

# IGen: Scalable Data Generation for Robot Learning from Open-World Images

## Supplementary Material

### 1. Single-Image Scene Reconstruction Details

In this section, we describe how IGen reconstructs the 3D scene from a single open-world RGB image to facilitate robot data generation, as illustrated in Fig. 1.

We use Metric3Dv2 [3] to estimate the depth and convert image pixels into a point cloud. For open-world images, the focal length is fixed to 1000, while for specific camera types (e.g., iPhone or Microsoft Kinect Camera), we adopt their corresponding intrinsic parameters. The resulting point cloud preserves the same spatial resolution and dimensions as the original RGB image.

For object-level reconstruction, we utilize the TRELLIS model [6] to perform monocular 3D reconstruction and convert the outputs into colored point clouds. The input images to TRELLIS are pre-processed using segmentation masks obtained from SAM [4], ensuring that the reconstruction focuses on the target objects.

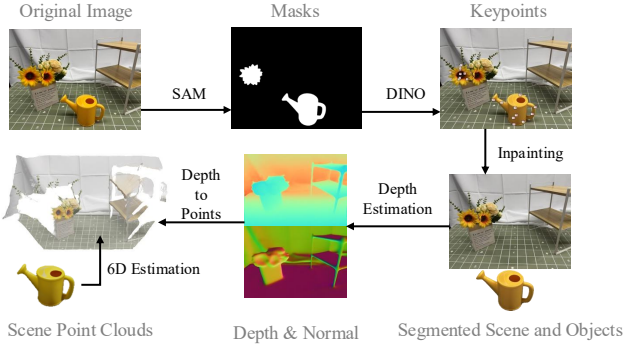


Figure 1. Single-Image Scene Reconstruction Pipeline.

### 2. Simulation Environment Details

This section describes the details of building the robotic manipulation platform in simulation. We adopt Isaac Sim as the simulation environment and deploy both the Franka Emika Panda and Franka Research 3 robotic arms within it. For motion planning, we utilize Curobo as the solver, which computes feasible trajectories given the target end-effector poses. We use the default illumination settings in the simulation environment.

As shown in Fig. 2, we place a virtual depth camera at the origin  $[0, 0, 0]$  of the simulation scene, oriented relative to the robot’s base frame. The camera’s focal length is set to match that used in the depth estimation module. The robotic arm is positioned at a predefined spatial coordinate  $[x_r, y_r, z_r]$  within the reconstructed point cloud space. During robot motion, the camera operates at a sampling rate

of 30 fps, capturing synchronized RGB and depth frames for subsequent reconstruction of the robot’s dynamic point cloud sequences.

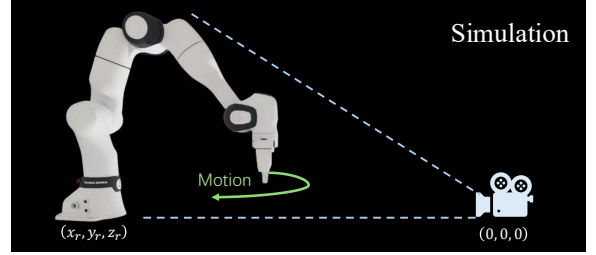


Figure 2. **Robot and Camera Placement in Simulation.** In simulation platforms such as IsaacSim, the virtual camera is placed at the position  $(0, 0, 0)$ , while the robotic arm base is positioned at the corresponding point in the point cloud, denoted as  $(x_r, y_r, z_r)$ . RGB and depth data are collected during the robotic arm’s motion.

### 3. Manipulation Synthesis Details

We divide the point cloud sequence into three components: the background point cloud, the robot point cloud, and the object point cloud. Among them, the robot and object point clouds are dynamic, while the background point cloud remains static. We use GraspGen [5] for grasp pose estimation.

