Human2Any: Human-to-Robot Transfer via Constraint-Aware Compositional Planning

Author Names Omitted for Anonymous Review. Paper-ID [add your ID here]

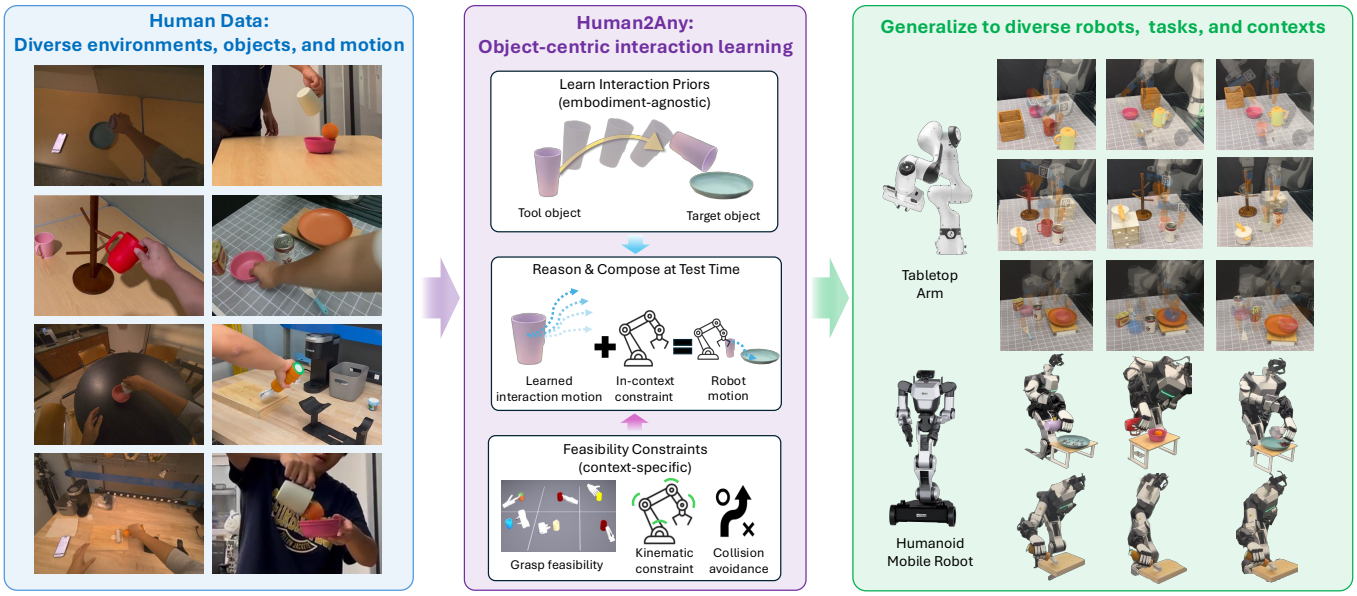


Fig. 1: **Human-to-robot transfer with Human2Any.** Human2Any learns object-centric interaction priors from human videos and adapts them to diverse robot embodiments, tasks, and contexts.

***Abstract*—Human videos are a scalable source of supervision for robot manipulation, as they are abundant and naturally capture rich object interactions. However, transferring human demonstrations to robots remains challenging due to embodiment mismatch, scene variation, and robot-specific feasibility constraints. We present Human2Any, a framework for learning reusable object-centric interaction priors from human videos without requiring real-world robot demonstrations in the target task contexts. Human2Any represents manipulation through object–object interaction motion, capturing task-relevant scene changes while abstracting away embodiment-specific details. It composes learned interaction priors with robot-side feasibility reasoning and motion planning, allowing the same human-derived knowledge to adapt to different embodiments, scene geometries, and task contexts. We validate Human2Any across diverse manipulation settings, including real-world experiments on a Franka tabletop setup and an RBY-1 humanoid mobile robot, demonstrating robust interaction-centric manipulation without real-world robot training data. Project website: https://human2any.github.io/**

I. Introduction

Learning manipulation systems that can generate feasible and purposeful motions for solving everyday tasks remains a fundamental yet challenging problem [19, 36, 77, 62]. While behavior cloning (BC) has achieved remarkable progress in recent years [6, 42, 3, 21, 75], its generalization to diverse environments—such as varying object geometries, poses, and scene layouts—heavily depends on both the quality and quantity of training data [42, 54]. Unfortunately, the collection of high-quality, embodiment-aligned demonstrations, typically obtained through robot teleoperation [10, 31], is time-consuming and costly, which significantly limits scalability.

In contrast, *learning from human videos* has emerged as a promising paradigm for teaching robots at scale [29, 72, 2]. Unlike robot demonstrations, human videos are abundant and diverse, but their most transferable signal is not the human body motion itself; rather, it is the *object-centric interaction structure* underlying the task. Human videos capture rich manipulation strategies [26, 1]: they show how objects change state *because of* purposeful interactions—reach, grasp, insert, pour—across long horizons and varied objects, layouts, and styles. This diversity is ideal for learning *compositional* skills, but directly imitating human motion is mismatched to robot embodiment. Trajectory-level reproducibility is inherently context-dependent: a motion that works in one scene may be infeasible on the robot due to reachability, joint limits, or collisions. As the layout changes, previously infeasible motions may become feasible, and vice versa. Thus, the key

is not to copy a particular trajectory, but to extract structure that remains stable across embodiments and contexts.

This raises a central challenge: *how can we harness the breadth of human videos to learn transferable interaction models that can be adapted to robot-specific constraints?* We tackle this problem through the lens of **object-centric motion generation**. Our key insight is that what transfers is the *interaction structure*: which objects participate, when contacts form and break, and how object states evolve. Accordingly, we represent the manipulation process as a factor graph that explicitly disentangles *agent–object* and *object–object* interactions (Fig. 1). The object–object factors capture agent-agnostic motion distributions learned from human videos, while the agent–object factors and feasibility terms account for embodiment-specific constraints such as grasping, reachability, and collision avoidance. At test time, we sample and refine graph variables under the robot's in-context constraints to produce robot-feasible interaction sequences that preserve the goal-achieving structure observed in video. This formulation scales the acquisition of diverse object-centric motion models while retaining structure-aware adaptability across embodiments and task scenarios, unifying compositional reasoning and constraint satisfaction within a sampling-and-scoring procedure.

Our main contributions are summarized as follows:

- We formulate manipulation as an *object-centric factor graph* that separates human-transferable *object–object* interaction factors from robot-specific *agent–object* feasibility factors, enabling motion learning from human videos with test-time robot adaptation.
- We represent skills as compositions of object-centric motion models and develop a foundation-model-based pipeline [53, 70, 68] to extract object motions from unstructured human videos.
- We introduce a constraint-aware sampler that scores and resamples candidate motion compositions under in-context constraints to produce robot-feasible motions.
- We validate the approach across multiple robot embodiments and long-horizon tasks, demonstrating generalization, cross-embodiment adaptability, and planning efficiency.

II. Related Work

Human-video-based manipulation learning. Human videos provide a scalable source of supervision for robot manipulation because they capture diverse object interactions without requiring robot teleoperation. Prior work learns from human demonstrations through visual imitation, retargeting, affordance prediction, or intermediate object-centric signals such as 2D/3D object flow and point tracks [72, 2, 74, 48, 37, 56, 29, 64, 1]. Other works co-train robot policies with human videos and robot data, often using human videos to improve visual representations or rewards [50, 47, 52, 40]. Retargeting-based methods explicitly address embodiment mismatch by mapping human motion to robot actions, while flow- or trajectory-conditioned methods use object motion as a transfer interface. However, these methods typically predict robot actions directly or rely on learned translation policies, making it difficult to enforce robot-specific feasibility, scene geometry, and contact constraints at deployment. In contrast, Human2Any extracts embodiment-agnostic object-object motion from human videos and resolves embodiment- and scene-specific constraints at test time.

