Appendix

A. Prompts for LLM modules.

Imagination Analyzer:

In an experiment to check whether a novel object possesses the affordance of a certain class, given the object specification, you are a helper to generate abstract analysis for the verification pipeline.

For example, given the task:

Check whether the object can be a(n) .... First you need to make a preliminary judgement whether the object meets the normal dimension of the class, and verifiable affordance here only refers how it can passively interact with other objects at static depending on its geometry shape, including no state switches or complicated functions.

Please strictly follow the template:

<Start of analysis>

<Class of object>:<the class requested by user>

<Preliminary judgment>: <Yes or No>

<Judgment analysis>: <The reason for yes or no>

<Requested affordance>:<the main affordance of the class>

<Verifiability of affordance>: <Yes or No>

If it is ..., it should afford.... Then we can use ... to test its affordance.

<Agent choice>: <the agent object to interact with the testing object, should be representative and in simple shape like cube, sphere, cylinder and so on>

<Agent number>: <Single or Multiple>

<Agent size>: [x, y, z]

<Agents arrangement>: Arranged in the <you need to specify in which plane the agents are arranged, e.g. in x-y, x-z or y-z plane, should be NA if it is single agent case>

<Affordance verification>: you can move the agent(s) from ... to ... (and then ...), if ..., the affordance can be verified.

<end of analysis>

Remember:

1. When choosing a single or multiple agents, it should comply to commonsense of use case.

2. In multi-agent case, consider using identical agents with possibly small size in case there are defects on the object affordance, which cannot be detected by a single large agent.

3. In case of multiple agents, you need to consider the common contact plane of interaction with the object in which the agents should be arranged.

4. Subsequent output is applicable only when Preliminary judgment and the Verifiability of affordance are Yes.

5. All dimensions are in meters.

Agent Generator:

In an experiment to check whether a novel object possesses the affordance of a certain class, given the analysis of the affordance imagination, you are a helper to generate the model of the agent used for interaction in a .urdf file.

You have to follow the answer template:

<output>:

```xml

Remember:

1. You do not need to add explanation following the output file.

2. For efficiency, you should use built-in basic geometry.

3. You should include necessary information about the agent's properties in the output urdf file.

## Agent Configurator:

In an experiment to check whether a novel object possesses the affordance of a certain class, given the abstract agent configuration, you are a helper to determine the numerical values of the configuration, which includes how the agents are distributed geometrically. And for this configuration, there is a center of the distribution of agents.

Remember:

1. You need to specify translations along different axis of each agent relative to the distribution center.

2. The gap should be larger than the maximum dimension of the agent to avoid collision.

3. No comments of explanation or operation process are allowed in the output.\n4. Agents should be symmetrically distributed around the center of distribution, if arranged in a grid pattern or only a single agent is used, the relative displacement in z should be 0.

You have to follow the answer template:

<Start of configuration>

<Number of agents>:<an int value from 1 to 25, consider using more agents to reduce chance case>

<Shape of configuration>:<a common geometry matching the Agent configuration in the analysis>

<Array of agents>:<a matrix as x*y*z, ,which means the number of agents distributed in different axis, and 1*1*1 means only one agent used>

<Space between centers of adjacent agents>:<a float value>

<End of configuration>

<data output>:

{"agent_1": [relative position to the center],

"agent_2": [relative position to the center] ...}"
```

Interaction Plan Generator:

In an experiment to check whether a novel object possesses the affordance of a certain class, given the generated analysis, you are a helper to explore the paradigm of using the affordance of the object by interacting with the agent(s). In simulation, each agent is controlled to follow some key poses and then released.

Answer template:

Plan 1:

1. Start the agent from ... 2. Move the agent to ... 3. Stop the agent when ...

Remember:

1. According to the affordance verification, you need to generate 5 different candidate plans, each one means a trajectory of the agent.

2. The object is at static, so there should be no step moving the object.

3. For each plan, you do not need to specify the numeric value of the trajectory.

4. State the plan step by step.

5. You should keep a safe distance at the beginning to avoid collision between the agent and object.

6. You do not need to include the step of releasing.

7. You can consider planning from five directions: above, front, back, left and right."

Trajectory Generator:

In an experiment to check whether a novel object possesses the affordance of a certain class, you are a helper to transfer the abstract plans into numerical trajectories of the agent. I will provide the affordance imagination analysis and the plans of interaction between the agent and the object.

Remember:

1. For each plan, you need to generate a reasonable trajectory and output two key poses of the agent along the trajectory, which are written as 3*1 vectors.

2. Each vector is the translation of the agent in x, y, and z axes respectively.

3. Consider the bounding box of the agents and object to avoid collision along the trajectory.

4. Arithmetic symbols are not allowed in the output.

5. Output the data in a json file and do not include text explanation.

6. The final pose of the agent should have no displacement in x and y, which is 0. And the final height z should not be too high.

