

7 Appendix

In this section, we provide additional implementation details of RoVi-Aug and our physical experiments.

7.1 Robot Augmentation

7.1.1 Training Data Generation

To train our robot segmentation and Robot-to-Robot generation models, we use the Robosuite simulator [142] to generate a large dataset of paired robot images with corresponding masks with randomly sampled robot poses and camera poses. The sampling procedure is as follows: The robot pose is specified by the end-effector pose. The translation component is sampled uniformly with $(x, y, z) \in [-0.25, 0.25] \times [-0.25, 0.25] \times [0.6, 1.3]$ (unit in meters). For the rotation component, we parameterize it as [inward, rightward, z_axis]. To bias the unit vector z_axis towards pointing downward, we parameterize it using spherical coordinate θ, ϕ where θ (zenith angle) is sampled from a normal distribution $\mathcal{N}(\pi, \pi/3.5)$ and ϕ (azimuthal angle) is uniformly sampled between 0 and 2π .

After sampling the robot pose, we randomly sample the camera pose with the following procedure: The position is sampled from a half hemisphere with radius $r \in \mathcal{N}(0.85, 0.2)$ and zenith angle $\theta \in \mathcal{N}(\pi/4, \pi/2.2)$, and azimuthal angle $\phi \in \text{Unif}[-\pi \cdot 3.7/4, \pi \cdot 3.7/4]$. The viewing direction is towards the center of the hemisphere, which we offset as the gripper position. We also sample camera field of view between 40 and 70. Finally, we randomly perturb the camera pose with noises.

We randomly sample robot poses, and for each robot pose, we randomly sample 5 different camera poses. In addition to pure random sampling, we also add some camera poses and robot poses similar to those in the RT-X datasets and add perturbations. We obtain paired images between different robots and their segmentation mask from Robosuite, and we add random brightness augmentation with range $[-40, 40]$ to the source robot images to increase the robustness of the segmentation model and R2R model to real-world lighting. In this way, we obtain 800k images for each of the 4 robot types: Franka, UR5, Sawyer, and Jaco. See Fig. 5 for some example images.

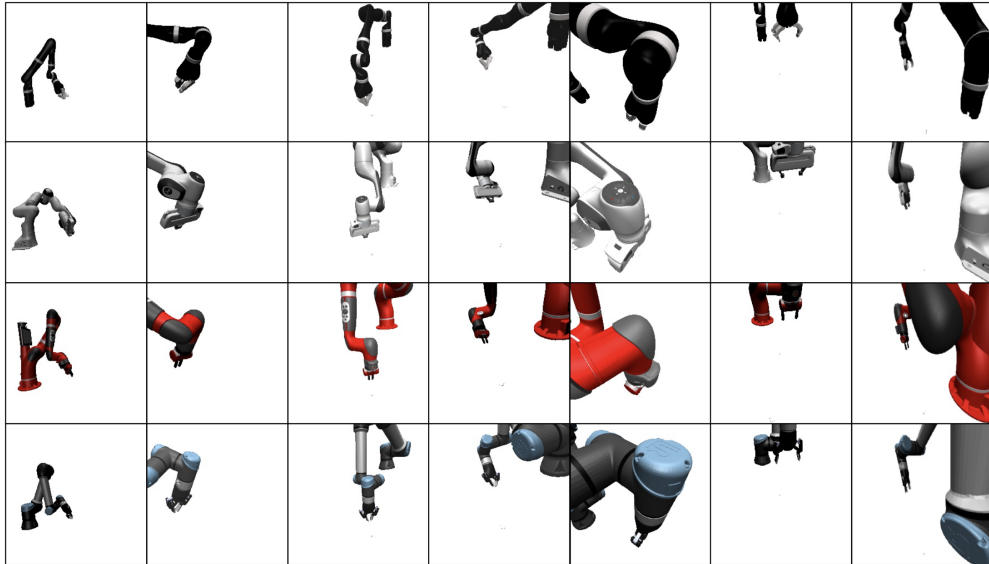


Figure 5: **Example of paired images for training the R2R model.** We use Robosuite [142] to generate pairs of Jaco, Franka, Sawyer, and UR5 at the same pose.

To create the dataset for training the segmentation model, we paste the generated robot image onto backgrounds from ImageNet [136]. See Fig. 6 for some example images.



Figure 6: Example of pasted images on ImageNet used for training the segmentation model.

