

## A. Results of Segment Anything

We first explain how we use **Segment Anything** to obtain the manipulation point:

Given one RGB image, We first frame a region as the segmentation region of the Segment Anything model, which is the area where garment are concentrated. Based on the segmentation area, Segment Anything model will return us several segmentation part, then we get the coordinates of the center point of each part by calculating the average value of all points in the segmentation part, which is also called candidate grasp point (as shown by the yellow circle in the Figure 1, 2 and 3). By comparing these candidate points, we choose the point closest to the exit (for washing machine) or with the highest height (for sofa and laundry basket) as the final retrieval point (as shown by the red star in the Figure 1, 2 and 3).

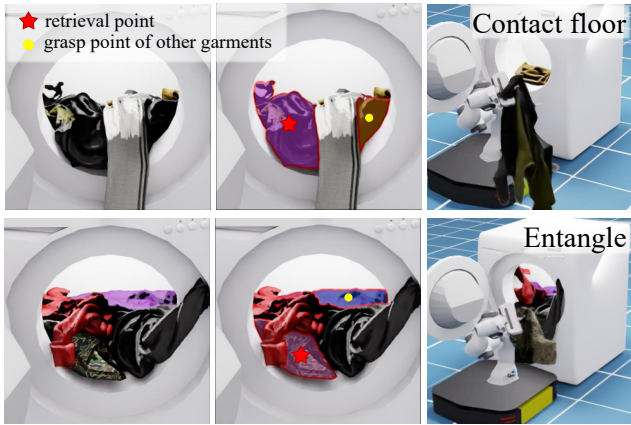


Figure 1. Segment Anything Results in **Washing Machine** Scene.

We want to share some interesting circumstances in Segment Anything results: we found that the success rate of sofa scene is quite high based on Segment Anything, while the success rate of washing machine and basket scene based on Segment Anything is not so high (you can check the success rate in Table 2 of the main paper).

We think this is due to the characteristics of different scenes. In sofa scene, the stacking and occlusion relationship between garments is not so serious, so the model can segment the whole garment well and get the exact center point of garment, but for other scenes, the stacking and occlusion relationship between garments is too serious, which makes Segment Anything no longer perform well.

## B. Finetune SAM for Support-M

Due to the complexity of cluttered garments, it is difficult to obtain GT real-world segmentation masks, especially in real world scenarios. We finetuned SAM using GT masks in simulation. Shown in Tab. 1, baseline success rate improved 11% in SIM (with **0.73** using GT segmentation as

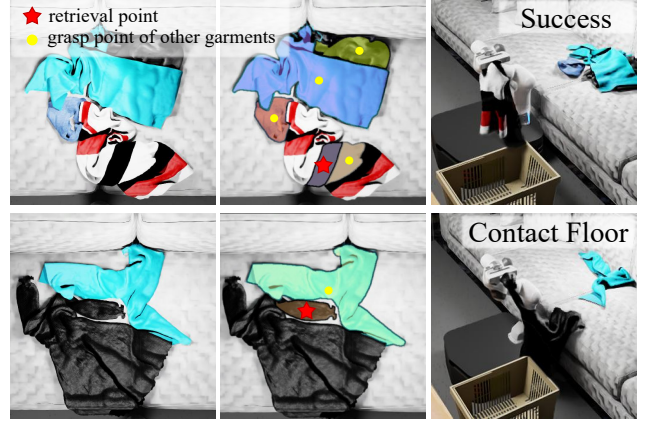


Figure 2. Segment Anything Results in **Sofa** Scene.



Figure 3. Segment Anything Results in **Basket** Scene.

upper bound), while real world success rate remains **8 / 15** (qualitative results in Fig. 4). Reasons: (1) only specific points, instead of all the segmented part, can be manipulated, which is only learned by our point-level representation; (2) gap between simulation and real-world images.

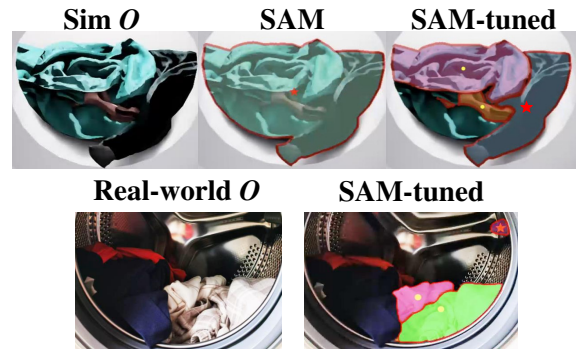


Figure 4. **Segmentation of SAM** before/after finetuning. Segmentation is improved in simulation but not in the real world.

