Bridging Handheld and Teleoperated Supervision for Contact-Rich Manipulation via State-Gated Experts

Vidullan Surendran[1], **Neehar Peri**[1,2], **David Watkins**[1]

Abstract: Handheld data collection systems such as the Universal Manipulation Interface (UMI) enable scalable data collection across environments but capture only *observed* actions rather than the *desired* actions sent to a robot controller. Teleoperation captures desired actions directly but is slow to collect. We revisit this trade-off through the lens of *action validity* across task phases; handheld trajectories provide valid supervision in tolerant, free-space phases, but lack dynamic feasibility in contact-sensitive phases, where tracking observed trajectories at high stiffness produces large, unsafe contact forces. Rather than teleoperating entire tasks, we collect a small number of targeted teleoperated demonstrations only for segments where handheld policies fail. However, naively mixing the two data types performs worse than handheld data alone. We therefore propose Bi-modal Routing for Imitation Data via Gated Experts (BRIDGE), a mixture of diffusion policy experts that routes between specialist heads conditioned on the current robot state. BRIDGE improves success rates over handheld-only baselines by up to 36.7% across three contact-rich tasks. See our project page for videos.

Keywords: Mixture-of-Experts Diffusion Policy, Universal Manipulation Interface, Heterogeneous Data Collection, Contact-Rich Manipulation

1 Introduction

The Universal Manipulation Interface (UMI) [1] enables scalable, in-the-wild data collection by decoupling collection from robot hardware. This portability introduces an embodiment gap at execution time, particularly for tasks involving contact forces, friction, and compliance. Teleoperation systems such as GELLO [2] and ALOHA [3] provide high-fidelity on-robot data, but sacrifice collection speed and diversity.

We examine this trade-off through *action validity*. An *observed* action is the physically realized state of the end-effector (or handheld device), while a *desired* action is the reference trajectory sent to the controller. Due to finite stiffness, filtering, and contact, the two often differ. Our key insight is that handheld data (observed actions only) is valid supervision in tolerant free-space motion but lacks dynamic feasibility in contact-sensitive phases, where tracking observed actions at high stiffness generates large contact forces (Figure 1). We show that augmenting handheld data with targeted teleoperated demonstrations (which capture both action types) improves success in contact-sensitive phases without substantially increasing collection effort.

Naively combining the two data types degrades performance because observed and desired actions are incompatible supervision. We propose BRIDGE, a mixture of diffusion policy experts that routes between specialized heads conditioned on robot state. Cheap handheld data learns the task scaffold, while teleoperation is reserved for contact-rich bottlenecks.

Contributions. (1) We formalize *action validity*, showing observed actions suffice for free-space motion but fail in force-sensitive phases. (2) We introduce BRIDGE, a mixture of diffusion experts that jointly learns from data with observed and desired action labels. (3) We show BRIDGE improves task success over handheld-only baselines by up to 36.7% across three contact-rich tasks.

[1] RAI Institute, [2] Carnegie Mellon University

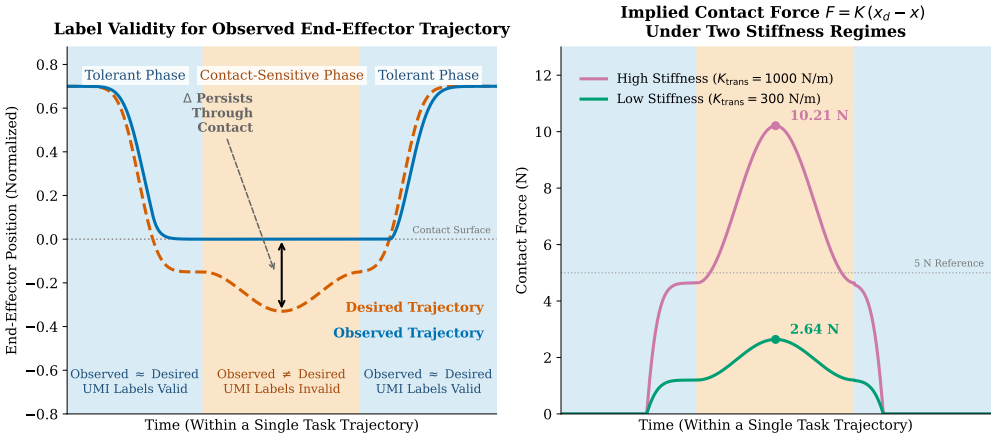


Figure 1: **Action Validity Under Contact.** NIST pulley routing [4]. *Left:* In tolerant phases (blue) the observed action approximates the desired action. In contact-sensitive phases (yellow) the desired action drives below the surface, and the persistent error Δ makes the observed trajectory an invalid proxy for intent. *Right:* Proportional control converts Δ into contact force via $F \propto K(x_d - x)$. High stiffness tracks closely but yields large peak forces; low stiffness reduces force but cannot track through contact.