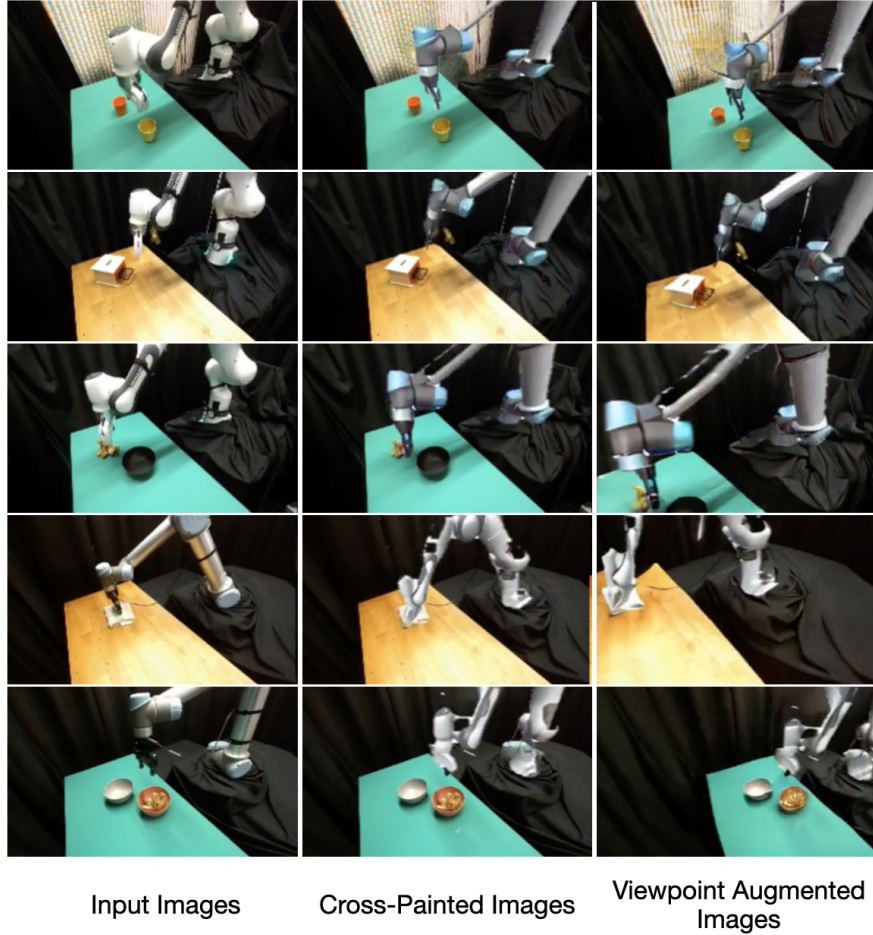


Figure 7: **Example of RoVi-Aug results.** We show some example results of RoVi-Aug applied to the training images of the 5 tasks.

7.1.2 Throughputs of Robot Augmentation

Our pipeline is well optimized to seamlessly transfer the different types of robots. Specifically, our robot segmentation model, robot-to-robot model, and robot inpainting models can operate in parallel to efficiently process batchified video frames. We validate the throughputs of each modules as shown below: Robot segmentation model achieves 4.1FPS, Robot-to-Robot achieves 3.2FPS, and robot inpainting model achieves 4.6FPS.

7.1.3 Example Augmented Images

In Fig. 7, we show some example results of RoVi-Aug applied to the training images of the 5 tasks. The left column is the original images; the middle column is the cross-painted images using the

robot augmentation pipeline; the right column shows the view augmented images applied on top of the robot augmented images. The black regions in the generated robot are due to incomplete segmentation mask (missing some regions in the generated robot) when pasting the generated robot to the original image. We can see that in general, RoVi-Aug generates diverse view angles of the target robot performing the task of interest.

7.2 Additional Details of the RT-X Data Augmentation

We select a subset of the OXE datasets, including Roboturk [145] for Sawyer, Jaco Play [146] for Jaco, Berkeley UR5 [147] for the UR5 robot, and 4 datasets for Franka: Viola [148], Austin Sailor [149], Austin Buds [150], and Hydra [151], and apply RoVi-Aug on this subset to generate Franka and UR5 synthetic images and finetune Octo [143] on the augmented datasets. Specifically, since the “Pick Cup” and “Bowl in Oven” are from the Jaco Play [146] dataset, and we augment them into UR5, the policy would be familiar with UR5 performing the task, even though the distractor objects are different and the camera angles are different, and thus exhibits better fine-tuning sample efficiency on our UR5 fine-tuning data. Similarly, the “Sweep Cloth” and “Transport Tiger” tasks are from the Berkeley UR5 [147] dataset, and we augment them into Franka, it exhibits better fine-tuning sample efficiency on our Franka fine-tuning data.