Supplementary Material for LOVMM

A Appendix

A.1 Natural Language-conditioned OVMM Task Details

We construct our natural language-conditioned OVMM tasks by extending the CLIPort benchmark [Shridhar *et al.*, 2022] into 10 different indoor scenes from the Matterport3D dataset [Chang *et al.*, 2017] using the Habitat simulator [Szot *et al.*, 2021], as shown in Figure 6. To evaluate the multi-task learning and zero-shot generalizing capabilities of our model, we build the seen tasks based on 2 simple scenes that only involve 4 workspaces with clean backgrounds such as the top of a flat sofa, while using 8 completely different scenes with 12 complex workspaces such as an uneven tabletop with messy background to construct the unseen tasks. Furthermore, the seen tasks are built based on the original benchmark without additional new objects, while the unseen OVMM tasks involve completely different novel objects from the Google Scanned Objects dataset [Downs *et al.*, 2022], as presented in Figure 7. The unseen tasks also use more ambiguous, close-to-life language instructions with more difficult manipulation requirements. The natural language instructions used for each task are presented in Table 4. For each task demonstration, the mobile robot is initialized in a random position in the scene. We name all the tasks with the corresponding scenes, target workspaces, and target manipulation descriptions. Unlike the original fixed workspace settings, we add an additional pick-and-place action step after the robot places the object from the previous workspace for more robust manipulation. For more details regarding the seen tasks, we refer the reader to the original CLIPort paper [Shridhar *et al.*, 2022].

Bedroom-Sofa-Pack-Boxes (Task-A)

In this task, multiple blocks of different colors and sizes and a brown container box are placed on the sofa in the bedroom A scene. The robot needs to pack all the blocks of specified colors into the box to fill it tightly, as shown in Figure 5(a). This task requires precise spatial and semantic understanding to handle different block sizes and colors. The task success rate (TSR) is defined as the volume of the specified blocks that are placed in the brown box divided by the total volume of all specified blocks in the scene.

Bedroom-Sofa-Pack-Google-Group (Task-B)

Multiple objects from the Google Scanned Objects dataset [Downs *et al.*, 2022] and a brown container box are placed on the sofa in the bedroom A scene. The robot needs to pack all the specified objects into the box. This task requires strong open-vocabulary understanding capabilities to deal with different objects, as shown in Figure 5(b). The TSR is defined as the volume of the specified objects that are placed in the container box divided by the total volume of all specified objects in the scene.

Bedroom-Sofa-Chair-Pack-Google-Seq (Task-C)

Similar to *bedroom-sofa-pack-google-group*, multiple objects from the Google Scanned Objects dataset [Downs *et al.*, 2022] are placed on the sofa, while a brown container box is placed on the chair in the bedroom A scene. The robot needs to pick up the specified objects from the sofa and place them into the container box on the chair. This task requires the model to perceive varying workspaces and generalize to handle different objects, as shown in Figure 5(c). The TSR is defined as the volume of the specified objects that are placed in the container box divided by the total volume of all the specified objects in the scene.

Bedroom-Sofa-Chair-Assemble-Kits (Task-D)

Multiple shaped objects are placed on the sofa, while a brown board with different shaped holes is placed on the chair in the bedroom A scene. The robot needs to pick up objects of specified shapes and colors from the sofa and place them into the corresponding holes on the board. This task is particularly difficult as it requires not only precise spatial manipulation but also accurate semantic understanding across different workspaces to match different shapes and colors, as shown in Figure 5(d). The TSR is defined as the number of correctly placed shaped objects divided by the total number of the shaped objects in the scene.

Livingroom-Carpet-Bed-Pack-Shapes (Task-E)

In this task, multiple shaped objects of randomized colors are placed on the carpet, and a brown container box is placed on the balcony bed in the living room A scene. The robot needs to pick up objects of specified shapes from the carpet and place them into the container box. This task requires strong environment perception capabilities to distinguish the shaped object from the complex carpet background, as shown in Figure 5(e). The TSR is defined as the number of correctly placed shaped objects divided by the total number of the shaped objects in the scene.

Livingroom-Carpet-Bed-Put-Block-In-Bowl (Task-F)

Multiple blocks of different colors are placed on the carpet, and multiple bowls of different colors are placed on the balcony bed in the living room A scene. The robot needs to pick up blocks of specified colors from the carpet and place them into the bowls of the corresponding colors. This task requires accurate spatial manipulation to pick the block from the complex carpet background and demands the semantic understanding ability to match blocks and bowls of corresponding colors, as shown in Figure 5(f). The TSR is defined as the number of correctly placed shaped objects divided by the total number of the shaped objects in the scene.