2 BRIDGE: Gated Mixture of Experts

DM-UMI and Data Collection. UMI [1] relies on offline monocular SLAM and ArUco tags, but teleoperation requires online pose and gripper estimates. We therefore build Dual-Mode UMI (DM-UMI), which uses an XVISIO DS80 SLAM camera for online 6-DoF pose and magnetic sensors for gripper width, supporting both handheld and teleoperated collection. We first collect handheld demonstrations across diverse environments (the `base` dataset) and train a `base` policy to identify failure modes. We then collect targeted teleoperated demonstrations (the `support` dataset) capturing embodiment-specific effects such as controller dynamics, contact, and grasp stability. `Base` data (observed actions) provides broad coverage; `support` data (desired actions) isolates regimes where the `base` policy is unreliable.

Stage 1: `Base` Policy. Our base policy extends Diffusion Policy [6], denoising future action chunks conditioned on visual and proprioceptive observations. Rather than keeping only the ViT CLS token, which compresses spatial structure early, we retain all DINOv2 [5] patch tokens $V \in \mathbb{R}^{B\times N\times D}$ (1369 tokens for a 518×518 image) and reduce them with a PerceiverIO-style [7] aggregation using $M \ll N$ learnable queries Q: $Z_{vision} = \text{XATTN}(Q, V)$. State inputs (end-effector pose, gripper width) are projected to $Z_{state} \in \mathbb{R}^D$ and cross-attend to the vision tokens, giving

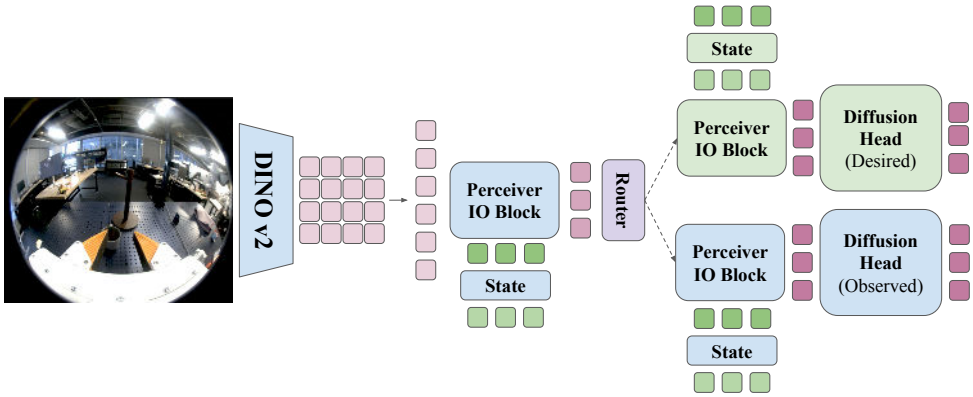


Figure 2: **BRIDGE Architecture and Training.** DINOv2 [5] features pass through a Perceiver IO block and are fused with state via cross-attention. A state-conditioned router hard-switches between the observed (`base`) and desired (`support`) diffusion heads. We sequentially train the `base` model (blue), collect targeted teleoperated demonstrations to train the `support` head (green), and train the router (purple).

$Z_{latent} = \text{XATTN}(Z_{state}, Z_{vision})$. We call these modules the latent adapter ϕ_b. Z_{latent} conditions a temporal diffusion policy trained with the standard diffusion loss on the `base` dataset.

Stage 2: `Support` Head. With the `base` expert π_b and shared vision encoder frozen, we train a `support` latent adapter ϕ_s and head π_s on the `support` dataset, whose targets are desired trajectories: $\hat{a}^s_{t:t+H} = \pi_s(\phi_s(Z_{latent}))$, using the standard diffusion loss.