7. There should be an enough distance between the agent and the object at the beginning to avoid collision.

8. Follow the direction described in the plans as much as possible without deviation.

You have to follow the answer template:

{"plan_1": [[x1, y1, z1], [x2, y2, z2], ...],

"plan_2\": [[x1, y1, z1], [x2, y2, z2], ...] ...}"

Evaluation Function Generator:

In an experiment to check whether a novel object possesses the affordance of a certain class, given the analysis on the affordance imagination and the simulation results, you are a helper to define a discriminant function called "is_successful" in Python to check if each trial is successful.

In the simulation result, there are six values returned:

(1) agent_pos_list (list of 3x1 list): list of ending poses of each agent in this trial.

(2) agent_rotm_list (list of 3x3 list): list of ending orientation of each agent in this trial.

(3) agent_obj_contact_number_list (list of int): list of numbers of contact points between each agent with the object in this trial, 0 means no contact, otherwise there is contact.

(4) agent_ground_contact_list (list of bool): whether each agent contacts with the ground in this trial, true means there is contact and false vice versa.

(5) obj_center (a 3x1 array):position of the center of object.

(6) obj_bbox(a 3x1 array): size of the bounding box of the object.

You have to strictly follow the answer template:

<Analysis>:Here are some criterion and you need to check which of them should be considered shown in true or false.1. Contact point with the object: <true or false>2. Contact with the ground: <true or false>3. Position of the agent: <true or false, where the agent should be relative to the object, making a rough judgement is enough.>4. Rotation of the agent: <true or false, if true you need to specify along which directions the agent should not be rotated.>

<Discrimination function>:

```python

There are rules you have to follow: 1. Design metrics to determine if each agent meets the conditions and set a percentage threshold like 30% for the ratio of successful agents to judge this trial is successful, for the single agent case, there should not be a threshold. 2. Consider multiple factors like the agents' three-dimensional position and rotation, and if it is symmetrical around a certain rotation axis, then rotation around this axis does not affect. 3. There can be a tolerance for the agent's rotation. 4. Return in total three variables, which are the score, the judgement and a list of indices of agents meeting the criteria for this trial. 5. There should be no example usage in the code."

## Functional Pose Analyzer:

In an experiment to check whether a novel object possesses the affordance of a certain class, given the analysis on the affordance imagination and 5 interaction plans, you are a helper to analyze the simulation results. When an object is in a certain pose, based on the simulation results for the 5 plans, you need to determine whether this pose should be the functional pose of the object.

You need to strictly follow the answer template:

<Judgement>: <'true' or 'false'>

<Analysis>: <brief reasons why you think so>

Remember: 1. The criterion for judging success for each plan is the proportion of agents that meet the conditions. 2. In the simulation results, true means success and false means failure for this plan. 3. The functional pose should be identified only when the trial is successful through a reasonable interaction plan."
```

B. Details for Experiment.

1. Test on object data

For synthetic objects, we test each data sample for two runs. We run the experiment for two times, with requests of its correct functionality and an unrelated one, respectively. For example, we test a cup with two different requests like “Can it be used as a cup” for the first time and “Can it be used as c chair” for the second test.

For real data experiment, we reconstruct the object by 3D scanning and import it into the imagination environment. For each experiment of imagining a type of affordance, we perform three tests. First we choose the object with corresponding functionality and put it in a random pose and a functional pose, respectively, for example, a cup in a lying down and upright pose, we do this to check if our method can verify its affordance despite the conditions of the object.; besides, with the same request, we test with another object that does not provide the requested functionality but it is placed in a pose that affords its own functionality, like the example shown in Fig. 1 below. By this, we test if our method can tell the correct object from those that do not possess the desired affordance.

Fig. 1. Experiment on real objects

2. Evaluation on experiment result

For the results of each test sample, we use two evaluation criteria: affordance classification and functional pose prediction. For the imagination of affordance, the judgment is given by the scoring function, and the affordance is verified if the simulation results of the agent satisfy all the conditions taken into considerations and in the case of multiple agents the proportion of agents satisfying the conditions exceeds a threshold.

For the functional pose prediction, for each object for which affordance is verified the system stores its convinced functional pose in which simulation result shows success as the functional pose, and if there are more than one success trials, the one with the highest score is selected.

We recruit volunteers to annotate the experiment result. For each object, we present it to the volunteer and ask, “Do you think it can be a(n) (object with requested affordance)?” For each predicted functional pose, with the result of imagination, we ask the volunteer “Do you think this is the functional pose for (object with requested affordance)?”

In addition, for each trial of the real robot experiment, we show the experiment video and ask, “Do you think the task was successfully performed?” to ensure that the pipeline runs in a reasonable manner. The saved functional pose is then judged by our volunteer by asking "Do you think this is the functional pose for..." to check whether it is consistent with common sense when it provides functionality.

Tab1. Baseline Results on Real Object Affordance Classification

Baseline	**Random Pose (%)**	**Functional Pose (%)**
BLIP	71.4	96.4
GPT-4o	78.9	100