

## 399 A Formulation Derivations

### 400 Eye-in-hand Local-frame Reparameterization

401 For an eye-in-hand camera setup, let the current camera frame at time  $t$  be denoted as  $\mathbf{X}_{C_t} \in \text{SE}(3)$ ,  
 402 where  $\text{SE}(3)$  represents the Special Euclidean group. The initial and previous camera frames are  
 403 denoted as  $\mathbf{X}_{C_0}$  and  $\mathbf{X}_{C_{t-1}}$ , respectively. Let the end-effector pose at time  $t$  be denoted as  $\mathbf{X}_{H_t}$ .  
 404 Expressed in the initial eye-in-hand camera frame, and relative to the initial end-effector pose, the  
 405 delta transformation  $\Delta\mathbf{X}_{H_t}$  is:

$$\Delta\mathbf{X}_{H_t} := \mathbf{X}_{H_0}^{-1} \mathbf{X}_{H_t} = \left( {}^{C_0}\mathbf{X}_{H_0} \right)^{-1} \left( {}^{C_0}\mathbf{X}_{H_t} \right).$$

406 Similarly, the relative pose with respect to the previous timestep is:

$$\delta\mathbf{X}_{H_t} := \mathbf{X}_{H_{t-1}}^{-1} \mathbf{X}_{H_t} = \left( {}^{C_{t-1}}\mathbf{X}_{H_{t-1}} \right)^{-1} \left( {}^{C_{t-1}}\mathbf{X}_{H_t} \right).$$

### 407 Fixed Points of the Pose Mirroring Mapping

408 For successful transfer of the mirrored trajectory to the current robot configuration, it is crucial that  
 409 the initial state of the manipulation trajectory lies near a fixed point of the pose mirroring mapping.  
 410 That is, the pose should remain close to configurations that are invariant under the mirror mapping  
 411  $\mathcal{M}(\cdot)$ . For an arbitrary pose  $\mathbf{X} \in \text{SE}(3)$ , the mirroring mapping is defined as  $\mathcal{M}(\mathbf{X}) = \mathbf{E}\mathbf{X}\mathbf{E}$ , as  
 412 introduced in Eq. (1), where  $\mathbf{E} = \text{diag}([-1, 1, 1, 1])$ . The fixed points of this mapping must satisfy  
 413  $\mathbf{E}\mathbf{X}\mathbf{E} = \mathbf{X}$ . Applying the mirroring operation to a general pose  $\mathbf{X} \in \text{SE}(3)$ :

$$\mathbf{E} \begin{bmatrix} r_{xx} & r_{yx} & r_{zx} & t_x \\ r_{xy} & r_{yy} & r_{zy} & t_y \\ r_{xz} & r_{yz} & r_{zz} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{E} = \begin{bmatrix} r_{xx} & -r_{yx} & -r_{zx} & -t_x \\ -r_{xy} & r_{yy} & r_{zy} & t_y \\ -r_{xz} & r_{yz} & r_{zz} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

414 Therefore, for a pose to be a fixed point under  $\mathcal{M}$ , the following symmetry conditions must hold:

$$r_{yx} = r_{zx} = r_{xy} = r_{xz} = 0, \quad t_x = 0,$$

415 which means that the rotation matrix corresponds to a pure rotation about the x-axis. The translation  
 416 vector is only mildly affected by the local reparameterization ( $\delta\mathbf{X}_{H_t}$ ,  $\Delta\mathbf{X}_{H_t}$ ) around the origin (i.e.,  
 417  $\mathbf{t}^* \approx \mathbf{t}$  for small motions). Likewise, small rotational motions that follow this fixed-point structure,  
 418 i.e., rotations about the x-axis, are only slightly perturbed by the mirroring operation.