Support-M	SIM	SIM (tune)	SIM (GT seg)	Real	Real (tune)	Ours
Succ Rate	0.56	0.67	0.73	8/15	8/15	0.81

Table 1. **Success Rate of Support-M** with different segmentations in simulation and real world. *GT seg* uses GT masks (upperbound).

### C. Results of Chatgpt-4o

We first explain how we use **Chatgpt-4o** to obtain the manipulation point:

Given one RGB image and one depth image, we first encode them in Base64 format and send them to Chatgpt-4o as conditions, while we also give Chatgpt-4o some relevant prompts to guide the model action, which is shown as below. Then Chatgpt-4o will return us one suitable retrieval point (if the point is not in the area of garments, we will make chatgpt-4o regenerate one point).

I will give you two images, one is RGB and the other is a depth map. The scene shows several pieces of clothing inside a basket. *(this line should be changed according to different scenes)* Assume you are a robot wanting to pick up each piece of clothing from the basket individually, *(this line should be changed according to different scenes)* ensuring that no other garments are accidentally pulled out during the process. Provide me with the optimal grabbing point as precisely as possible, which should be the coordinates of a pixel in the RGB image. The point you select must meet the basic requirements of being located on a piece of clothing. After generating a point, check if it is on the clothing. If not, select a new point and repeat the process until it is on the clothing. Note, you only need to return the precise coordinates of the pixel you consider optimal. And precision is very important. No additional information is required. For example: (201, 313)

Here we show some additional results about Chatgpt-4o in the scene of washing machine (Figure 5), sofa (Figure 6) and laundry basket (Figure 7).

We unfortunately find that the performance of Chatgpt-4o is far below expectations. The model seems to just select random points, which were unreasonable in most cases and unsuitable for effective manipulation.



Figure 5. Chatgpt-4o Results in **Washing Machine** Scene.

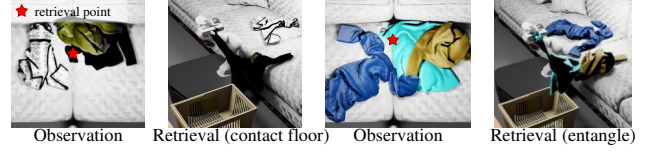


Figure 6. Chatgpt-4o Results in **Sofa** Scene.

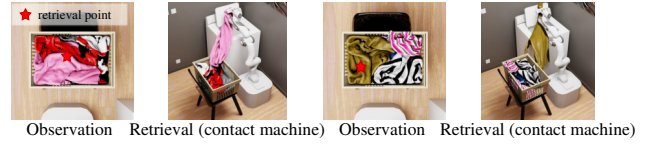


Figure 7. Chatgpt-4o Results in **Basket** Scene.

### D. Results of Real Machine

In this part we show the whole procedure about retrieval and adaptation in our real machine scenes (including washing machine, sofa and laundry basket). It is worth mentioning that our model can work well without fine-tuning based on online data in the real world, which proves that our model has good generalization and robustness.

We show the experimental results of the whole-process retrieval in the real-world washing machine scenario, real-world sofa scenario and real-world laundry basket scenario in Figure 8, 9 and 10 respectively.

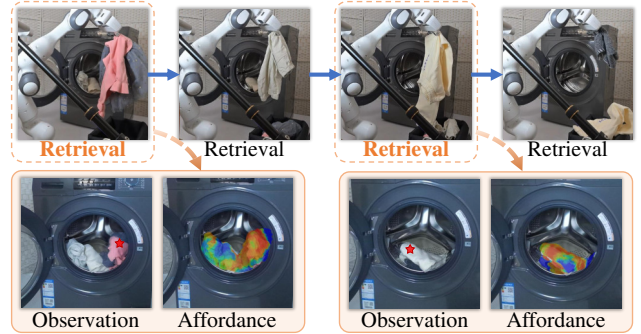


Figure 8. **Washing Machine** Retrieval Sequence (without adaptation).

