompt below:

> Q: Which time step is most relevant to the above failure?
> A: 00:49

1.3.4 Correction

Still take the failure scenario when the robot plan is wrong so that the robot placed the pot on a stove burner but toggled on another. A complete prompt for generating the failure correction plan is as follows:

system prompt:

> Provide a plan with the available actions for the robot to correct its failure and finish the task.
> Available actions: pick up, put in some container, put on some receptacle, open (e.g. fridge), close, toggle on (e.g. faucet), toggle off, slice object, crack object (e.g. egg), pour (liquid) from A to B. The robot can only hold one object in its gripper, in other words, if there's object in the robot gripper, it can no longer pick up another object.
> The plan should 1) not contain any if statements 2) contain only the available actions 3) resemble the format of the initial plan.

user prompt:

Task: boil water
Initial plan:
1. pick_up (pot)
2. put_in (pot, sink)
3. toggle_on (faucet)
4. toggle_off (faucet)
5. pick_up (pot)
6. put_on (pot, stove burner)
7. toggle_on (stove burner)
Failure reason: The robot placed the pot on the fourth stove burner but turned on the second stove burner, causing a mismatch between the pot's location and the active burner.
Current state: sink, pot (filled with water and clean), fourth stove burner (turned off), third stove burner (turned off), faucet (turned off), first stove burner (turned off), second stove burner (turned on). pot (filled with water and clean) is on top of fourth stove burner (turned off). nothing is inside robot gripper.
Success state: a pot is filled with water, the pot is on top of a stove burner that is turned on.
Correction plan: toggle_off (stoveburner-2), toggle_on (stoveburner-4)

2 Evaluation Details

2.1 Dataset Details

Task descriptions for the 10 simulation tasks and 12 real world tasks in the RoboFail dataset are shown below.

Task	Task Description / Goal State
boil water	A pot is filled with water, the pot is on top of a stove burner that is turned on
toast bread	A bread slice is inside a toaster that is turned on
fry egg	A cracked egg is in a pan, the pan is on top a stove burner that is turned on
heat potato	A potato is on a plate and inside a microwave that is turned on
serve coffee	A clean mug is filled with coffee and on top of the countertop
store egg	A bowl with an egg is stored inside the fridge
make salad	A bowl of sliced lettuce, tomato and potato is stored inside the fridge
water plant	The house plant is filled with water
switch devices	Laptop is closed on the TV stand and the television is turned on
serve warm water	A mug of water is heated in the microwave and served on the dining table

Table 1: Simulation Tasks

Task	Task Description / Goal State
boil water	A pot is filled with water, the pot is on top of a stove burner that is turned on
sauté carrot	A sliced carrot is inside a pan, the pan is on top of a stove burner that is turned on
heat potato	A potato is heated in the microwave and then put on the countertop
serve coffee	A mug is filled with coffee and on top of the countertop
store egg	A bowl with an egg is stored inside the fridge
secure objects	Knife is stored in a drawer and pear is stored in the fridge
apple in bowl	Apple is inside bowl
pear in drawer	Pear is inside a closed drawer
cut carrot	Carrot is sliced
fruits in bowl	All visible fruits are inside a bowl
heat pot	A pot is on top of a stove burner that is turned on

Table 2: Real World Tasks

2.2 Human Evaluation

Similar to the approach for human evaluation in Ahn et al. [2], we ask 2 groups of users, 3 in each group to compare the ground truth failure explanation labelled in the dataset and REFLECT-generated failure explanation for each failure scenario. The failure scenarios are randomly shuffled in the questionnaires sent to the users. The users are instructed to score 0 if the predicted explanation is incorrect, 1 if the predicted explanation is correct, and 2 if they are unsure. The final score reflected in the tables are the majority vote without counting “unsure”. If there’s a tie in the answers or more than one “unsure” is given, we will ask the users to re-score the specific case.

References

[1] X. Li, D. Guo, H. Liu, and F. Sun. Embodied semantic scene graph generation. In A. Faust, D. Hsu, and G. Neumann, editors, *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pages 1585–1594. PMLR, 08–11 Nov 2022.

[2] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, C. Fu, K. Gopalakrishnan, K. Hausman, A. Herzog, D. Ho, J. Hsu, J. Ibarz, B. Ichter, A. Irpan, E. Jang, R. J. Ruano, K. Jeffrey, S. Jesmonth, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, K.-H. Lee, S. Levine, Y. Lu, L. Luu, C. Parada, P. Pastor, J. Quiambao, K. Rao, J. Rettinghouse, D. Reyes, P. Sermanet, N. Sievers, C. Tan, A. Toshev, V. Vanhoucke, F. Xia, T. Xiao, P. Xu, S. Xu, M. Yan, and A. Zeng. Do as i can and not as i say: Grounding language in robotic affordances. In *arXiv preprint arXiv:2204.01691*, 2022.