## 419 B Reflection-Equivariant Diffusion Policy (MirrorDiffusion)

420 The general architecture of MirrorDiffusion follows the  $\text{SO}(2)$ -Equivariant Diffusion Policy pro-  
 421 posed by Wang et al. [7], with the key difference being a change in structural equivariance from  
 422 rotation to reflection. As illustrated in Fig. 8, the Equivariant ResNet used in [7] is modified to be  
 423 reflection-equivariant by constructing the ResNet architecture using the abstract group `Flip2dOnR2`  
 424 provided in the `E(n)-CNN` library [23]. The end-effector states are arranged following the repre-  
 425 sentation specified by the color coding in the figure. The reflection-equivariant linear layers are  
 426 implemented by overloading the Dihedral group in the `E(n)-CNN` library [23], with the number of  
 427 group elements set to 1 (a group only contains the original element and reflected counter part).

428 During the *encoding phase* (generating the global condition), two independent reflection-equivariant  
 429 ResNets encode the third-person and eye-in-hand views, each producing a pair of 128-dimensional  
 430 regular representations (i.e.,  $128 \times 2$ ). The robot states are arranged according to the corresponding  
 431 irregular and trivial group representations, as formulated in Eq. (3) and illustrated in Fig. 8. A  
 432 subsequent reflection-equivariant linear layer encodes these mixed representations into a  $128 \times 2$   
 433 regular representation. During the *denoising phase*, the noisy action is arranged according to the  
 434 representation specified in Fig. 8, and is processed by a reflection-equivariant linear layer, producing  
 435 a  $64 \times 2$  regular representation. A 1D Temporal U-Net with hidden dimensions [512, 1024, 2048]

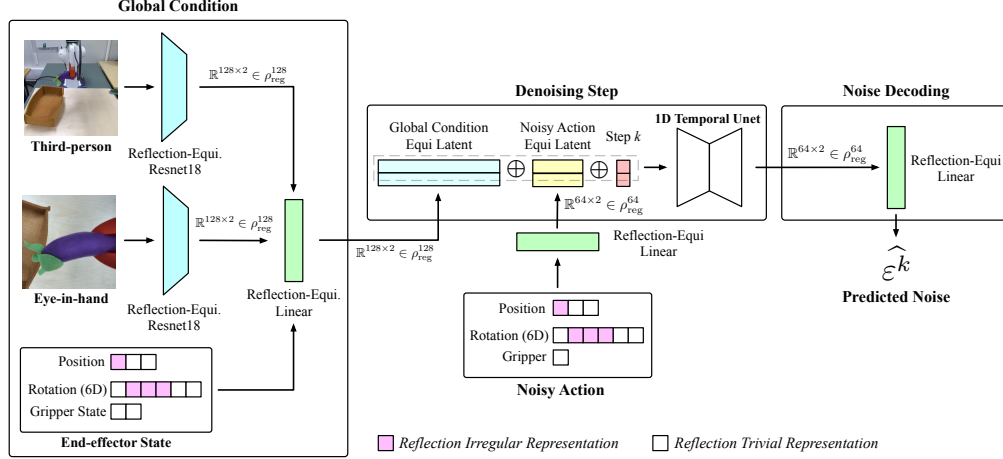


Figure 8: Illustration of Reflection Equivariant Diffusion (MirrorDiffusion) Network Architecture.

then processes each element of the embedding. Conceptually, this corresponds to applying the U-Net independently to the concatenated embedding (comprising the global condition and the action embedding) for both the original and reflected inputs, yielding a 64-dimensional embedding for each component. The separately denoised latents are recovered to shape  $64 \times 2$ , which is then passed through a final reflection-equivariant decoder to produce the predicted noise.

## C Experiment Implementation Details

### Baseline Networks

*Diffusion Policy* follows the hybrid-CNN architecture with global conditioning as described in [14], consistent with the baseline implementation in [7]. The input horizon, action horizon, and action prediction horizon are set to 2, 8, and 16, respectively. A fixed learning rate of  $1 \times 10^{-4}$  is used. The model utilizes a DDPM noise scheduler [24], with both training and inference configured for 100 diffusion steps across simulated and real-world experiments.