The grasp width is inferred from the inter-point distance along the principal axis of the reconstructed object point cloud. At the grasping moment  $t_g$ , the end-effector pose is denoted as  $\mathbf{T}_{t_g}$  and the object pose as  $\mathbf{T}_{\text{obj}, t_g}$ . During the subsequent manipulation at time  $t$ , given the current end-effector pose  $\mathbf{T}_t$ , the object pose in the scene can be computed through rigid-body transformation as:

$$\mathbf{T}_{\text{obj}, t} = \mathbf{T}_t \mathbf{T}_{t_g}^{-1} \mathbf{T}_{\text{obj}, t_g}. \quad (1)$$

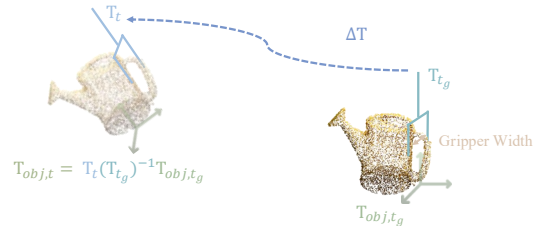


Figure 3. **Point Cloud Synthesis during Manipulation.** At time  $t_g$ , the object is grasped. The gripper width is calculated based on the point cloud, and the transformation of the object point cloud at time  $t$  is computed according to the end-effector’s pose.

## 4. Real-World Experiment Details

**Hardware Setup.** As shown in Fig. 5, we set up a real-world evaluation environment using the Franka Research 3 robotic arm. A Microsoft Kinect camera is placed in front of the robot to provide RGB visual input. The robotic arm performs task operations on the tabletop.

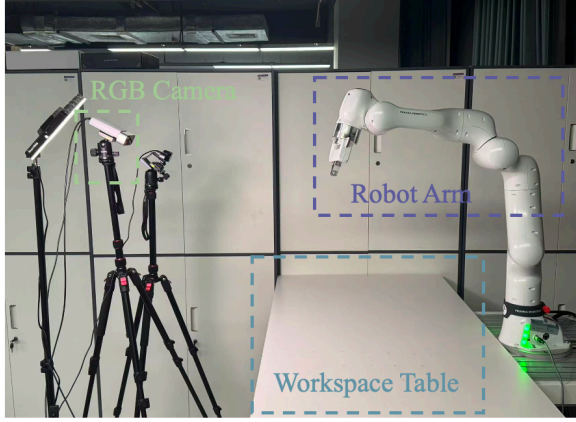


Figure 5. **Hardware Setup.** Our experimental setup consists of a Franka Research 3 robotic arm, a tabletop workspace, and a global RGB camera.

**Spatial Randomization.** For real-world data collection, we sample random object positions within a  $40\text{ cm} \times 30\text{ cm}$  tabletop grid. In IGen, spatial randomization is performed based on the point cloud of the placement area (e.g., the tabletop surface). We define a  $200 \times 150$  pixel grid as the sampling region for randomization, ensuring that the spatial distribution closely matches that of the real-world setup. Regarding the spatial randomization of real-world data and IGen-generated data, see Fig. 4, 6, and 7.

**Task Evaluation.** We design diverse manipulation tasks involving complex interactions between objects and the surrounding scene. For each task, we conduct 12 independent trials. Object initial positions are sampled on a  $30\text{ cm} \times 25\text{ cm}$  tabletop grid, with a spacing of 7 cm between adjacent positions. All models are evaluated using the same set of initial object positions.

**Policy Learning.** This section describes the fine-tuning process of policy. We fine-tune  $\pi_0$ -base [1] for 30k training steps using LoRA [2] with a batch size of 8. The model takes as input a single  $224 \times 224$  RGB image and the absolute joint positions as the state, and predicts a 10-step relative joint angle action chunk. Training is conducted on a single NVIDIA A40 GPU, requiring approximately 10.8 hours per training. The performance of the model in real-world deployment is shown in Fig. 8 and 9.