Object-centric and interaction-centric manipulation. Object-centric representations have been widely used to improve manipulation generalization across object instances, configurations, and scenes. Prior methods learn affordances, grasping strategies, object flows, motion fields, or post-grasp object trajectories to guide manipulation [71, 17, 65, 16, 59, 14, 27, 55, 76, 35, 73, 61, 34]. These approaches often decompose manipulation into grasp selection and motion generation, but many follow a cascaded workflow in which a grasp or intermediate motion representation is chosen before checking downstream feasibility. Recent work jointly optimizes grasp and trajectory from video demonstrations [11] or decomposes manipulation into alignment and interaction stages using robot-specific data [13]. Human2Any instead learns object–object interaction motion from human videos and composes it with robot-specific grasping, feasibility checking, and motion planning, enabling test-time adaptation to the current embodiment and scene context.

Compositional planning and guided sampling. Task-and-motion planning decomposes long-horizon manipulation into symbolic decisions and continuous motions [28, 23, 22]. Learning has been integrated into TAMP through learned skills, samplers, or constraints to improve scalability and generalization [4, 57, 43, 5, 41, 7, 8, 66, 49, 18, 32]. Meanwhile, diffusion models have become effective trajectory generators for robot learning and long-horizon composition [25, 58, 60, 6, 75, 44, 45, 39, 38, 9]. However, independently sampled motion components may be individually plausible but jointly infeasible under grasp, kinematic, collision, and scene constraints. Human2Any addresses this through constraint-aware test-time steering: sampled interaction and grasp motions are assembled into robot trajectories, scored with in-context feasibility checks, and progressively resampled toward executable compositions.

III. Learning Interaction Priors from Video

Human videos provide rich object-interaction demonstrations, but their motions are not directly executable by robots due to embodiment mismatch and scene-dependent constraints. We treat each video as a tool object interacting with a target object, e.g., a cup tilting toward a bowl or a utensil being placed on a bowl, and extract object geometry and relative object motion while discarding human-specific arm and hand motion. As shown in Fig. 2, Human2Any learns object–object interaction priors from human videos and robot-specific agent–object priors from embodiment-specific grasp data, then composes them with robot- and scene-specific feasibility reasoning at deployment.

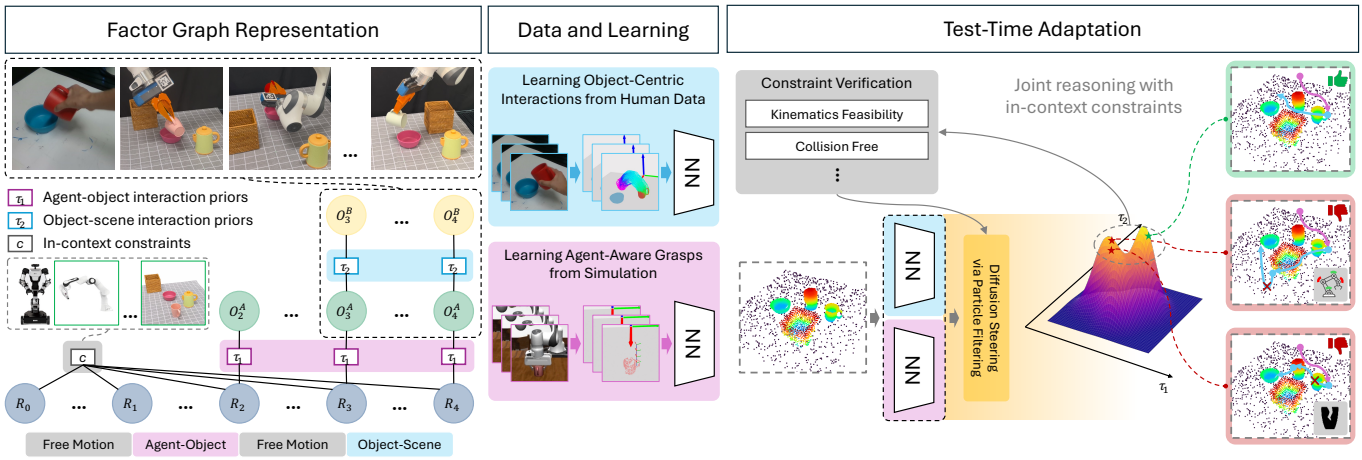


Fig. 2: **Human2Any overview.** Given human demonstration videos, Human2Any extracts object-centric interaction motion between tool and target objects and learns object–object interaction priors that are independent of the human embodiment. Separately, robot-specific agent–object priors model how a particular robot embodiment grasps and controls tool objects. At deployment, these learned priors are composed under the current robot, scene, and task constraints to generate executable manipulation trajectories.

Interaction-Centric Task Representation. We consider tasks specified by a high-level skeleton $\mathcal{S} = [s^1, \ldots, s^K]$ [23], where each phase $s^k = \langle a^k, \mathcal{O}^k \rangle$ denotes an interaction type a^k and the involved objects $\mathcal{O}^k$. There are three interaction types: free-space motion, agent–object interaction, and object–object interaction. Free-space motion connects interaction phases; agent–object interaction establishes robot-specific contact with a tool object; and object–object interaction describes how a tool object moves relative to a target object.

The key transferable representation is the object–object interaction trajectory $\tau^{\text{rel}} = \{\mathbf{T}^{\text{rel}}(h)\}_{h=0}^{H-1}$, where $\mathbf{T}^{\text{rel}}(h) \in \text{SE}(3)$ denotes the tool-object pose relative to the target object at time h. This representation captures task-relevant interaction motion while removing the human embodiment from the supervision signal. In this work, we focus on prehensile manipulation, where the robot maintains a rigid attachment to the tool object during interaction. This allows robot execution to be recovered from object–object interaction motion and a grasp pose.

Object–Object Priors from Human Videos. For each processed demonstration, we assume access to the tool and target object point clouds $\mathbf{P}^A, \mathbf{P}^B \in \mathbb{R}^{N\times 3}$ and the corresponding object-centric relative trajectory $\tau^{\text{rel}} = \{\mathbf{T}_h^{\text{rel}}\}_{h=0}^{H-1}$. In our implementation, these are estimated from RGB-D videos using off-the-shelf segmentation and tracking tools [53, 70]; details are provided in the appendix. Given tracked 3D keypoints $\xi^{3d} \in \mathbb{R}^{H\times M\times 3}$, we estimate τ^{rel} using the Kabsch algorithm [33] with RANSAC [20]. We trim each trajectory to frames near the interaction segment using a task-dependent distance threshold from the final interaction pose, approximately 0.15–0.2 m across tasks. This yields $\mathcal{D}_{\text{os}} = \{(\mathbf{P}_i^A, \mathbf{P}_i^B, \tau_i^{\text{rel}})\}_{i=1}^{|\mathcal{D}_{\text{os}}|}$ for training object–object interaction samplers.

Robot-Specific Agent–Object Priors. Object–object priors specify how objects should interact, but the robot must still choose an embodiment-specific way to grasp and control the tool object. We therefore train an agent–object prior for each robot embodiment using simulated grasp data. Each sample contains a tool-object point cloud $\mathbf{P}^A$ and a robot end-effector trajectory τ^e for reaching and grasping the object, forming $\mathcal{D}_{\text{ro}} = \{(\mathbf{P}_i^A, \tau_i^e)\}_{i=1}^{|\mathcal{D}_{\text{ro}}|}$. The resulting prior $p_{\theta_{\text{ro}}}(\tau^e \mid \mathbf{P}^A)$ captures embodiment-specific grasping behavior while remaining separate from the object–object interaction prior. This separation allows object-level interaction knowledge learned from human videos to be reused across different robot embodiments, while robot-specific execution is handled through the agent–object prior and test-time feasibility reasoning.

Diffusion Trajectory Prior Training. We instantiate object–object and agent–object priors as conditional diffusion models over pose trajectories. Given a clean trajectory $\mathbf{x}_0$, the forward process generates noisy samples $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The model predicts the injected noise by minimizing $\mathcal{L}(\theta) = \mathbb{E}_{t,\mathbf{x}_0,\boldsymbol{\epsilon}}[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, \tilde{c})\|_2^2]$, where $\tilde{c}$ denotes the corresponding point-cloud conditioning input. For object–object priors, $\tilde{c} = (\mathbf{P}^A, \mathbf{P}^B)$; for agent–object priors, $\tilde{c} = \mathbf{P}^A$. We apply rigid transformations to both point clouds and trajectories during training to improve robustness to object poses and scene layouts. Additional details are provided in the Appendix.