*BC-RNN* follows the network architecture and hyperparameters specified in RoboMimic [12]. Specifically, the image encoder comprises a ResNet18 [25] followed by a Spatial Softmax layer. The extracted image features are concatenated with the robot’s proprioceptive states and passed through a 2-layer LSTM, whose final hidden state is used as input to a Gaussian Mixture Model (GMM) policy head. As in RoboMimic [12], during rollout, the learned standard deviations of each GMM mode are clamped and replaced with a fixed value of  $1 \times 10^{-4}$ .

### Random Overlay

Random Overlay plays an integral role in MirrorDuo. For diffusion policies, we follow the default setting described in [29]. Specifically, for each batch of trajectories, we randomly sample half and overlay their images with random backgrounds using a blend factor  $\alpha = 0.5$ . During preliminary experiments, MirrorDiffusion exhibited minor performance degradation under stronger overlays (i.e., lower  $\alpha$ ). As a result, we set the blend factor to  $\alpha = 0.75$ . The blending operation is defined as:

$$\text{overlaid\_image} = \alpha \cdot \text{image} + (1 - \alpha) \cdot \text{random\_background}, \quad \alpha \in [0, 1].$$

The number of warmup epochs, during which the ratio of sampled trajectories gradually ramps up to the designated threshold, is set to  $\min(20, 4000/\text{num\_demos})$  to accommodate varying numbers of demonstrations.

## Training Epochs and Evaluation Protocols

The total number of training epochs is scaled according to the number of demonstrations, computed as  $50000/\text{num\_demos}$ . Evaluation is performed every  $2000/\text{num\_demos}$  steps. At each evaluation step, 50 rollouts are performed. For experiments involving additional demonstrations from mirrored arrangements, the number of demonstrations used to compute the total training epochs and evaluation frequency is fixed to the base number of demonstrations, i.e., 200.

For the data points of the SO(2)-Equivariant Diffusion Policy [7] in Table 2 and Table 6, results for tasks with 100 and 200 demonstrations are directly taken from the published results. Results for 50 and 500 demonstrations are not publicly available and are therefore newly generated using the authors’ released code.

**Image size.** The image inputs for all experiments are of size  $3 \times 84 \times 84$ , with a random crop of size  $76 \times 76$  applied during training. The crop is set to  $76 \times 76$  center crop during evaluation.

**Initial Pose.** The local reparameterization of poses and actions (Eq. (2)) require centering all trajectories around a fixed initial pose. In simulation, where the starting pose is constant, we use this fixed pose directly. In real-world experiments, where initial poses vary within a neighborhood, we use the average initial pose across demonstrations.

## Simulation Task Descriptions

In this work, we use five simulation tasks from MimicGen [17], using the provided datasets. All tasks employ the Franka Panda robot as the manipulator, operating in a 7-dimensional action space comprising 6 degrees of freedom for the end-effector pose and 1 dimension for gripper open/close. Each task uses two camera views: a third-person view and an eye-in-hand view. Task descriptions and key properties are summarized below:

- *Square D0*: Grasp the square nut by the handle and insert it into a matching square peg. The nut undergoes  $360^\circ$  random rotation around the z-axis, with limited positional variation. The target peg remains fixed.
- *Square D2*: Same objective as Square D0, but with a broader distribution over both the nut’s and peg’s positions and orientations.
- *Coffee D2*: Pick up the coffee pod from one side, insert it into the coffee machine on the opposite side, and close the lid. The coffee pod has constrained positional variation, and the coffee machine has limited variation in position and z-axis orientation.
- *Stack Three D1*: Sequentially stack three cubes on top of each other. Positions and z-axis orientations of the cubes are randomized within the workspace.
- *Three Piece Assembly D1*: Sequentially assemble three pieces, requiring stricter precision on orientation and placement.