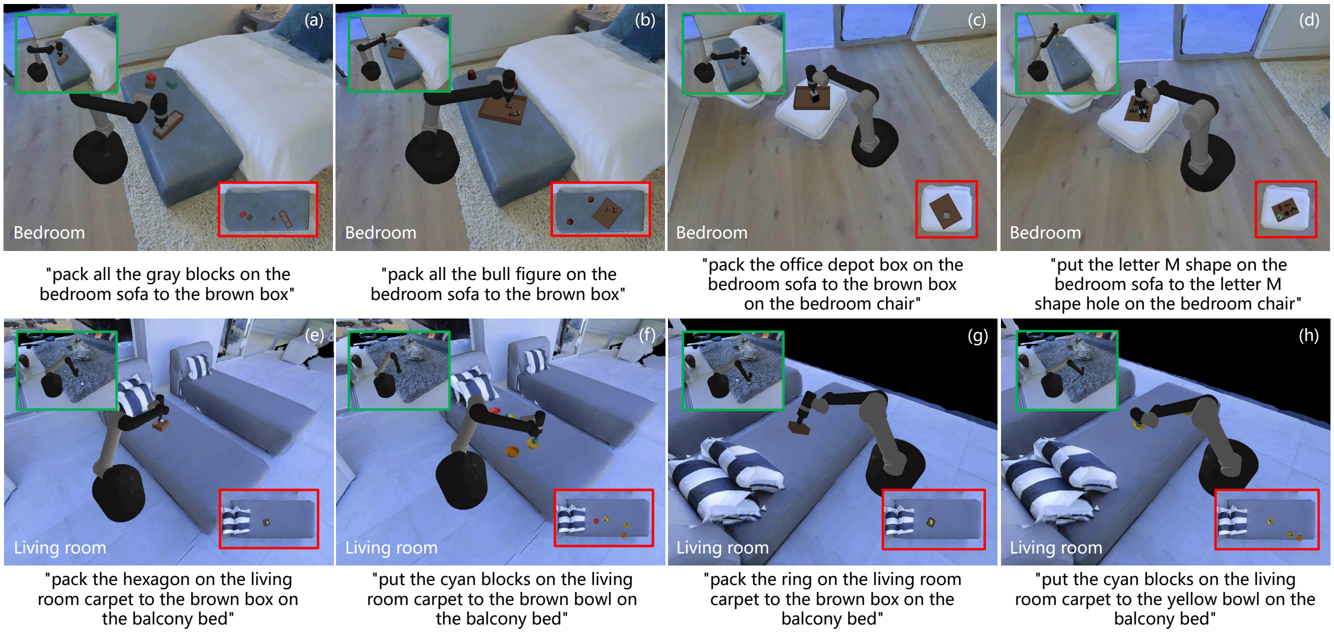

Figure 5: Natural language-conditioned seen OVMM tasks.

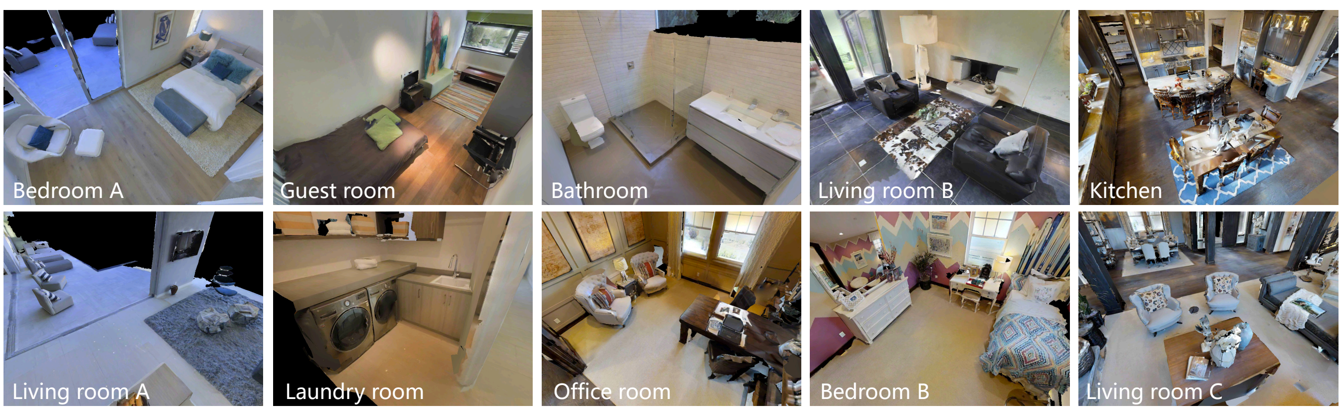

Figure 6: Scenes for OVMM tasks.