Stage 3: Router. We train an MLP gate $G_\psi(z)$ on the intermediate Z_{latent} of π_b for samples from both datasets. To avoid ambiguous labels where the datasets overlap, we remove `base` latents within an ϵ-neighborhood of `support` latents. Given the filtered `base` bank B_b and `support` bank B_s, we compute average cosine similarity to the $k = 16$ nearest neighbors in each:

$$\sigma_+(z) = \frac{1}{k} \sum_{z_s \in \mathcal{N}_k(z, B_s)} \cos(z, z_s), \quad \sigma_-(z) = \frac{1}{k} \sum_{z_b \in \mathcal{N}_k(z, B_b)} \cos(z, z_b), \tag{1}$$

and assign the label $\rho(z) = \mathbb{I}[\sigma_+(z) - \sigma_-(z) > \eta_\rho]$. We distill this k-NN classifier into G_ψ using binary cross-entropy.

Inference. Each observation is encoded into shared visual and state features, and the adapters produce expert-specific latents z^b_t, z^s_t. The router predicts support probabilities $g_{t:t+H} = G_\psi(z^b_t)$, and given a threshold η we hard switch:

$$m_{t+j} = \mathbb{I}[g_{t+j} > \eta], \qquad \hat{a}_{t+j} = (1 - m_{t+j})\, \hat{a}^b_{t+j} + m_{t+j}\, \hat{a}^s_{t+j}. \tag{2}$$

Unlike residual policies, BRIDGE does not require the `support` expert to correct the full `base` distribution; it treats observed and desired actions as distinct supervision types and routes to the `support` expert when in its action manifold.

3 Experiments

We evaluate on a Franka FR3 7-DoF arm with a Cartesian impedance controller on three contact-rich tasks (Figure 3). All methods share the architecture above and differ only in supervision and training schedule: (i) *`Base` (Handheld)*, trained on handheld observed actions; (ii) *`Base` (Teleoperated)*, trained on full-task teleoperated desired actions as an upper bound; (iii) *Naive Mix*, trained on combined handheld and `support` data; and (iv) *BRIDGE*.

Tasks. *NIST Pulley Routing*: route a deformable O-ring around pulleys (NIST Task Board #2 [4]) while maintaining tension. *Pipe Insertion*: grasp a pipe and insert it into a tight-tolerance opening, requiring 6-DoF alignment and force-aware insertion. *Battery Insertion*: insert an AA battery into a spring-loaded compartment while keeping the spring compressed.

Data. Table 1 summarizes demonstrations. We collect handheld data until the `base` policy converges in free-space phases. Although `support` data covers a large fraction of trajectory time (due to slower collection), it is spatially sparse (3–15% of path length). The full-task teleoperation dataset used approximately the same wall-clock budget as `base+support`, and was considerably more cognitively demanding to collect.

Results. Table 2 reports task success. Pulley routing uses 25 trials over five rig positions (± 2 cm); pipe and battery insertion use 30 trials over three positions (rig shifted ± 2 cm and ± 1 cm respectively).

Table 1: **Dataset Statistics.** Support Temporal % and Dist. % are the median fractions of time and end-effector path length covered by a `support` demonstration. †Full-task teleoperation under an approximately time-matched budget.

Task	`Base` #	`Support` #	`Teleop`† #	`Support` Temporal %	`Support` Dist. %
NIST Pulley Routing	201	50	60	37.8%	7.4%
Pipe Insertion	100	81	60	24.5%	3.4%
Battery Insertion	290	59	100	61.1%	14.9%