Except for *Square D0* and *Coffee D2*, which involve constrained initialization, all other tasks allow full  $360^\circ$  rotation around the z-axis and broad position variation for all relevant objects.

## D Simulation Results with Standard Deviation

Table 6 presents the complete simulation results from Table 2, including standard deviations.

## E Mismatch Between Mirrored and Actual Demonstrations

As discussed in the limitations section and observed in Table 2, when the number of demonstrations increases to 500 for the *Square D2* and *Three Pieces Assembly D1* tasks, MirrorDuo exhibits a marginal decrease in performance. We hypothesize that this is due to the mirrored demonstrations increasing the level of multi-modality in the data, leading to a more fragmented decision boundary.

	Method	Stack Three (D1)			Square (D2)			3-Part Assembly (D2)		
		50	100	200	100	200	500	100	200	500
Delta	EquiDiff.	20.7±0.9	54.7±5.2	77.3±1.8	25.3±8.7	41.3±9.8	60.0±7.5	15.3±1.8	39.3±1.8	63.0±3.0
	MirrorDiff.	51.3±0.9	77.3±0.9	89.3±0.9	24.0±2.8	48.7±2.5	59.3±3.4	25.3±2.5	49.3±3.4	61.3±4.1
	DiffPo.	19.3±3.8	47.3±2.5	80.7±5.0	20.7±0.9	40.0±2.8	58.0±1.6	11.0±3.0	31.3±2.5	64.0±7.5
	DiffPo. + $\mathcal{M}$	50.7±1.9	68.0±3.3	91.3±0.9	32.7±2.5	49.3±8.4	56.7±2.5	21.0±1.0	47.3±6.6	61.3±5.0
	BC-RNN	0.7±0.9	2.0±0.0	8.0±1.6	4.0±1.6	9.3±2.5	32.0±4.3	0.0±0.0	0.0±0.0	6.0±0.0
	BC-RNN + $\mathcal{M}$	0.0±0.0	3.3±1.9	37.3±6.2	4.7±2.5	12.7±0.9	43.3±8.2	1.0±1.0	3.3±1.9	12.0±2.8
Relative	EquiDiff.	6.7±2.5	25.3±3.3	62.7±3.5	11.3±1.3	20.7±4.1	40.0±2.0	1.3±0.7	4.7±0.7	22.0±2.8
	MirrorDiff.	28.7±2.5	57.3±0.9	80.0±3.3	18.0±1.6	32.0±3.3	47.3±1.9	13.3±2.5	23.3±2.5	50.0±1.6
	DiffPo.	19.3±0.9	31.3±3.4	58.0±3.3	18.0±3.3	30.0±4.3	44.0±1.6	4.0±0.0	13.3±0.9	32.0±4.3
	DiffPo. + $\mathcal{M}$	29.3±3.4	50.0±2.8	68.0±0.0	32.7±2.5	41.3±3.4	45.3±5.7	11.0±1.0	26.7±5.7	48.0±1.6
	BC-RNN	6.7±4.1	18.0±1.6	51.3±11.1	8.0±1.6	19.3±2.5	45.3±3.8	2.0±0.0	3.3±1.9	11.3±5.7
	BC-RNN + $\mathcal{M}$	18.0±5.7	35.3±6.8	73.3±3.4	16.0±2.8	24.7±0.9	48.0±8.6	1.0±1.0	8.7±2.5	22.7±1.9

Table 6: **Setting III: Wide-view, two-sided demonstrations.** Success rate (%) on three MimicGen tasks as the number of demonstrations increases. *Delta* and *Relative* refer to action controllers. Results are averaged over the top-1 rollout (50 trials) from three training seeds. EquiDiff denotes the SO(2)-equivariant diffusion policy [7], DiffPo. denotes the diffusion policy [14],  $\mathcal{M}$  denotes MirrorDuo augmentation, and MirrorDiff. denotes the proposed mirror-equivariant diffusion policy.