IV. Constraint-Aware Composition

At deployment, the robot observes a context c containing the robot model, scene point cloud, and task skeleton. Each learned prior can generate diverse candidate motion segments, but independently plausible segments may not compose into an executable robot trajectory. We therefore formulate test-time execution as constraint-aware composition: jointly selecting motion segments that remain likely under the learned priors

while satisfying robot- and scene-specific constraints.

A. Test-Time Context and Trajectory Assembly

Let $\mathbf{x}^{1:K} = \{\mathbf{x}^1, \dots, \mathbf{x}^K\}$ denote sampled motion segments for the K phases in the task skeleton. We define an assembly operator $\Gamma(\mathbf{x}^{1:K}, c)$ that converts these segments into a full robot trajectory and evaluates feasibility under the current context. For agent–object segments, Γ directly uses the sampled end-effector trajectory. For object–object segments, Γ converts the sampled relative object trajectory into end-effector targets using the grasp transform provided by the preceding agent–object segment. Under the rigid grasp assumption, this conversion is $\mathbf{T}^e(h) = \mathbf{T}^{\text{rel}}(h)\mathbf{T}^e(0)$, where $\mathbf{T}^e(0)$ is the end-effector pose after grasping the tool object. The assembled trajectory is then checked for kinematic feasibility, collision avoidance, and task-specific constraints between segments.

This feasibility depends on both the robot embodiment and the current scene. For example, in POURINBOWL, a sampled cup-to-bowl trajectory may be plausible at the object level but infeasible if the wrist collides with nearby obstacles or the pouring pose lies outside the robot workspace. Another candidate with a different grasp or pouring direction may remain feasible. Thus, object–object priors alone are insufficient; sampled interactions must be composed with robot- and scene-specific constraints at test time.

Composition by Rejection Sampling. A natural composition strategy is to sample each segment independently from its learned prior and accept the composition only if $\Gamma(\mathbf{x}^{1:K}, c)$ is executable. Let $\mathcal{E}$ denote the event that the assembled trajectory satisfies all constraints, and let $q(c) = \Pr(\mathcal{E} = 1 \mid c)$ be the feasible mass under this independent proposal. The expected number of trials is $\mathbb{E}[N_{\text{trial}}] = 1/q(c)$. This strategy becomes inefficient when feasible compositions occupy only a small region of the joint sample space, which often occurs under tight grasp, kinematic, and collision constraints.

B. Constraint-Aware Steering

Fig. 3 illustrates why independent sampling can be inefficient. Each sampler assigns high probability to locally plausible samples, shown as the two circles, but only a small subset of their joint combinations satisfies the global composition constraint, such as being parallel to a reference direction. Rejection sampling checks feasibility only after complete samples are generated, wasting computation on incompatible combinations. In contrast, our method guides generation toward feasible compositions during denoising.

We formulate the desired distribution as $p(\mathbf{x}^{1:K} \mid c, \mathcal{E} = 1) \propto p(\mathcal{E} = 1 \mid \mathbf{x}^{1:K}, c) \prod_{k=1}^{K} p_{\theta_k}(\mathbf{x}^k \mid \tilde{c}^k)$, where $\tilde{c}^k$ is the segment-level conditioning input and p_{θ_k} is the diffusion prior for phase k. The feasibility likelihood is $p(\mathcal{E} = 1 \mid \mathbf{x}^{1:K}, c) \propto \exp(\beta \mathcal{S}(\Gamma(\mathbf{x}^{1:K}, c), c))$, where $\mathcal{S}$ scores the assembled trajectory under the current in-context constraints, and β controls the strength of steering.

We approximate this posterior using particle filtering over the joint reverse diffusion process. At denoising step t, we maintain M particles $\{\mathbf{x}_{t,i}^{1:K}\}_{i=1}^{M}$. For each particle, we estimate the clean segment trajectories using Tweedie's formula, $\hat{\mathbf{x}}_{0,i}^k = \frac{1}{\sqrt{\bar{\alpha}_t}}(\mathbf{x}_{t,i}^k - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_{\theta_k}(\mathbf{x}_{t,i}^k, t, \tilde{c}^k))$. The denoised segment estimates are assembled by Γ and scored with weights $w_{t,i} \propto \exp(\beta \mathcal{S}(\Gamma(\hat{\mathbf{x}}_{0,i}^{1:K}, c), c))$. Particles are resampled according to these weights before the next reverse-diffusion step.

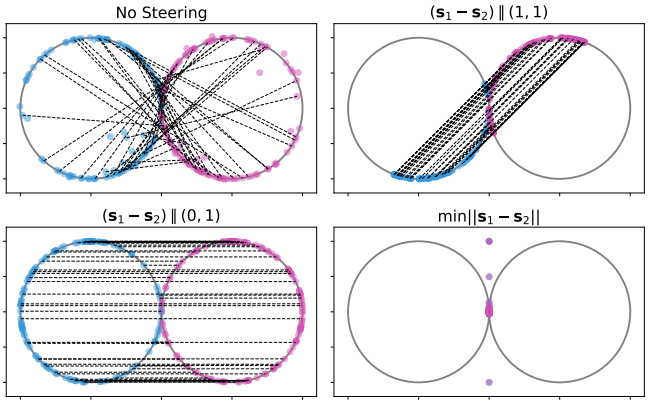


Fig. 3: **Toy illustration.** Independent samplers produce locally plausible samples, but feasible joint compositions occupy only a small subset of the product space. Steering uses feasibility scores during denoising to concentrate particles around valid compositions.

The score $\mathcal{S}$ is soft rather than binary: it can incorporate continuous measures such as IK residuals, collision margins, and motion-planning feasibility. Therefore, particles need not be fully executable at early denoising steps; steering gradually reallocates computation toward particles whose denoised estimates are closer to satisfying the constraints. The full procedure is provided in Alg. 1

Algorithm 1 Particle-Filter-Based Guided Joint Diffusion Sampling

Require: test-time context c; segment-level conditioning $\tilde{c}^{1:K}$; pretrained diffusion models $\{\epsilon_{\theta_k}\}_{k=1}^{K}$; assembly operator Γ; constraint score $\mathcal{S}$; temperature β; number of particles M; diffusion steps T
Ensure: feasible compositional motion $\tau^\star$

Sample initial particles $\mathbf{x}_{T,1:M}^k \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $\forall k \in \{1, \dots, K\}$
for $t = T, \dots, 1$ **do**
  for $k = 1, \dots, K$ **do**
    $\hat{\mathbf{x}}_{0,1:M}^k \leftarrow \frac{1}{\sqrt{\bar{\alpha}_t}}\left(\mathbf{x}_{t,1:M}^k - \sqrt{1-\bar{\alpha}_t}\,\epsilon_{\theta_k}(\mathbf{x}_{t,1:M}^k, t, \tilde{c}^k)\right)$ ▷ Compute denoised estimate
  $\tau_i \leftarrow \Gamma(\hat{\mathbf{x}}_{0,i}^{1:K}, c), \forall i \in \{1, \dots, M\}$ ▷ Assemble trajectory
  $w_{t,i} \leftarrow \exp\big(\beta\, \mathcal{S}(\tau_i, c)\big), \forall i \in \{1, \dots, M\}$ ▷ Evaluate constraints to compute weights
  Normalize $w_{t,1:M}$ to obtain $\tilde{w}_{t,1:M}$
  Resample $\{\tilde{\mathbf{x}}_{t,i}^{1:K}\}_{i=1}^{M}$ according to $\tilde{w}_{t,1:M}$
  for $k = 1, \dots, K$ **do**
    $\mathbf{x}_{t-1,1:M}^k \sim p_{\theta_k}(\mathbf{x}_{t-1,1:M}^k \mid \tilde{\mathbf{x}}_{t,1:M}^k, \tilde{c}^k)$ ▷ Sample reverse step
$i^\star \leftarrow \arg\max_i w_{1,i}$
$\tau^\star = \Gamma(\mathbf{x}_{0,i^\star}^{1:K}, c)$
return $\tau^\star$

C. Implementation Details

We implement Γ using MPLIB [24] to synthesize free-space motions and verify kinematic and collision constraints directly on observed scene point clouds, avoiding the need for explicit object mesh models during planning. Once a feasible trajectory is found, we execute it with platform-specific controllers: OSC [30] at 20 Hz for Franka end-effector targets, and joint-position control for the RBY-1 torso, arms, and dexterous hands, with PD control for mobile-base tracking. Camera, perception, and additional planning details are provided in the appendix.