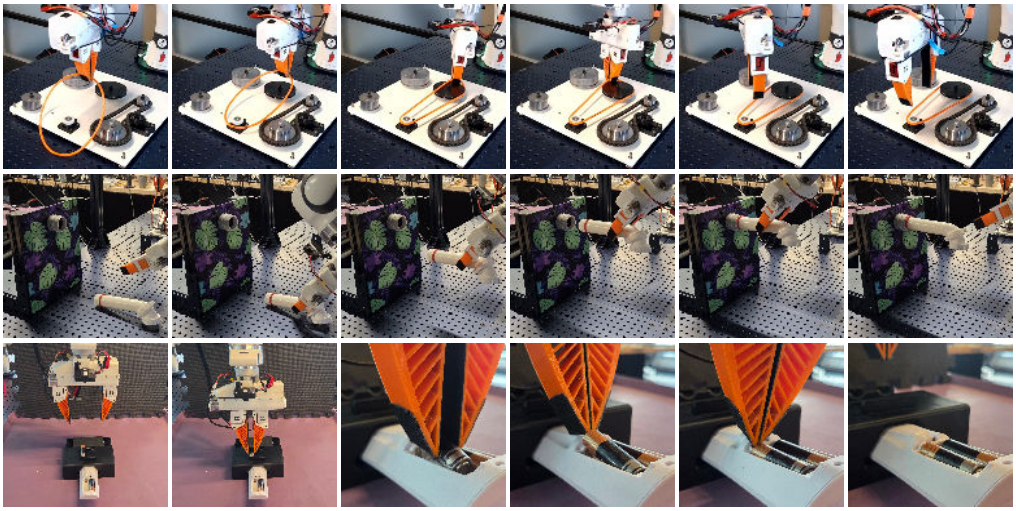


Figure 3: **Policy Rollouts.** NIST pulley routing (*top*), pipe insertion (*middle*), and spring-loaded battery insertion (*bottom*).

Table 2: **Task Success Rate (%).** BRIDGE substantially improves over handheld-only training and approaches the teleoperated upper bound. Naive Mix performs worst on all tasks.

Method	NIST Pulley	Pipe Insertion	Battery Insertion	Avg.
`Base` Policy (Handheld)	44.0	13.3	10.0	22.4
Naive Mix	0.0	6.7	0.0	2.2
BRIDGE (Ours)	76.0	50.0	33.3	53.1
`Base` Policy (Teleoperated, Upper Bound)	84.0	63.3	40.0	62.4

Pulley Routing. The handheld policy (44.0%) drops sharply in the top and bottom rig positions, where high-stiffness tracking frequently triggered controller faults. Naive Mix fails entirely: mixing observed and desired labels yields imprecise manipulation, with the policy touching the top of the pulley and missing the groove. BRIDGE routes to the `support` expert for loaded routing, executing that segment at lower stiffness and reaching 76.0% versus 84.0% for full teleoperation.

Pipe Insertion. The handheld policy usually aligns the pipe but fails at insertion due to tracking errors near contact (13.3%). Naive Mix is worse (6.7%), suggesting that even at high stiffness, combining observed and desired actions in one policy yields inconsistent alignment and contact behavior. BRIDGE commands desired actions during final alignment and insertion, quadrupling success to 50.0% (teleoperated: 63.3%).

Battery Insertion. The handheld policy transports well in free space but struggles against the spring (10.0%). Naive Mix picks up the battery roughly 40% of the time but exhibits a systematic rightward lateral bias during transport, so it never inserts; this shows that mixing labels can distort the action distribution even before contact. BRIDGE preserves free-space competence while correcting insertion, more than tripling success to 33.3% and raising the partial-credit score from 11.25 to 19.75 (of 30). The remaining gap to teleoperation (40.0%) suggests that regulating final contact forces remains the main challenge.

4 Conclusion

We address the embodiment gap between handheld collection and teleoperation through *action validity*: handheld data (observed actions) is valid supervision in error-tolerant phases but fails in contact-sensitive phases requiring dynamically feasible desired actions. BRIDGE uses scalable handheld data for the task scaffold and sparse teleoperated data for contact bottlenecks. Across tasks, this sparse data covers less than 15% of end-effector path length, yet BRIDGE recovers up to 79% of the gap to the full teleoperation upper bound.

References

[1] C. Chi, Z. Xu, C. Pan, E. Cousineau, B. Burchfiel, S. Feng, R. Tedrake, and S. Song. Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots. *arXiv preprint arXiv:2402.10329*, 2024.

[2] P. Wu, Y. Shentu, Z. Yi, X. Lin, and P. Abbeel. Gello: A general, low-cost, and intuitive teleoperation framework for robot manipulators. *arXiv preprint arXiv:2309.13037*, 2023.

[3] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

[4] National Institute of Standards and Technology. Assembly performance metrics and test methods. https://www.nist.gov/el/intelligent-systems-division-73500/robotic-grasping-and-manipulation-assembly/assembly, 2026. Accessed: 2026-05-19.

[5] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[6] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

[7] A. Jaegle, S. Borgeaud, J.-B. Alayrac, C. Doersch, C. Ionescu, D. Ding, S. Koppula, D. Zoran, A. Brock, E. Shelhamer, et al. Perceiver io: A general architecture for structured inputs & outputs. *arXiv preprint arXiv:2107.14795*, 2021.