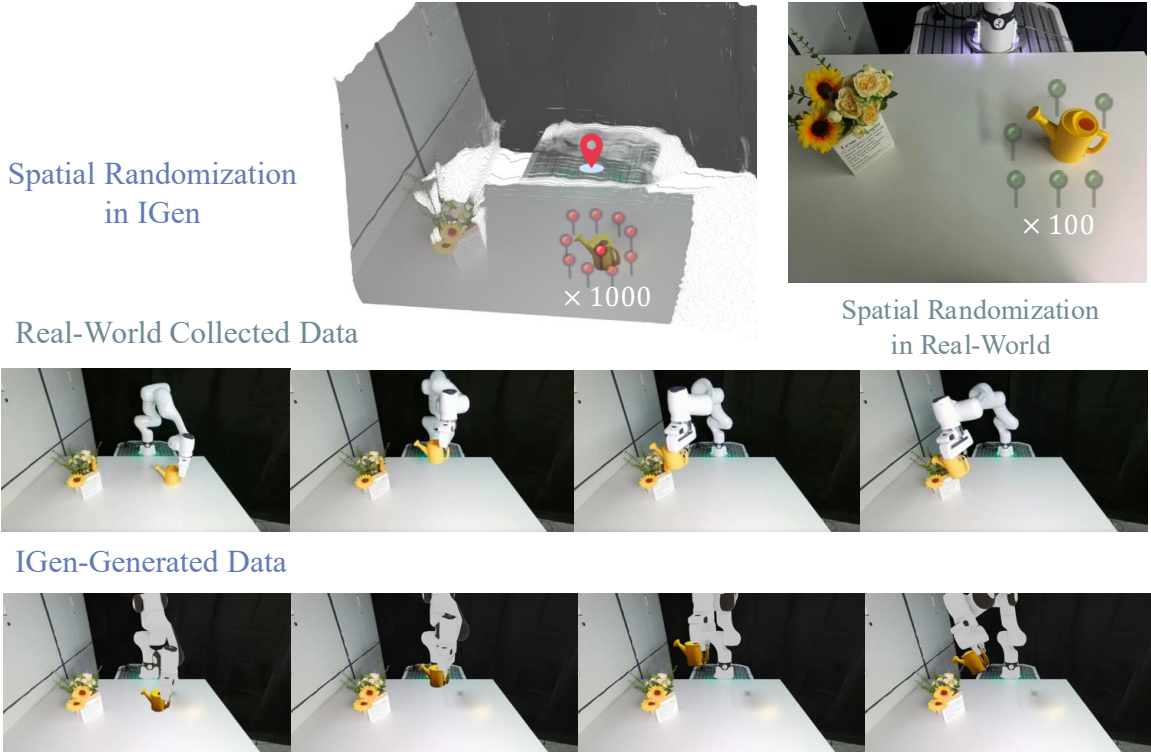


Figure 4. Spatial randomization of real-world data and IGen-generated data. The task is *Grab the watering can and water the flowers*.

Spatial Randomization  
in IGen



Real-World Collected Data



Spatial Randomization  
in Real-World

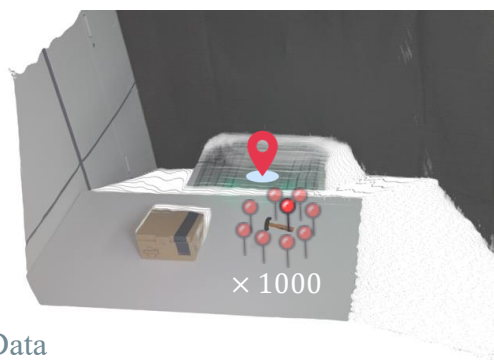


IGen-Generated Data

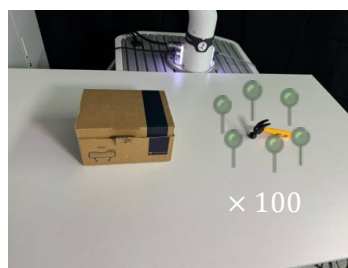


Figure 6. Spatial randomization of real-world data and IGen-generated data. The task is *Pick up the bottle and place it into the basket.*

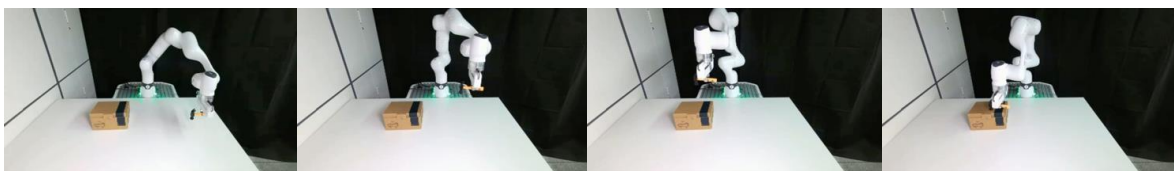
Spatial Randomization  
in IGen



Real-World Collected Data



Spatial Randomization  
in Real-World



IGen-Generated Data



Figure 7. Spatial randomization of real-world data and IGen-generated data. The task is *Use the hammer to hit the cardboard box.*