V. EXPERIMENTS

Our experiments evaluate whether Human2Any enables scalable and transferable manipulation learning from object-centric interaction motion. Specifically, we test six hypotheses. **H1:** Human2Any can acquire challenging manipulation skills without embodiment-aligned robot data. **H2:** Human2Any generalizes to new task contexts without in-context data. **H3:** Human2Any can execute real-world manipulation tasks without real-world robot training data. **H4:** Human2Any transfers across substantially different robot embodiments. **H5:** Constraint-aware steering is critical for efficient and feasible composition. **H6:** Human2Any benefits from scaled interaction-motion data.

A. Experimental Setup and Baselines

Simulation tasks. We design three MuJoCo-based domains from MimicLab [63, 54]: POURINBOWL, HANGMUGTREE, and PREPARETABLE (Fig. 4, top). These tasks evaluate fine-grained motion, long-horizon control, and compositional generalization. Each domain includes three variants with distinct layouts, object arrangements, and interaction contexts. Within each variant, objects are further randomized by approximately 0.1 m translation and 30° rotation. We train on two variants and reserve the third for OOD evaluation. Detailed task descriptions are provided in the appendix.

Real-world tasks. We evaluate real-world execution without real-world robot training data on two distinct platforms: a Franka tabletop setup and an RBY-1 humanoid with dexterous hands (Fig. 4, bottom). For Franka, we use POURINBOWL, HANGMUGTREE, and SORTUTENSILS; for RBY-1, we use POURCUP and USEROLLER. We run ten trials per task with varied object shapes and randomized poses to test robustness across real-world configurations.

Baselines. We compare against DP3 [75], an in-domain behavior cloning policy trained with 100 robot demonstrations per task in simulation and 50 real-robot demonstrations per task in real-world experiments; Im2Flow2Act [72], which predicts 2D object flow from human videos and translates it into robot actions using simulation data; and a Rejection Sampling ablation that uses the same learned components as Human2Any but disables steering.

B. Main Results: Zero-Shot Composition and Generalization

We first evaluate Human2Any on the simulation benchmark in Tab. I, testing challenging skill acquisition without embodiment-aligned robot demonstrations (**H1**), generalization to new task contexts without in-context data (**H2**), and the benefit of constraint-aware steering (**H5**). Each domain contains two in-distribution variants (S1–S2) and one OOD variant with substantially different scene layouts, object arrangements, and interaction contexts.

Human2Any solves challenging tasks without embodiment-aligned data. Across the three domains, Human2Any achieves the strongest overall performance. DP3 performs substantially worse despite using 100 robot demonstrations per task, suggesting that direct action regression struggles with diverse object shapes, poses, layouts, and long-horizon error accumulation. Im2Flow2Act also underperforms, indicating that flow-based action translation does not sufficiently capture embodiment feasibility and 3D scene constraints. These results support **H1**.

Human2Any generalizes to new contexts without in-context data. On OOD variants with unseen layouts and object arrangements, Human2Any maintains strong performance and outperforms baselines across all domains. This suggests that object–object interaction motion decouples task-relevant interaction structure from specific training layouts, enabling generalization to new poses, shapes, and scene contexts. These results support **H2**.

Constraint-aware steering improves feasible composition. Compared with Rejection Sampling, which uses the same learned components but disables steering, Human2Any achieves more consistent performance and stronger OOD generalization. Thus, jointly steering generation with in-context constraints is more effective than independently sampling and filtering candidates, supporting **H5**.

C. Real-world Evaluation

We further evaluate Human2Any on real robot platforms to test whether human-derived interaction priors can be deployed without real-world robot data in the target task contexts (**H3**) and transferred across substantially different embodiments (**H4**). Additional snapshots and videos are provided in the appendix and supplementary material.

Human2Any solves real-world tasks without real-world robot data. On the Franka tabletop setup, we evaluate Human2Any on POURINBOWL, HANGMUGTREE, and SORTUTENSILS (Tab. II). These tasks require precise object interactions, grasp-conditioned execution, and robustness to variation in object shape and state. Across ten trials per task, Human2Any executes the learned interaction motions in real scenes without real-world robot demonstrations, supporting **H3**.

Human2Any transfers across diverse robot embodiments. We further deploy Human2Any on the RBY-1 humanoid mobile robot for POURCUP and USEROLLER. Compared with the Franka setup, RBY-1 has substantially different kinematics. Human2Any achieves success rates of 0.6 and

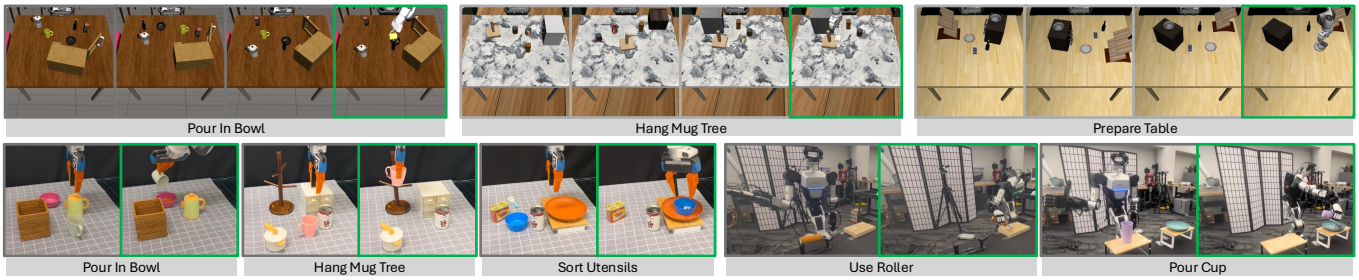


Fig. 4: **Tasks.** We evaluate Human2Any on a diverse set of simulated and real-world manipulation tasks spanning fine-grained object interactions, pick-and-place, tool use, and long-horizon skill chaining. For each simulation task, we show the initial states of three variants, along with one successful final state highlighted by a green border. These tasks cover diverse object shapes, poses, and scene layouts, and are instantiated on different robot embodiments, including a Franka tabletop setup and an RBY-1 humanoid mobile robot.

Method	**PourInBowl**			**HangMugTree**			**PrepareTable**			**Overall**	
	S1	**S2**	**OOD**	**S1**	**S2**	**OOD**	**S1**	**S2**	**OOD**	**ID**	**OOD**
DP3	0.46	0.50	0.00	0.18	0.10	0.00	0.00	0.10	0.00	0.22	0.00
Im2Flow2Act	0.10	0.04	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.03	0.00
Reject Sampling (No Steering)	0.72	**0.56**	0.52	**0.72**	0.42	0.40	0.46	**0.66**	**0.72**	0.59	0.55
Ours	**0.86**	**0.56**	**0.80**	**0.72**	**0.50**	**0.60**	**0.60**	0.56	**0.72**	**0.63**	**0.71**

TABLE I: **Simulation benchmark results.** Success rate comparison across three manipulation tasks under in-distribution (ID; S1–S2) and out-of-distribution (OOD) settings. ID and OOD report average success rates over the corresponding settings.