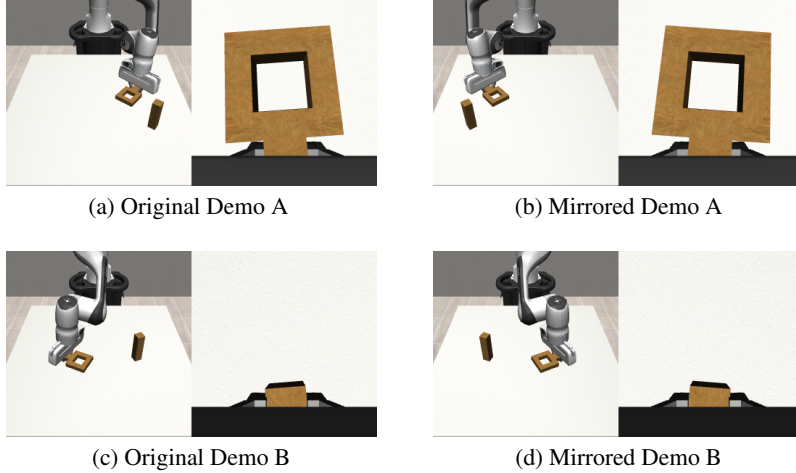


Figure 9: **Illustration of Conflicting Visual Cues and Trajectories** introduced by mirrored demonstrations in the *Square D2* task. Each mirrored demonstration features an approximately co-located square nut relative to its original counterpart (e.g., Fig.(b, c) and Fig.(d, a)), yet exhibits a distinct eye-in-hand view. This discrepancy suggests that while the mirrored and original demonstrations share a similar initial setup (i.e., the first subtask), they require oppositely rotating actions.

Specifically, for a given original demonstration, its mirrored counterpart may represent a valid but conflicting trajectory from the actual sample contained in the original dataset. For *Square D2*, as illustrated in Fig. 9, the mirrored demonstration in Fig. 9b shows the square nut approximately co-located with that in the other original demonstration (Fig. 9c). However, the mirrored eye-in-hand view (Fig. 9b) shows the entire square nut clearly, while in the original view, only the handle of the nut is visible. Fig. 10 shows one of the examples in the *Three-piece Assembly D1* Task. Each mirrored demonstration features an approximately co-located T-shaped piece relative to its original counterpart (e.g., Fig.(b, c) and Fig.(d, a)), yet exhibits a distinct eye-in-hand view, one oriented toward the workspace, the other facing outward.

These examples indicate that, under similar subtask setups, the augmented data introduces additional valid trajectories that involve opposing end-effector rotations and result in distinct eye-in-hand views. This divergence introduces ambiguity into the learned policy, and the likelihood of such ambiguity increases as the density of demonstrations grows.