Almost all retrieval operations are aware of the target garment's structure (the robot tends to grasp the middle part rather than the corners, even in scenarios with complex garment entanglements and severe occlusion) and the interrelation between garments (garments piled on the bottom or back generally do not produce highlights). Moreover, our affordance can also produce multi-modal output, in other



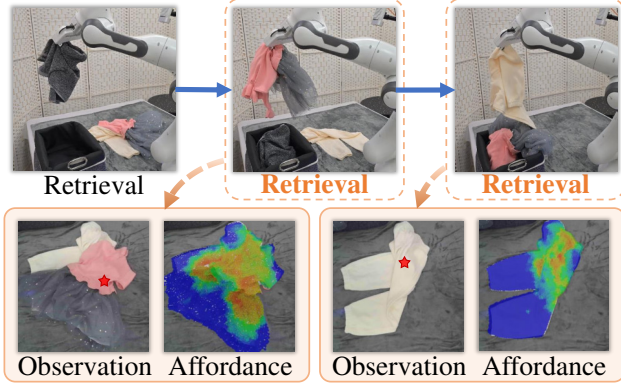


Figure 9. **Sofa** Retrieval Sequence (without adaptation).

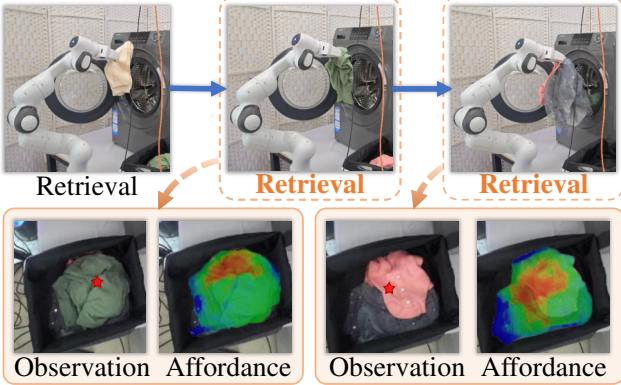


Figure 10. **Basket** Retrieval Sequence (without adaptation).

words, when multiple pieces of garments can be retrieved, multiple highlights appear, and they are all reasonable.

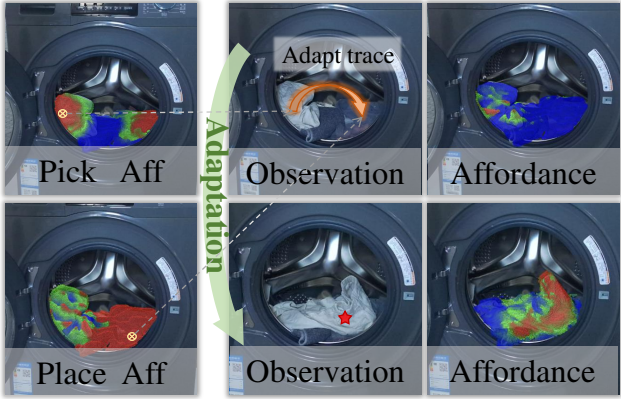


Figure 11. **Washing Machine** Adaptation.

We also tested our adaptation module in real-world scenarios. As shown in Figure 11, 12 and 13, when garments are severely tangled, the corresponding retrieval affordance appears poor. At this time, the model tends to execute an adaptation operation and find reasonable pick point and place point to adapt, and thus garments plausible for manipulation will exist.

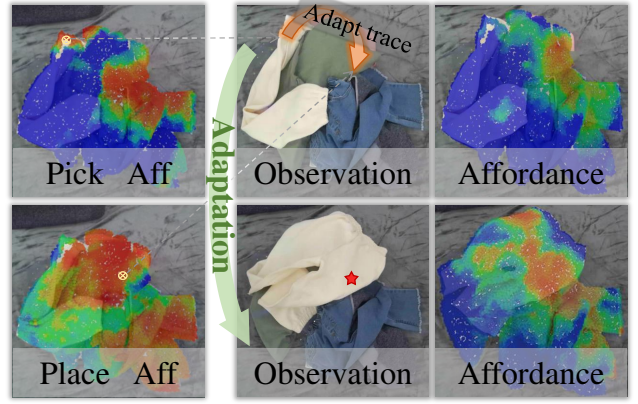


Figure 12. **Sofa** Adaptation.