Livingroom-Carpet-Bed-Pack-Shapes-6dof (Task-G)

This task is identical to *livingroom-carpet-bed-pack-shapes*, except that the target box is initialized with a random fixed 6-DoF pose within the angular range of $\pm\frac{\pi}{6}$ and height limit of 10cm, which requires SE(3) manipulation. The task example is presented in Figure 5(g).

Livingroom-Carpet-Bed-Put-Block-In-Bowl-6dof (Task-H)

This task is identical to *livingroom-carpet-bed-put-block-in-bowl*, except that each target bowl is initialized with a random fixed 6-DoF pose using the same angular range and height limit as *livingroom-carpet-bed-pack-shapes-6dof*, which requires SE(3) manipulation. The task example is presented in Figure 5(h).

Guestroom-Bed-Table-Organize-Bottles (Task-I)

In this task, multiple different supplement bottles and some distractor objects are placed on the bed, while a brown wood container box is placed on the table in the guest room scene. The robot needs to pack all the supplement bottles into the box to fill it tightly, as shown in Figure 1(a). We use "supplement bottles" to refer to all the targets without specifying each one in the language instruction. This task requires precise spatial and semantic understanding to handle different kinds of unseen bottles of different sizes. The TSR is defined as the volume of the supplement bottles that are placed in the brown wood box divided by the total volume of all the supplement bottles in the scene.

Laundry-Bathroom-Basket-Pack-Shoes (Task-J)

Multiple kinds of different shoes are placed on the ground in the laundry room scene and a black basket is placed in front