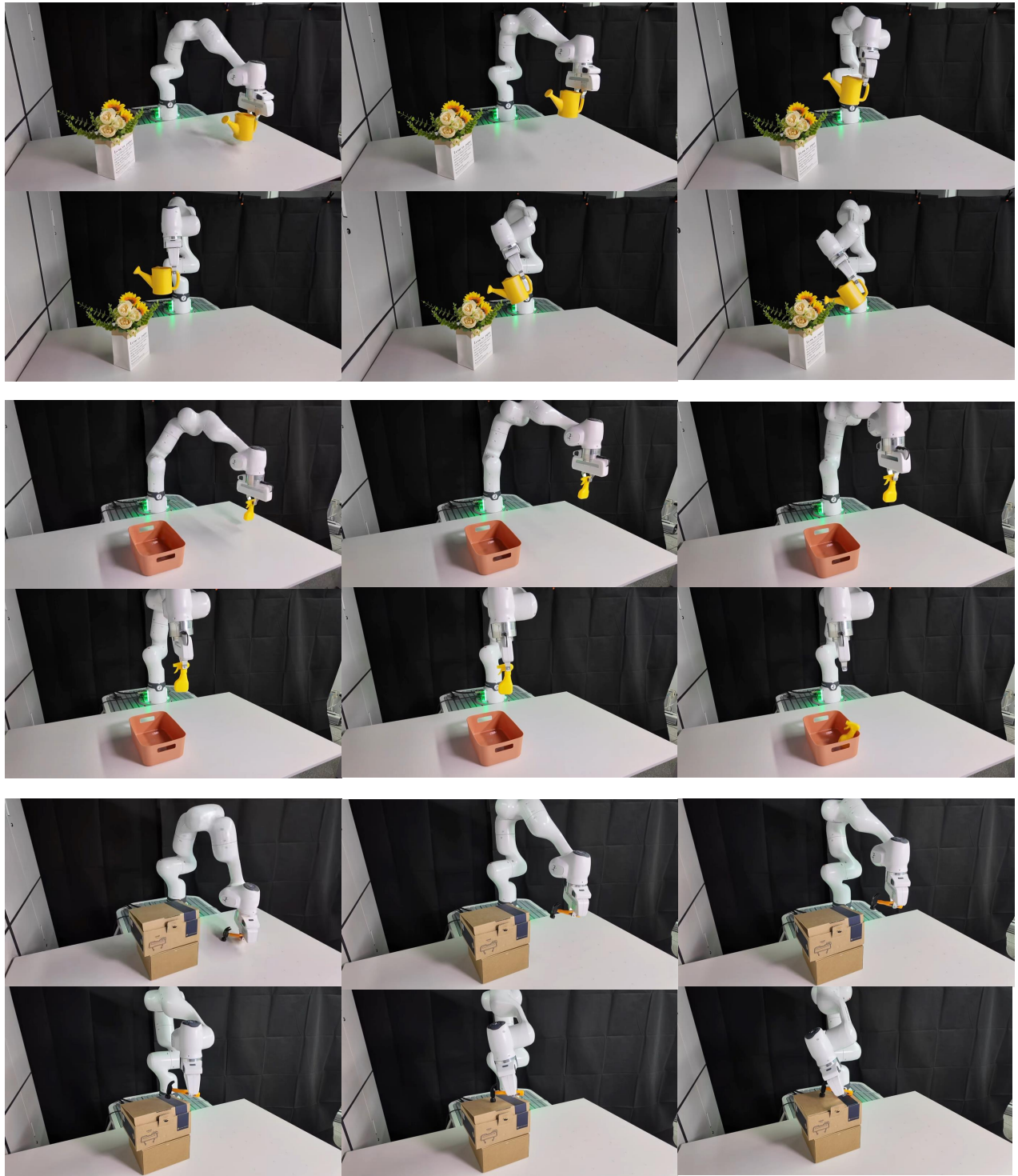


Figure 8. **Real-World Deployment of Policy trained with IGen-Generated Data.** The task instructions are as follows: *Grab the watering can and water the flowers. Pick up the bottle and place it into the basket. Use the hammer to hit the cardboard box.*