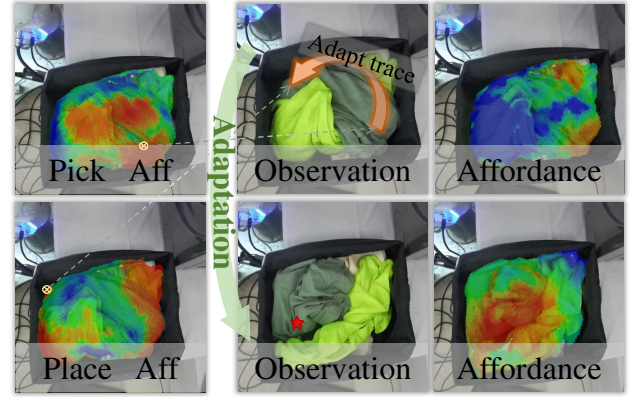


Figure 13. **Basket** Adaptation.

## E. Offline training details

We employed a random strategy to collect offline data, which enables us to select different but plausible points in similar observations, thereby capturing the multi-modal action distributions. Success rates of our offline trained models are **0.678, 0.792, 0.682** in 3 scenarios, consistently outperforming baselines (Tab.2 in main paper).

## F. Details of adaptation rounds

Tab. 2 shows the relation of adaptation rounds and success rates. With Tab.1 in main paper, we found up to 3 rounds of our proposed adaptation (instead of random adaptations) lead to plausible clutter states and make the success rate converge.

Rounds	3-rand	0	1	2	3
Success Rate	0.719	0.712	0.782	0.803	0.805

Table 2. **Success Rate on Different Adaptation Rounds.** 3-rand denotes 3 rounds of random adaptations.

## G. Why only raw PC without RGB?

We agree using color as additional info can better distinguish scenes with very similar point cloud. However, there is a significant gap in color information between simulation and reality, particularly in low-light scenes like washing machine.

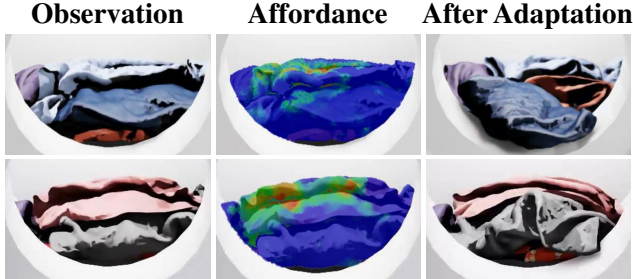


Figure 14. **Affordance and adaptation** of similar point cloud.

For two clutters with similar point clouds, the adaptation will help distinguish the clutters (Fig. 14).

Point cloud (depth) is sensitive to wrinkles and spatial relations between garments with similar colors (Fig. 15), enabling our method to effectively handle most clutters.

## Observation GT Objects Affordance

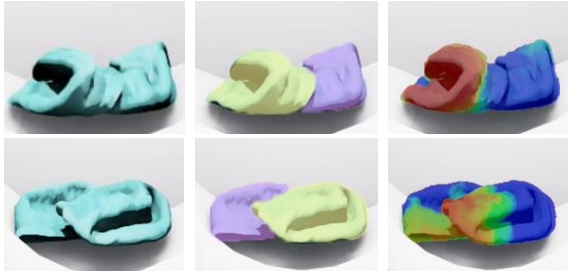


Figure 15. **Point cloud** can distinguish similar-colored garments.

## H. Generalization to novel clutters

Each clutter is specific due to various garment states. Besides, manipulation success rates of clutters with seen shapes, novel shapes in seen categories, and novel categories are **0.805**, **0.754** and **0.725** respectively. Fig. 16 shows affordance predictions in clutters with novel garment categories.

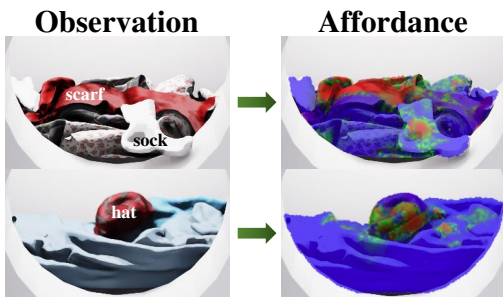


Figure 16. Affordance on novel categories (scarf, sock and hat).

## I. Limitations

For the simulation limitation, some extreme cases like knots between garments, cannot be simulated. For such difficult cases, manipulation requires more dexterous actions with 2 robots and even dexterous hands, instead of only parallel gripper's retrieving. Other garment configurations and correlations (e.g., two garments are entangled) are possible.