Method	PIB	HMT	SU	Avg.
DP3	4/10	4/10	0/10	0.27
Im2Flow2Act	1/10	2/10	0/10	0.10
Ours	**8/10**	**7/10**	**9/10**	**0.80**

TABLE II: **Real-world tabletop results.** Success rate comparison across three real-world tabletop manipulation tasks. Task names are abbreviated using initials.

0.7 on the two tasks, respectively, showing that the learned interaction priors are not tied to a specific robot morphology. By resolving embodiment-specific constraints at test time, Human2Any adapts the same object–object interaction knowledge across diverse robot platforms, supporting **H4**.

D. Ablation Study: Dissecting Key Components

Steering improves generation quality and efficiency. (H5) As shown in Fig. 5a, steering reallocates particles from low-feasibility regions toward samples that better satisfy grasp, kinematic, collision, and scene constraints during denoising. Qualitative snapshots show assemblies becoming increasingly feasible, with high-quality samples in green and low-quality samples in red. At the task level, Human2Any achieves the highest throughput across all simulation domains, measured as successful trials within a one-hour budget including inference, planning, and execution (Fig. 5b). This shows that steering improves both motion quality and sampling efficiency by reducing computation spent on infeasible candidates.

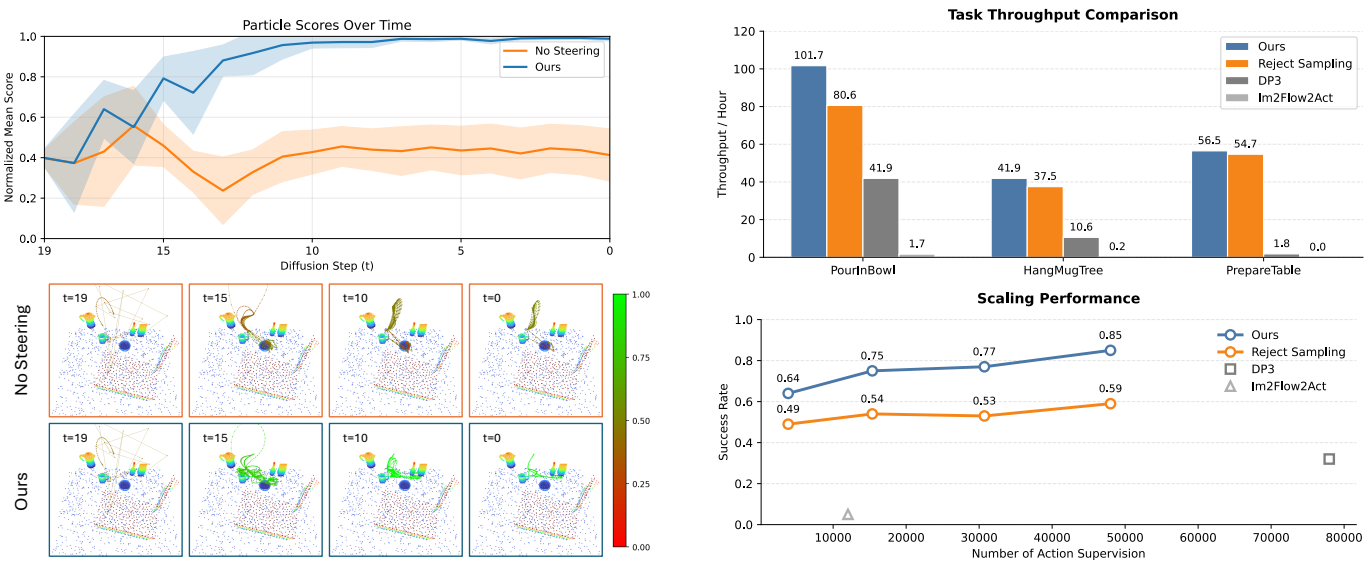
Performance improves with interaction-motion data scale. (H6) We study data scaling on POURINBOWL by varying the quantity and diversity of object-centric interaction trajectories (Fig. 5b). Performance improves with richer motion data, suggesting that Human2Any can effectively leverage diverse interaction supervision. Since these trajectories share the same form as motion extracted from human videos, this provides indirect evidence for scalability with larger human-video sources.

VI. LIMITATIONS

Human2Any focuses on prehensile manipulation and assumes a rigid attachment between the end-effector and tool object, allowing robot control targets to be recovered from a fixed grasp pose and object–object interaction motion. Relaxing this assumption with learned dynamics models could enable broader non-prehensile skills such as in-hand dexterous manipulation. Human2Any assumes that the task skeleton is provided; integrating it with task planning or language-guided decomposition would yield a more complete task-and-motion planning system. The current implementation lacks online failure detection and replanning, which would improve robustness under perception noise, object slippage, and unexpected scene changes, especially in long-horizon execution.

VII. CONCLUSION

We presented Human2Any, a framework for transferring interaction-centric manipulation knowledge from human videos to robots without real-world robot demonstrations.



(a) **Steering visualization.**

(b) **Throughput and performance scaling.**

Fig. 5: **Steering, efficiency, and scaling.** Left: particle-filtering-based steering guides sampled compositions toward higher-feasibility regions under grasp, kinematic, and scene constraints. Right: Human2Any improves task-level throughput and achieves better performance with more interaction-motion training data.

By representing manipulation as object-centric motion, Human2Any separates embodiment-agnostic interaction learning from robot-specific execution through in-context feasibility reasoning and motion planning. Experiments in simulation and on real Franka and RBY-1 platforms show that Human2Any can solve fine-grained and long-horizon tasks while generalizing across object shapes, scene layouts, and robot embodiments.

References

[1] Homanga Bharadhwaj, Abhinav Gupta, Vikash Kumar, and Shubham Tulsiani. Towards generalizable zero-shot manipulation via translating human interaction plans, 2023.

[2] Homanga Bharadhwaj, Roozbeh Mottaghi, Abhinav Gupta, and Shubham Tulsiani. Track2act: Predicting point tracks from internet videos enables generalizable robot manipulation. In *European Conference on Computer Vision*, pages 306–324. Springer, 2024.

[3] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alex Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *arXiv preprint arXiv:2307.15818*, 2023.

[4] Shuo Cheng and Danfei Xu. League: Guided skill learning and abstraction for long-horizon manipulation. *IEEE Robotics and Automation Letters*, 8(10):6451–6458, 2023.

[5] Shuo Cheng, Caelan Garrett, Ajay Mandlekar, and Danfei Xu. Nod-tamp: Generalizable long-horizon planning with neural object descriptors. *arXiv preprint arXiv:2311.01530*, 2023.

[6] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Proceedings of Robotics: Science and Systems (RSS)*, 2023.

[7] Rohan Chitnis, Dylan Hadfield-Menell, Abhishek Gupta, Siddharth Srivastava, Edward Groshev, Christopher Lin, and Pieter Abbeel. Guided search for task and motion plans using learned heuristics. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 447–454. IEEE, 2016.

[8] Rohan Chitnis, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. Learning quickly to plan quickly using modular meta-learning. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 7865–7871. IEEE, 2019.

[9] Quentin Clark and Florian Shkurti. What do you need for diverse trajectory stitching in diffusion planning? *arXiv*

preprint arXiv:2505.18083, 2025.