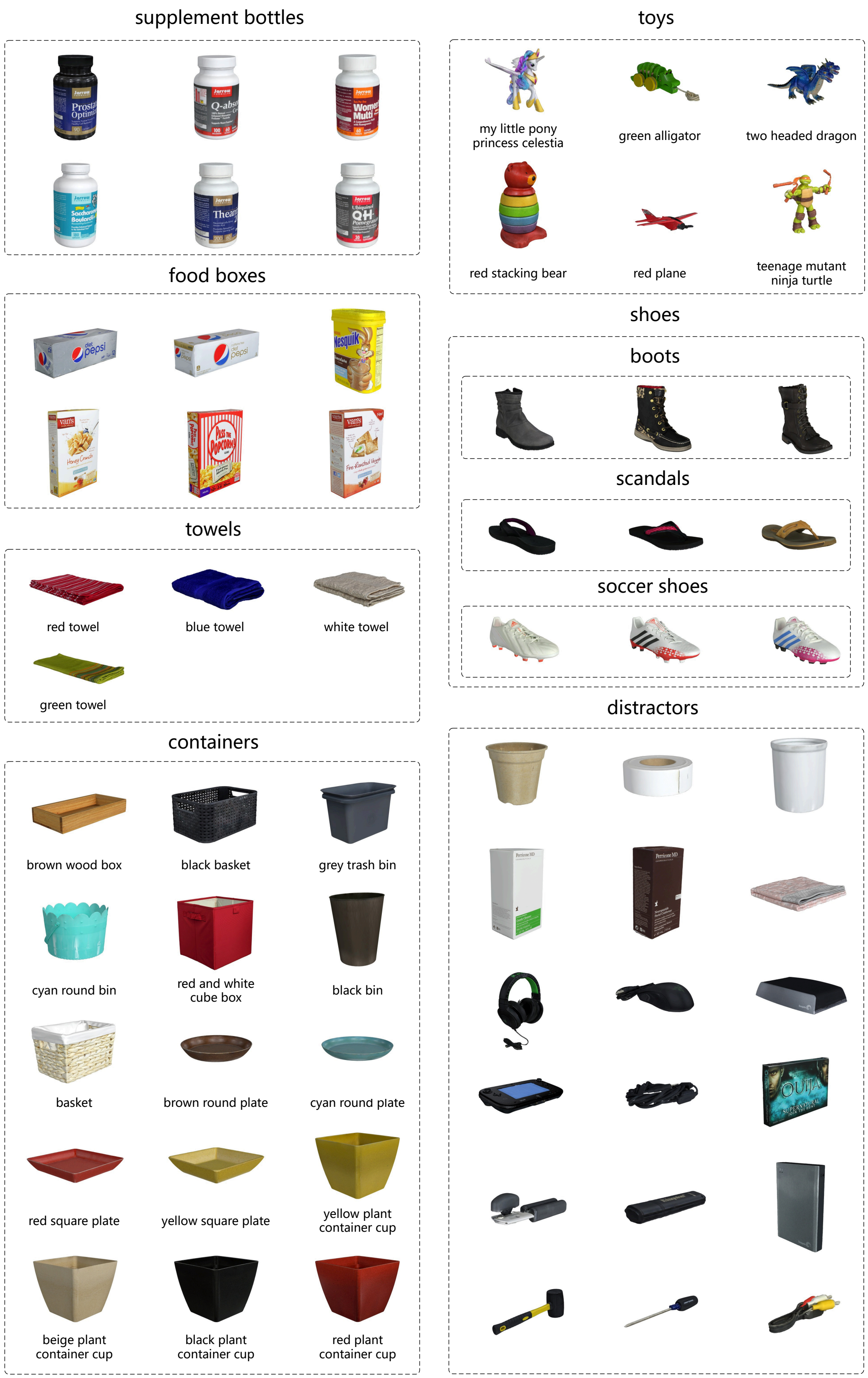

Figure 7: Objects for OVMM tasks.

Task	Natural Language Instruction
Task-A	"pack all the {color} blocks on the bedroom sofa to the brown box"
Task-B	"pack all the {google scan objects} on the bedroom sofa to the brown box"
Task-C	"pack the {google scan objects} on the bedroom sofa to the brown box on the bedroom chair"
Task-D	"put the {shapes} on the bedroom sofa to the {shapes} hole on the bedroom chair"
Task-E	"put the {shapes} on the living room carpet to the brown box on the balcony bed"
Task-F	"put the {color} blocks on the living room carpet to the {color} bowl on the balcony bed"
Task-G	"put the {shapes} on the living room carpet to the brown box on the balcony bed"
Task-H	"put the {color} blocks on the living room carpet to the {color} bowl on the balcony bed"
Task-I	"pack/put/pick all the supplement bottles from the bed to the brown wood box on the table in the guest room"
Task-J	"pack/put/pick the {shoes} in the laundry room to the black basket in front of the bathroom sink"
Task-K	"toss/put/pick the food boxes on the office room desk to the grey trash bin in the corner"
Task-L	"stack the {containers} from the living room carpet to the living room corner"
Task-M	"pick/pack/take the {toys} on the bedroom ground to the {containers} beside the drawer dresser"
Task-N	"take/put/pick the {towels} from the kitchen pantry basket and put it on the {containers} on the living room table"
Task-O	"pick/pack/take the {toys} on the bedroom ground to the {containers} beside the drawer dresser"
Task-P	"take/put/pick the {towels} from the kitchen pantry basket and put it on the {containers} on the living room table"

Table 4: Natural language instructions for OVMM tasks.

of the sink in the bathroom scene. The robot needs to pack all the specified shoes into the basket, as shown in Figure 1(b). We use the category name such as "boots" to refer to all the target shoes without specifying each one in the language instruction. This task requires strong open-vocabulary generalization capabilities to distinguish different unseen kinds of shoes. The TSR is defined as the volume of the specified kind of shoes that are placed in the basket divided by the total volume of all specified shoes in the scene.

Officeroom-Table-Corner-Tidy-Food (Task-K)

In the office room scene, multiple different food boxes and some distractor objects are placed on the desk, while a grey trash bin is placed in the corner. The robot needs to put all the food boxes in the trash bin. We use "food boxes" to refer to all the targets and use "toss" instead of the seen "pack" in the language instruction to simulate real-life scenarios, as shown in Figure 1(c). This task requires the model to correctly parse the language instruction and perceive the complex environment for accurate open-vocabulary manipulation. The TSR is defined as the volume of the food boxes that are placed in the trash bin divided by the total volume of all the food boxes in the scene.