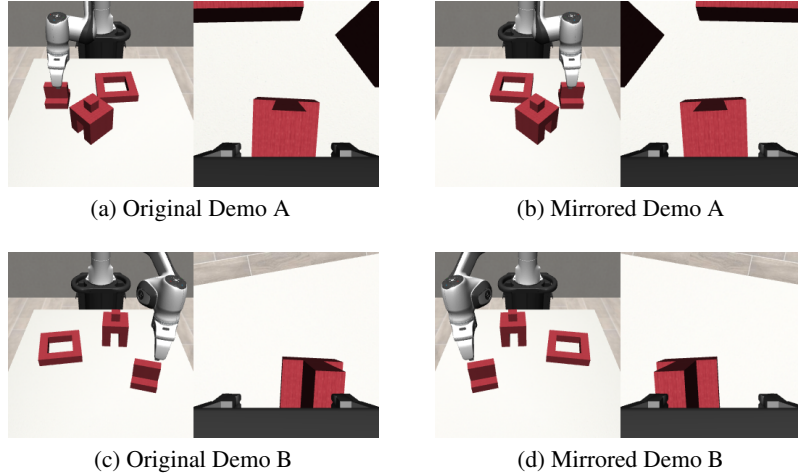


Figure 10: **Illustration of Conflicting Visual Cues and Trajectories** introduced by mirrored demonstrations in the *Three Piece Assembly D1* task. Each mirrored demonstration features an approximately co-located T-shaped piece relative to its original counterpart (e.g., (b, c) and (d, a)), yet exhibits a distinct eye-in-hand view, one oriented toward the workspace, the other facing outward. This discrepancy suggests that while the mirrored and original demonstrations share a similar initial setup (i.e., the first subtask), they require oppositely rotating actions.

## 519 **F MirrorDuo with Off-Centered Third-Person Camera**

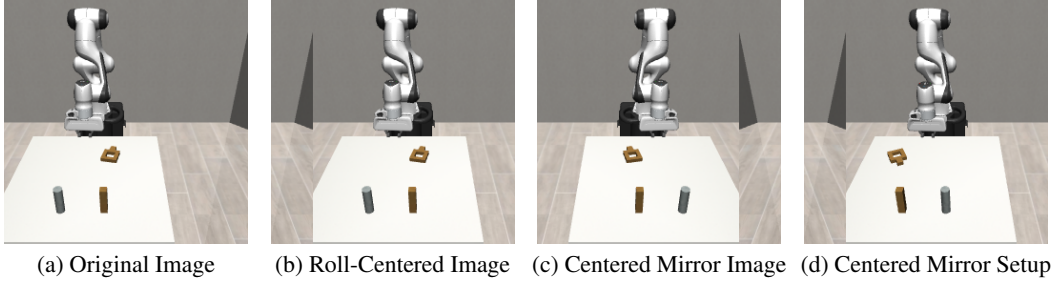


Figure 11: **Illustration of Mirroring with an Off-Centered Camera.** (a) Original image from an off-centered third-person camera. (b) Roll-centered image with the end-effector aligned to the mirroring axis. (c) Mirrored version of the roll-centered image used by MirrorDuo. (d) Roll-centered image from the actual mirrored setup.

520 To transfer the mirrored skill to the initial configuration based on the given demonstrations, Mirror-  
 521 Duo requires alignment not only in the proprioceptive states but also in the image space. This means  
 522 that the end-effector should be near the mirroring axis, i.e. the horizontal center of the image. Al-  
 523 though random cropping alleviates the strictness of centering to the midline, a general alignment is  
 524 still required. For off-centered third-person cameras, a pre-alignment step is necessary. Otherwise,  
 525 even if MirrorDuo successfully learns the mirrored skill, the starting configuration and workspace  
 526 setup of the mirrored demonstration will not align with the ideal scenario of transferring under the  
 527 current robot configuration to a mirrored object arrangement.

528 Here we show that MirrorDuo can be applied to off-centered third-person camera scenarios by pre-  
 529 centering the view, demonstrated in two settings: one with one-sided demonstrations (*Square D0*,  
 530 Fig.11) and another with two-sided demonstrations (*Stack Three D1*, Fig.12). Let the off-centered  
 531 camera be denoted as  $\{C\}$  and the re-centered camera as  $\{C_{\text{ref}}\}$ . The mirrored setup is derived using  
 532 Eq. (1), where the mirroring is applied with respect to the re-centered camera pose, i.e.,  $\mathbf{X}_{C_{\text{ref}}}$ .