[10] Open X-Embodiment Collaboration, Abby O'Neill, Abdul Rehman, Abhinav Gupta, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, Albert Tung, Alex Bewley, Alex Herzog, Alex Irpan, Alexander Khazatsky, Anant Rai, Anchit Gupta, Andrew Wang, Andrey Kolobov, Anikait Singh, Animesh Garg, Aniruddha Kembhavi, Annie Xie, Anthony Brohan, Antonin Raffin, Archit Sharma, Arefeh Yavary, Arhan Jain, Ashwin Balakrishna, Ayzaan Wahid, Ben Burgess-Limerick, Beomjoon Kim, Bernhard Schölkopf, Blake Wulfe, Brian Ichter, Cewu Lu, Charles Xu, Charlotte Le, Chelsea Finn, Chen Wang, Chenfeng Xu, Cheng Chi, Chenguang Huang, Christine Chan, Christopher Agia, Chuer Pan, Chuyuan Fu, Coline Devin, Danfei Xu, Daniel Morton, Danny Driess, Daphne Chen, Deepak Pathak, Dhruv Shah, Dieter Büchler, Dinesh Jayaraman, Dmitry Kalashnikov, Dorsa Sadigh, Edward Johns, Ethan Foster, Fangchen Liu, Federico Ceola, Fei Xia, Feiyu Zhao, Felipe Vieira Frujeri, Freek Stulp, Gaoyue Zhou, Gaurav S. Sukhatme, Gautam Salhotra, Ge Yan, Gilbert Feng, Giulio Schiavi, Glen Berseth, Gregory Kahn, Guangwen Yang, Guanzhi Wang, Hao Su, Hao-Shu Fang, Haochen Shi, Henghui Bao, Heni Ben Amor, Henrik I Christensen, Hiroki Furuta, Homanga Bharadhwaj, Homer Walke, Hongjie Fang, Huy Ha, Igor Mordatch, Ilija Radosavovic, Isabel Leal, Jacky Liang, Jad Abou-Chakra, Jaehyung Kim, Jaimyn Drake, Jan Peters, Jan Schneider, Jasmine Hsu, Jay Vakil, Jeannette Bohg, Jeffrey Bingham, Jeffrey Wu, Jensen Gao, Jiaheng Hu, Jiajun Wu, Jialin Wu, Jiankai Sun, Jianlan Luo, Jiayuan Gu, Jie Tan, Jihoon Oh, Jimmy Wu, Jingpei Lu, Jingyun Yang, Jitendra Malik, João Silvério, Joey Hejna, Jonathan Booher, Jonathan Tompson, Jonathan Yang, Jordi Salvador, Joseph J. Lim, Junhyek Han, Kaiyuan Wang, Kanishka Rao, Karl Pertsch, Karol Hausman, Keegan Go, Keerthana Gopalakrishnan, Ken Goldberg, Kendra Byrne, Kenneth Oslund, Kento Kawaharazuka, Kevin Black, Kevin Lin, Kevin Zhang, Kiana Ehsani, Kiran Lekkala, Kirsty Ellis, Krishan Rana, Krishnan Srinivasan, Kuan Fang, Kunal Pratap Singh, Kuo-Hao Zeng, Kyle Hatch, Kyle Hsu, Laurent Itti, Lawrence Yunliang Chen, Lerrel Pinto, Li Fei-Fei, Liam Tan, Linxi "Jim" Fan, Lionel Ott, Lisa Lee, Luca Weihs, Magnum Chen, Marion Lepert, Marius Memmel, Masayoshi Tomizuka, Masha Itkina, Mateo Guaman Castro, Max Spero, Maximilian Du, Michael Ahn, Michael C. Yip, Mingtong Zhang, Mingyu Ding, Minho Heo, Mohan Kumar Srirama, Mohit Sharma, Moo Jin Kim, Muhammad Zubair Irshad, Naoaki Kanazawa, Nicklas Hansen, Nicolas Heess, Nikhil J Joshi, Niko Suenderhauf, Ning Liu, Norman Di Palo, Nur Muhammad Mahi Shafiullah, Oier Mees, Oliver Kroemer, Osbert Bastani, Pannag R Sanketi, Patrick "Tree" Miller, Patrick Yin, Paul Wohlhart, Peng Xu, Peter David Fagan, Peter Mitrano, Pierre Sermanet, Pieter Abbeel, Priya Sundaresan, Qiuyu Chen, Quan Vuong, Rafael Rafailov, Ran Tian, Ria Doshi, Roberto Mart'in-Mart'in, Rohan Baijal, Rosario Scalise, Rose Hendrix, Roy Lin, Runjia Qian, Ruohan Zhang, Russell Mendonca, Rutav Shah, Ryan Hoque, Ryan Julian, Samuel Bustamante, Sean Kirmani, Sergey Levine, Shan Lin, Sherry Moore, Shikhar Bahl, Shivin Dass, Shubham Sonawani, Shubham Tulsiani, Shuran Song, Sichun Xu, Siddhant Haldar, Siddharth Karamcheti, Simeon Adebola, Simon Guist, Soroush Nasiriany, Stefan Schaal, Stefan Welker, Stephen Tian, Subramanian Ramamoorthy, Sudeep Dasari, Suneel Belkhale, Sungjae Park, Suraj Nair, Suvir Mirchandani, Takayuki Osa, Tanmay Gupta, Tatsuya Harada, Tatsuya Matsushima, Ted Xiao, Thomas Kollar, Tianhe Yu, Tianli Ding, Todor Davchev, Tony Z. Zhao, Travis Armstrong, Trevor Darrell, Trinity Chung, Vidhi Jain, Vikash Kumar, Vincent Vanhoucke, Vitor Guizilini, Wei Zhan, Wenxuan Zhou, Wolfram Burgard, Xi Chen, Xiangyu Chen, Xiaolong Wang, Xinghao Zhu, Xinyang Geng, Xiyuan Liu, Xu Liangwei, Xuanlin Li, Yansong Pang, Yao Lu, Yecheng Jason Ma, Yejin Kim, Yevgen Chebotar, Yifan Zhou, Yifeng Zhu, Yilin Wu, Ying Xu, Yixuan Wang, Yonatan Bisk, Yongqiang Dou, Yoonyoung Cho, Youngwoon Lee, Yuchen Cui, Yue Cao, Yueh-Hua Wu, Yujin Tang, Yuke Zhu, Yunchu Zhang, Yunfan Jiang, Yunshuang Li, Yunzhu Li, Yusuke Iwasawa, Yutaka Matsuo, Zehan Ma, Zhuo Xu, Zichen Jeff Cui, Zichen Zhang, Zipeng Fu, and Zipeng Lin. Open X-Embodiment: Robotic learning datasets and RT-X models. https://arxiv.org/abs/2310.08864, 2023.

[11] Xiaoxiang Dong, Matthew Johnson-Roberson, and Weiming Zhi. Joint flow trajectory optimization for feasible robot motion generation from video demonstrations. *arXiv preprint arXiv:2509.20703*, 2025.

[12] Zibin Dong, Yifu Yuan, Jianye Hao, Fei Ni, Yi Ma, Pengyi Li, and Yan Zheng. Cleandiffuser: An easy-to-use modularized library for diffusion models in decision making. *arXiv preprint arXiv:2406.09509*, 2024. URL https://arxiv.org/abs/2406.09509.

[13] Kamil Dreczkowski, Pietro Vitiello, Vitalis Vosylius, and Edward Johns. Learning a thousand tasks in a day. *Science Robotics*, 10(108):eadv7594, 2025.

[14] Ben Eisner, Harry Zhang, and David Held. FlowBot3D: Learning 3D Articulation Flow to Manipulate Articulated Objects. In *Proceedings of Robotics: Science and Systems*, New York City, NY, USA, June 2022. doi: 10.15607/RSS.2022.XVIII.018.

[15] Clemens Eppner, Arsalan Mousavian, and Dieter Fox. ACRONYM: A large-scale grasp dataset based on simulation. In *2021 IEEE Int. Conf. on Robotics and Automation, ICRA*, 2020.

[16] Clemens Eppner, Arsalan Mousavian, and Dieter Fox. Acronym: A large-scale grasp dataset based on simulation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6222–6227. IEEE, 2021.

[17] Hao-Shu Fang, Chenxi Wang, Hongjie Fang, Minghao

Gou, Jirong Liu, Hengxu Yan, Wenhai Liu, Yichen Xie, and Cewu Lu. Anygrasp: Robust and efficient grasp perception in spatial and temporal domains. *IEEE Transactions on Robotics (T-RO)*, 2023.

[18] Xiaolin Fang, Caelan Reed Garrett, Clemens Eppner, Tomás Lozano-Pérez, Leslie Pack Kaelbling, and Dieter Fox. Dimsam: Diffusion models as samplers for task and motion planning under partial observability. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1412–1419. IEEE, 2024.