Livingroom-Carpet-Corner-Stack-Cups (Task-L)

Multiple plant container cups of different colors are placed on the carpet and the corner in the living room B scene. The robot needs to pick up the specified cups from the carpet and stack them on top of the cups of corresponding colors in the corner. This task is particularly difficult as it requires not only precise spatial manipulation but also accurate perceiving capabilities to pick from the complex carpet background and match different cups across different workspaces, as shown in Figure 1(d). The TSR is defined as the number of correctly placed cups divided by the total number of the specified cups in the scene.

Bedroom-Ground-Drawer-Sort-Toys (Task-M)

In this task, multiple different toys and some distractor objects are placed on the ground, while three different containers are placed beside the drawer dresser in the bedroom B scene. The robot needs to pick up the specified toys and place them into the specified container. This task requires strong open-vocabulary generalization and semantic understanding capabilities to distinguish and match the specified unseen toys and containers, as shown in Figure 1(e). The TSR is defined as the volume of the specified toys that are correctly placed in the corresponding containers divided by the total volume of all the toys in the scene.

Kitchen-Livingroom-Basket-Table-Put-Towels (Task-N)

Multiple towels of different sizes and colors are placed in the pantry basket in the kitchen scene, and multiple plates of different sizes and colors are placed on the desk table in the living room C scene. The robot needs to pick up towels of specified colors from the pantry basket and place them into the

Method	bed-table-organize-bottles			basket-pack-shoes			desk-corner-tidy-food		
	1	10	100	1	10	100	1	10	100
Transporter6DoF	0.0	0.0	0.1	0.0	0.0	8.0	1.0	**21.5**	0.7
CLIPort	24.5	7.1	0.0	**21.5**	11.9	0.0	0.0	0.7	3.0
LOVMM	**55.9**	**26.7**	**8.3**	19.7	**21.3**	**23.2**	**7.9**	8.3	**26.6**
Method	carpet-corner-stack-cups			ground-drawer-sort-toys			basket-table-put-towels		
	1	10	100	1	10	100	1	10	100
Transporter6DoF	0.0	0.0	0.0	0.0	0.0	0.2	0.0	0.0	0.0
CLIPort	1.3	0.1	3.0	3.2	1.1	7.9	2.7	1.8	**10.5**
LOVMM	**7.3**	**1.3**	**8.3**	**5.3**	**2.7**	**11.2**	**6.2**	**3.1**	9.6
Method	ground-drawer-sort-toys-6dof			basket-table-put-towels-6dof			Average TSR		
	1	10	100	1	10	100	1	10	100
Transporter6DoF	0.0	0.0	0.0	0.0	0.0	0.0	0.1	2.7	1.1
CLIPort	0.3	0.4	3.2	0.0	0.0	1.3	6.7	2.9	3.6
LOVMM	**6.3**	**1.2**	**10.2**	**3.9**	**2.0**	**5.6**	**14.1**	**8.3**	**12.9**

Table 5: Tabletop manipulation tasks evaluation results.

plates of the specified colors, as shown in Figure 1(f). This task not only requires accurate spatial manipulation but also demands the semantic generalization ability to match unseen towels and plates. Moreover, it also requires accurate language parsing for the robot to open-vocabulary navigate to the correct workspaces. The TSR is defined as the volume of the specified towels that are correctly placed divided by the total volume of all the specified towels in the scene.

Bedroom-Ground-Drawer-Sort-Toys-6dof (Task-O)

This task is identical to *bedroom-ground-drawer-sort-toys*, except that the target container is initialized with a random fixed 6-DoF pose within the angular range of $\pm\frac{\pi}{4}$ and height limit of 10cm, which requires SE(3) manipulation. The task example is presented in Figure 1(g).

Kitchen-Livingroom-Basket-Table-Put-Towels-6dof (Task-P)

This task is identical to *kitchen-livingroom-basket-table-put-towels*, except that each target bowl is initialized with a random fixed 6-DoF pose using the same angular range and height limit as *bedroom-ground-drawer-sort-toys-6dof*, which requires SE(3) manipulation. The task example is presented in Figure 1(h).

A.2 Evaluation Results for Unseen OVMM Tasks

To further validate the performance of LOVMM, we implement the tabletop manipulation baselines with manually annotated fixed navigation routes (FNR) to complete unseen OVMM tasks. All models are trained on the same set of 100 demonstrations from seen tasks. Detailed evaluation results are presented in Table 6. It shows that LOVMM outperforms both baselines across all tasks with a notable margin. Specifically, LOVMM achieves over 18.0% higher TSR than FNR+CLIPort for *Task-K* and generalizes nearly 3 times better than FNR+Transporter6DoF for *Task-J*. Moreover, our model is capable of solving challenging tasks such as *Task-L*, where both baselines fail to complete even a single instance. Such results further demonstrate the strong manipulation learning and generalizing abilities of LOVMM in solving complex OVMM tasks.