	In-domain	# M-Demos		
		0	5	10
MirrorDiff.	<b>89.3</b> $\pm$ 1.9	0.0 $\pm$ 0.0	<b>72.7</b> $\pm$ 1.9	<b>90.0</b> $\pm$ 1.6
DiffPo. + $\mathcal{M}$	83.3 $\pm$ 1.9	0.0 $\pm$ 0.0	69.3 $\pm$ 0.9	84.0 $\pm$ 1.6
DiffPo.	85.3 $\pm$ 2.5	0.0 $\pm$ 0.0	23.3 $\pm$ 1.9	32.7 $\pm$ 3.8

Table 7: **Off-Centered Third-Person Camera, One-Sided.** Success rate (%) on the *Square D0* task under the *mirrored arrangement*, with an additional 5 or 10 demonstrations from the mirrored setup (denoted as M-Demos in this table) added on top of the original 200 demonstrations. Each data point reports the average of the top-3 evaluations, with 50 rollouts per evaluation.

	Third-person Camera	
	Centered	Off-centered
MirrorDiff.	89.3 $\pm$ 0.9	<b>84.7</b> $\pm$ 1.9
DiffPo. + $\mathcal{M}$	<b>91.3</b> $\pm$ 0.9	81.3 $\pm$ 1.9
DiffPo.	80.7 $\pm$ 5.0	70.7 $\pm$ 3.4

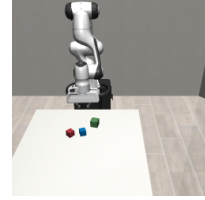


Table 8: **Off-Centered Third-Person Camera, Two-Sided.** Success rates (%) on the *Stack Three D1* task.

Figure 12: Illustration of Off-centered camera view for *Stack Three D1*

Following previous setups, we evaluate MirrorDiffusion, Diffusion + MirrorAug, and the Diffusion baseline using re-rendered demonstrations under off-centered cameras. In the one-sided case, we assess performance on mirrored arrangements with 0, 5, and 10 additional demonstrations. For the two-sided case, we directly evaluate in-domain performance. All experiments assume access to global camera extrinsics, enabling roll-centering by aligning the initial end-effector pose to the image center. The same offset is applied to subsequent frames, with the rolled region tinted green to indicate shifted areas. Networks receive these centered images as input.

In the off-centered camera settings, the visual domain gap between the (already-centered) mirrored and original samples arises not only from the robot’s asymmetry but also from perspective shifts across the left and right sides of the workspace. For instance, as shown in Fig. 11, in the original domain, the square peg appears near the horizontal center of the image, showing only its front face. In the mirrored setup, however, the peg shifts toward the left side of the image, revealing its right face, an angle not observed in the original demonstrations. Additionally, the appearance of the table also changes under this off-centered view.

Table 7 shows that the widened visual domain gap reduces the performance of direct transfer to the mirrored arrangement to zero, in contrast to the matched in-domain performance of Diffusion + MirrorAug when the third-person camera is centered (Fig. 4). However, the data efficiency benefit of MirrorDuo remains evident. Under the increased visual domain gap, the performance in the mirrored arrangement with five additional demonstrations is only 17% and 24% lower than the corresponding in-domain setting for MirrorDiffusion and Diffusion + MirrorAug, respectively. With ten additional demonstrations, both methods recover their performance in the mirrored setup, matching their in-domain success rates. In contrast, without MirrorDuo, simply adding ten demonstrations in the mirrored setup results in only a 32.7% success rate for the baseline diffusion policy.

For the two-sided setup with an off-centered camera, we evaluate on *Stack Three D1*. The centered camera results reported in Table 8, are drawn from Table 6 (averaged over three seeds). For the off-centered case, each entry reflects the average of the top 3 evaluations, with 50 rollouts per evaluation due to limited compute resources. As shown in Table 8, all methods exhibit a performance drop under the more challenging viewpoint. MirrorDiffusion and Diffusion Policy + MirrorAug maintain relatively high success rates (84.7% and 81.3%, respectively), while the baseline drops to 70.7%. This  $\sim 10\%$  decline aligns with trends observed in the centered-camera setting, highlighting the effectiveness of mirroring-based approaches to off-centered camera variations.