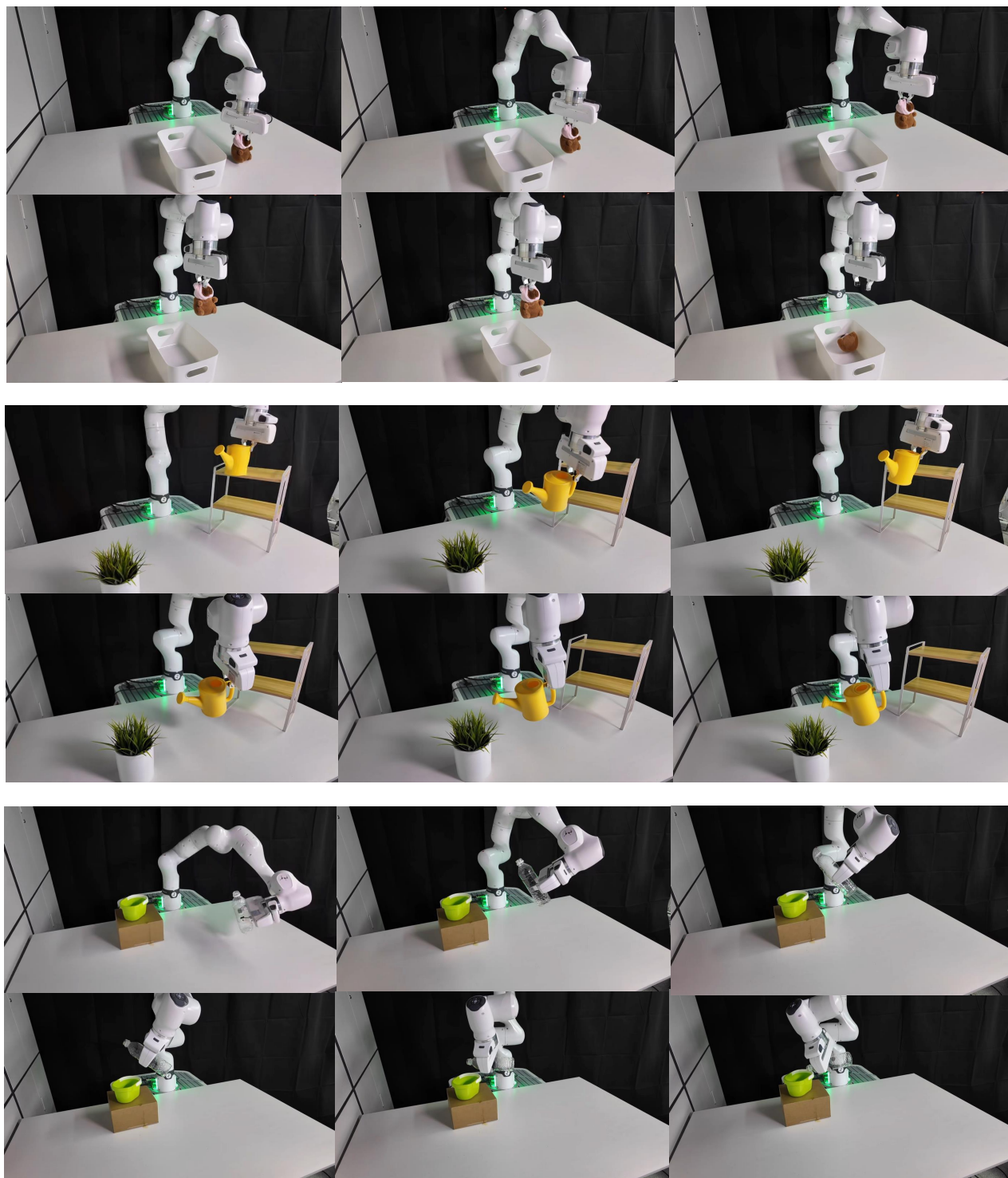


Figure 9. **Real-World Deployment of Policy trained with IGen-Generated Data.** The task instructions are as follows: *Grasp the toy and put it into the bin. Use the watering can on the cabinet to water the flowers. Pour water from the plastic bottle into the container.*

## 5. Ablation Study

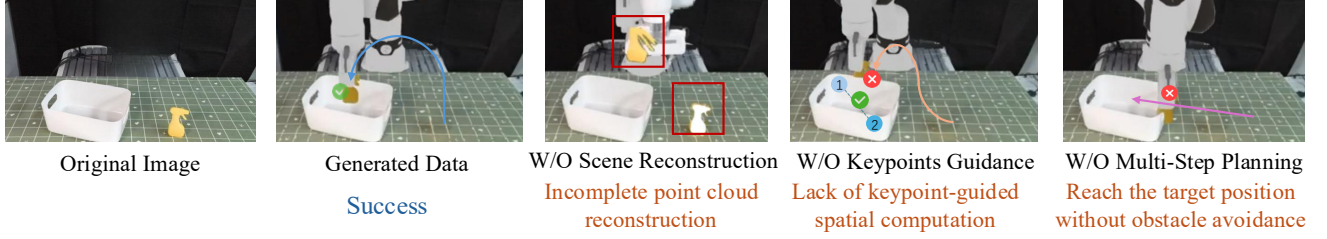


Figure 10. Ablation study on different components of the pipeline.