Method	Task-I	Task-J	Task-K	Task-L
FNR+Transporter6DoF	0.0	7.6	0.6	0.0
FNR+CLIPort	0.0	0.0	2.8	0.0
LOVMM	**7.3**	**21.2**	**21.0**	**3.9**
Method	Task-M	Task-N	Task-O	Task-P
FNR+Transporter6DoF	0.0	0.0	0.0	0.0
FNR+CLIPort	7.5	6.5	3.0	1.3
LOVMM	**8.9**	**7.8**	**9.1**	**3.2**

Table 6: Unseen OVMM tasks evaluation results.

A.3 Evaluation Results for Tabletop Manipulation Tasks

We evaluate the models on tabletop manipulation tasks by using the same target workspace observation for all models to complete unseen OVMM tasks. In this way, the tasks are simplified to completing tabletop manipulation at each workspace without any navigation requirements. The tasks are named the same as OVMM tasks but without the scene names. The detailed evaluation results are presented in Table 5. It shows that LOVMM outperforms all the other models in $21/24 = 87.5\%$ of the evaluated tasks. Specifically, LOVMM performs exceptionally well compared with the baselines in many tasks, achieving a best 55.9% TSR for *bed-table-organize-bottles* with only 1 demonstration, and a 26.6% performance for *desk-corner-tidy-food*. For tasks that are more challenging, CLIPort struggles to achieve less than 5.0% TSR with limited expert demonstrations and Trans-

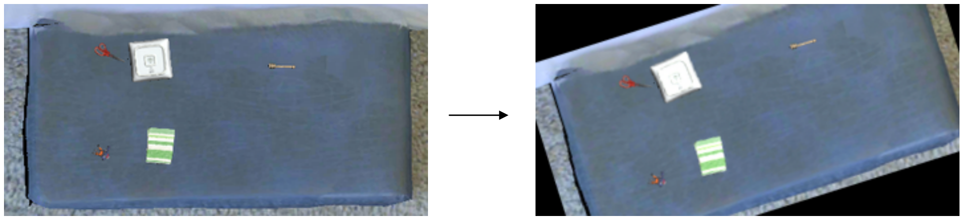

Figure 8: Data augmentation for training observation samples.

porter6DoF is unable to solve most of the tasks. On the other hand, the 6.3% performance of LOVMM in the challenging *ground-drawer-sort-toys-6dof* that involves both complex environments and 6-DoF manipulation compared with the baselines' near-zero TSR also indicates that our model is capable of adapting to unseen workspace environments and open-vocabulary generalize to different object attributes.

A.4 Data Augmentation

Following the data augmentation setting in the original CLIPort implementation [Shridhar *et al.*, 2022], we apply random SE(2) transformations to the training observation samples for better spatially-equivariant representation learning. Additionally, we apply multiple object lighting settings, adjust brightness, and add random noises and Gaussian blur to the observation samples to enhance the model's generalization ability, as shown in Figure 8.

A.5 Limitations

Imbalanced Dataset

Our proposed natural language-conditioned OVMM tasks involve a wide range of scenarios, covering single-step manipulation to long-horizon tasks. As a result, the dataset is heavily imbalanced. In order to have a fair and direct comparison with the baseline methods, we adopt the original random sampling strategy used by CLIPort, which inevitably introduces learning bias across tasks. In this case, more demonstrations may lead to sparser task coverage, resulting in degraded manipulation performance. Future work could incorporate weighted sampling methods such as [Team *et al.*, 2024] to address this issue.

Simplified Task Settings

Although LOVMM's simple and low-cost settings offer a practical and scalable solution for real-world deployment, real-world tasks often require multi-view observations and continuous 6-DoF actions, such as pouring a cup of water into the container. However, multi-view inputs may surge computational cost, and setting up multiple cameras is non-trivial. Future work could explore more flexible camera setups and action settings. Furthermore, in real-world situations, navigation and manipulation are often related (e.g., going to the bedroom needs opening the door first), so developing an entangled pipeline for OVMM would be a promising direction.