[19] Roya Firoozi, Johnathan Tucker, Stephen Tian, Anirudha Majumdar, Jiankai Sun, Weiyu Liu, Yuke Zhu, Shuran Song, Ashish Kapoor, Karol Hausman, et al. Foundation models in robotics: Applications, challenges, and the future. *The International Journal of Robotics Research*, 2024. doi: https://doi.org/10.1177/02783649241281508.

[20] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[21] Pete Florence, Corey Lynch, Andy Zeng, Oscar Ramirez, Ayzaan Wahid, Laura Downs, Adrian Wong, Johnny Lee, Igor Mordatch, and Jonathan Tompson. Implicit behavioral cloning. *Conference on Robot Learning (CoRL)*, 2021.

[22] Caelan Reed Garrett, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Pddlstream: Integrating symbolic planners and blackbox samplers via optimistic adaptive planning. In *Proceedings of the international conference on automated planning and scheduling*, volume 30, pages 440–448, 2020.

[23] Caelan Reed Garrett, Rohan Chitnis, Rachel Holladay, Beomjoon Kim, Tom Silver, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. Integrated task and motion planning. *Annual review of control, robotics, and autonomous systems*, 4(1):265–293, 2021.

[24] Runlin (Kolin) Guo, Xinsong Lin, Minghua Liu, Jiayuan Gu, and Hao Su. Mplib: a lightweight motion planning library, 2026. URL https://github.com/haosulab/MPlib. Software available at https://motion-planning-lib.readthedocs.io/latest/.

[25] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[26] Ryan Hoque, Peide Huang, David J Yoon, Mouli Sivapurapu, and Jian Zhang. Egodex: Learning dexterous manipulation from large-scale egocentric video. *arXiv preprint arXiv:2505.11709*, 2025.

[27] Cheng-Chun Hsu, Bowen Wen, Jie Xu, Yashraj Narang, Xiaolong Wang, Yuke Zhu, Joydeep Biswas, and Stan Birchfield. Spot: Se (3) pose trajectory diffusion for object-centric manipulation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4853–4860. IEEE, 2025.

[28] Leslie Pack Kaelbling and Tomás Lozano-Pérez. Hierarchical task and motion planning in the now. In *2011 IEEE international conference on robotics and automation*, pages 1470–1477. IEEE, 2011.

[29] Simar Kareer, Dhruv Patel, Ryan Punamiya, Pranay Mathur, Shuo Cheng, Chen Wang, Judy Hoffman, and Danfei Xu. Egomimic: Scaling imitation learning via egocentric video. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13226–13233. IEEE, 2025.

[30] Oussama Khatib. A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE Journal on Robotics and Automation*, 3(1):43–53, 2003.

[31] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

[32] Nishanth Kumar, William Shen, Fabio Ramos, Dieter Fox, Tomás Lozano-Pérez, Leslie Pack Kaelbling, and Caelan Reed Garrett. Open-world task and motion planning via vision-language model generated constraints. *IEEE Robotics and Automation Letters*, 2026.

[33] Jim Lawrence, Javier Bernal, and Christoph Witzgall. A purely algebraic justification of the kabsch-umeyama algorithm. *Journal of research of the National Institute of Standards and Technology*, 124:1, 2019.

[34] Hongyu Li, Lingfeng Sun, Yafei Hu, Duy Ta, Jennifer Barry, George Konidaris, and Jiahui Fu. Novaflow: Zero-shot manipulation via actionable flow from generated videos. *arXiv preprint arXiv:2510.08568*, 2025.

[35] Yishu Li, Wen Hui Leng, Yiming Fang, Ben Eisner, and David Held. Flowbothd: History-aware diffuser handling ambiguities in articulated objects manipulation. *arXiv preprint arXiv:2410.07078*, 2024.

[36] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023.

[37] Vincent Liu, Ademi Adeniji, Haotian Zhan, Siddhant Haldar, Raunaq Bhirangi, Pieter Abbeel, and Lerrel Pinto. Egozero: Robot learning from smart glasses. *arXiv preprint arXiv:2505.20290*, 2025.

[38] Yunhao Luo, Chen Sun, Joshua B Tenenbaum, and Yilun Du. Potential based diffusion motion planning. *arXiv preprint arXiv:2407.06169*, 2024.

[39] Yunhao Luo, Utkarsh A Mishra, Yilun Du, and Danfei Xu. Generative trajectory stitching through diffusion composition. *arXiv preprint arXiv:2503.05153*, 2025.

[40] Yecheng Jason Ma, Shagun Sodhani, Dinesh Jayaraman, Osbert Bastani, Vikash Kumar, and Amy Zhang. Vip: Towards universal visual reward and representation via value-implicit pre-training. *arXiv preprint arXiv:2210.00030*, 2022.

[41] Aditya Mandalika, Sanjiban Choudhury, Oren Salzman,

and Siddhartha Srinivasa. Generalized lazy search for robot motion planning: Interleaving search and edge evaluation via event-based toggles. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 29, pages 745–753, 2019.

[42] Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In *arXiv preprint arXiv:2108.03298*, 2021.

[43] Ajay Mandlekar, Caelan Garrett, Danfei Xu, and Dieter Fox. Human-in-the-loop task and motion planning for imitation learning. In *7th Annual Conference on Robot Learning*, 2023.

[44] Utkarsh Aashu Mishra, Shangjie Xue, Yongxin Chen, and Danfei Xu. Generative skill chaining: Long-horizon skill planning with diffusion models. In *Conference on Robot Learning*, pages 2905–2925. PMLR, 2023.

[45] Utkarsh Aashu Mishra, Yongxin Chen, and Danfei Xu. Generative factor chaining: Coordinated manipulation with diffusion-based factor graph. In *ICRA 2024 Workshop {\textemdash} Back to the Future: Robot Learning Going Probabilistic*, 2024.

[46] Adithyavairavan Murali, Balakumar Sundaralingam, Yu-Wei Chao, Jun Yamada, Wentao Yuan, 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. URL https://arxiv.org/abs/2507.13097.

[47] Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3m: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601*, 2022.

[48] Sangjun Noh, Dongwoo Nam, Kangmin Kim, Geonhyup Lee, Yeonguk Yu, Raeyoung Kang, and Kyoobin Lee. 3d flow diffusion policy: Visuomotor policy learning via generating flow in 3d space. *arXiv preprint arXiv:2509.18676*, 2025.

[49] Joaquim Ortiz-Haro, Jung-Su Ha, Danny Driess, and Marc Toussaint. Structured deep generative models for sampling on constraint manifolds in sequential manipulation. In *Conference on Robot Learning*, pages 213–223. PMLR, 2022.

[50] Ryan Punamiya, Dhruv Patel, Patcharapong Aphiwetsa, Pranav Kuppili, Lawrence Y Zhu, Simar Kareer, Judy Hoffman, and Danfei Xu. Egobridge: Domain adaptation for generalizable imitation from egocentric human data. In *Human to Robot: Workshop on Sensorizing, Modeling, and Learning from Humans*, 2025.

[51] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[52] Ilija Radosavovic, Tete Xiao, Stephen James, Pieter Abbeel, Jitendra Malik, and Trevor Darrell. Real-world robot learning with masked visual pre-training. In *Conference on Robot Learning*, pages 416–426. PMLR, 2023.

[53] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

[54] Vaibhav Saxena, Matthew Bronars, Nadun Ranawaka Arachchige, Kuancheng Wang, Woo Chul Shin, Soroush Nasiriany, Ajay Mandlekar, and Danfei Xu. What matters in learning from large-scale datasets for robot manipulation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=LqhorpRLIm.

[55] Daniel Seita, Yufei Wang, Sarthak J Shetty, Edward Yao Li, Zackory Erickson, and David Held. Toolflownet: Robotic manipulation with tools via predicting tool flow from point clouds. In *Conference on Robot Learning*, pages 1038–1049. PMLR, 2023.

[56] Junyao Shi, Zhuolun Zhao, Tianyou Wang, Ian Pedroza, Amy Luo, Jie Wang, Jason Ma, and Dinesh Jayaraman. Zeromimic: Distilling robotic manipulation skills from web videos. *arXiv preprint arXiv:2503.23877*, 2025.