## 6. Details for Keypoints Generation

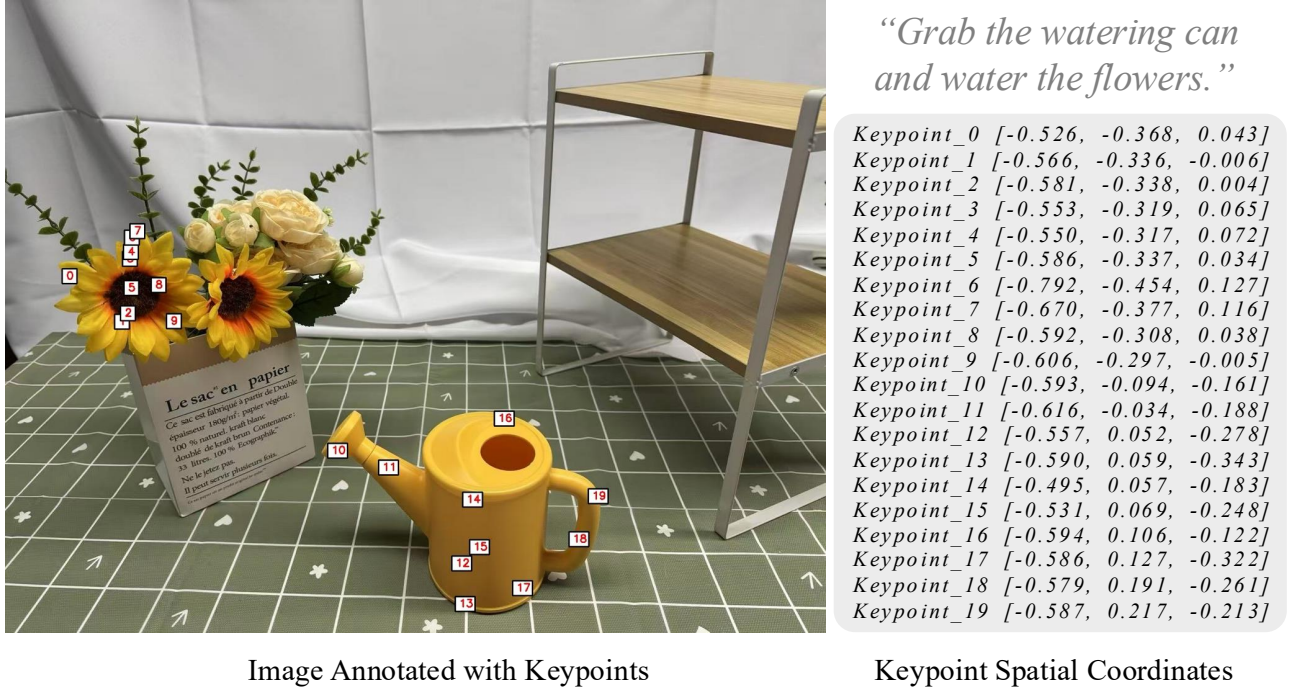


Figure 11. Image with annotated keypoints, keypoint coordinates, and task description as input for VLM.

## References

- [1] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. pi.0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024. 2
- [2] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. 2
- [3] Mu Hu, Wei Yin, Chi Zhang, Zhipeng Cai, Xiaoxiao Long, Hao Chen, Kaixuan Wang, Gang Yu, Chunhua Shen, and Shaojie Shen. Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 1
- [4] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023. 1
- [5] Adithyavairavan Murali, Balakumar Sundaralingam, Yu-Wei Chao, Wentao Yuan, Jun Yamada, Mark Carlson, Fabio Ramos, Stan Birchfield, Dieter Fox, and Clemens Eppner. Graspgen: A diffusion-based framework for 6-dof grasping with on-generator training. *arXiv preprint arXiv:2507.13097*, 2025. 1
- [6] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21469–21480, 2025. 1