[57] Tom Silver, Ashay Athalye, Joshua B Tenenbaum, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Learning neuro-symbolic skills for bilevel planning. *arXiv preprint arXiv:2206.10680*, 2022.

[58] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[59] Shuran Song, Andy Zeng, Johnny Lee, and Thomas Funkhouser. Grasping in the wild: Learning 6dof closed-loop grasping from low-cost demonstrations. *Robotics and Automation Letters*, 2020.

[60] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[61] Yue Su, Xinyu Zhan, Hongjie Fang, Yong-Lu Li, Cewu Lu, and Lixin Yang. Motion before action: Diffusing object motion as manipulation condition. *IEEE Robotics and Automation Letters*, 2025.

[62] TRI LBM Team, Jose Barreiros, Andrew Beaulieu, Aditya Bhat, Rick Cory, Eric Cousineau, Hongkai Dai, Ching-Hsin Fang, Kunimatsu Hashimoto, Muhammad Zubair Irshad, Masha Itkina, Naveen Kuppuswamy, Kuan-Hui Lee, Katherine Liu, Dale McConachie, Ian McMahon, Haruki Nishimura, Calder Phillips-Grafflin, Charles Richter, Paarth Shah, Krishnan Srinivasan, Blake Wulfe, Chen Xu, Mengchao Zhang, Alex Alspach, Maya Angeles, Kushal Arora, Vitor Campagnolo Guizilini, Alejandro Castro, Dian Chen, Ting-Sheng Chu, Sam

Creasey, Sean Curtis, Richard Denitto, Emma Dixon, Eric Dusel, Matthew Ferreira, Aimee Goncalves, Grant Gould, Damrong Guoy, Swati Gupta, Xuchen Han, Kyle Hatch, Brendan Hathaway, Allison Henry, Hillel Hochsztein, Phoebe Horgan, Shun Iwase, Donovon Jackson, Siddharth Karamcheti, Sedrick Keh, Joseph Masterjohn, Jean Mercat, Patrick Miller, Paul Mitiguy, Tony Nguyen, Jeremy Nimmer, Yuki Noguchi, Reko Ong, Aykut Onol, Owen Pfannenstiehl, Richard Poyner, Leticia Priebe Mendes Rocha, Gordon Richardson, Christopher Rodriguez, Derick Seale, Michael Sherman, Mariah Smith-Jones, David Tago, Pavel Tokmakov, Matthew Tran, Basile Van Hoorick, Igor Vasiljevic, Sergey Zakharov, Mark Zolotas, Rares Ambrus, Kerri Fetzer-Borelli, Benjamin Burchfiel, Hadas Kress-Gazit, Siyuan Feng, Stacie Ford, and Russ Tedrake. A careful examination of large behavior models for multitask dexterous manipulation, 2025. URL https://arxiv.org/abs/2507.05331.

[63] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012. doi: 10.1109/IROS.2012.6386109.

[64] Chen Wang, Linxi Fan, Jiankai Sun, Ruohan Zhang, Li Fei-Fei, Danfei Xu, Yuke Zhu, and Anima Anandkumar. Mimicplay: Long-horizon imitation learning by watching human play. *arXiv preprint arXiv:2302.12422*, 2023.

[65] Ruicheng Wang, Jialiang Zhang, Jiayi Chen, Yinzhen Xu, Puhao Li, Tengyu Liu, and He Wang. Dexgraspnet: A large-scale robotic dexterous grasp dataset for general objects based on simulation. *arXiv preprint arXiv:2210.02697*, 2022.

[66] Zi Wang, Caelan Reed Garrett, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. Learning compositional models of robot skills for task and motion planning. *The International Journal of Robotics Research*, 40(6-7):866–894, 2021.

[67] Zhenyu Wei, Zhixuan Xu, Jingxiang Guo, Yiwen Hou, Chongkai Gao, Zhehao Cai, Jiayu Luo, and Lin Shao. $\mathcal{D}(\mathcal{R}, \mathcal{O})$ grasp: A unified representation of robot and object interaction for cross-embodiment dexterous grasping. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4982–4988, 2025. doi: 10.1109/ICRA55743.2025.11127754.

[68] Bowen Wen, Wei Yang, Jan Kautz, and Stan Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17868–17879, 2024.

[69] Bowen Wen, Matthew Trepte, Joseph Aribido, Jan Kautz, Orazio Gallo, and Stan Birchfield. Foundationstereo: Zero-shot stereo matching. *CVPR*, 2025.

[70] Yuxi Xiao, Jianyuan Wang, Nan Xue, Nikita Karaev, Yuri Makarov, Bingyi Kang, Xing Zhu, Hujun Bao, Yujun Shen, and Xiaowei Zhou. Spatialtrackerv2: 3d point tracking made easy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025. URL https://arxiv.org/abs/2507.12462.

[71] Danfei Xu, Ajay Mandlekar, Roberto Martín-Martín, Yuke Zhu, Silvio Savarese, and Li Fei-Fei. Deep affordance foresight: Planning through what can be done in the future. In *2021 IEEE international conference on robotics and automation (ICRA)*, pages 6206–6213. IEEE, 2021.

[72] Mengda Xu, Zhenjia Xu, Yinghao Xu, Cheng Chi, Gordon Wetzstein, Manuela Veloso, and Shuran Song. Flow as the cross-domain manipulation interface. *arXiv preprint arXiv:2407.15208*, 2024.

[73] Zhao-Heng Yin, Sherry Yang, and Pieter Abbeel. Object-centric 3d motion field for robot learning from human videos. *arXiv preprint arXiv:2506.04227*, 2025.

[74] Chengbo Yuan, Chuan Wen, Tong Zhang, and Yang Gao. General flow as foundation affordance for scalable robot learning. *arXiv preprint arXiv gjo:2401.11439*, 2024.

[75] Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. In *Proceedings of Robotics: Science and Systems (RSS)*, 2024.

[76] Harry Zhang, Ben Eisner, and David Held. Flowbot++: Learning generalized articulated objects manipulation via articulation projection. *arXiv preprint arXiv:2306.12893*, 2023.

[77] Zhigen Zhao, Shuo Cheng, Yan Ding, Ziyi Zhou, Shiqi Zhang, Danfei Xu, and Ye Zhao. A survey of optimization-based task and motion planning: From classical to learning approaches. *IEEE/ASME Transactions on Mechatronics*, 30(4):2799–2825, August 2025. ISSN 1941-014X. doi: 10.1109/tmech.2024.3452509. URL http://dx.doi.org/10.1109/TMECH.2024.3452509.

APPENDIX

Supplementary Material

The supplementary material has the following contents:

- **Hardware Setup** (Sec. A): Describes camera, perception, planning, and control details for the Franka and RBY-1 real-world platforms;
- **Tasks and Baselines** (Sec. B): Details simulation and real-world task semantics, reset ranges, evaluation protocols, and baseline settings;
- **Human Data Processing Pipeline** (Sec. C): Describes trajectory fitting, filtering, and qualitative visualizations of RGB-D observations, 3D tracks, and recovered object-centric trajectories;
- **Simulation Data Generation** (Sec. D): Describes how robot-specific agent–object grasp data are generated in simulation for Franka and RBY-1;
- **Constraint-Aware Steering Algorithm** (Sec. **??**): Provides pseudocode for particle-filtering-based steering during constraint-aware composition;
- **Model and Training Details** (Sec. E): Summarizes model architectures, diffusion sampling settings, and training hyperparameters;
- **Additional Results** (Sec. F): Shows qualitative real-world and simulation rollouts across tasks, layouts, and robot embodiments.

Additional videos and visualizations are available on our project website: https://human2any.github.io/

A. Hardware Setup

Fig. 6 shows the real-world hardware setups used in our experiments. We obtain scene point clouds from calibrated cameras. For the Franka setup, we use an Intel RealSense D435; for the RBY-1 setup, we use Meta Aria Gen 2 glasses and estimate depth maps with FoundationStereo [69]. We segment objects with SAM 2 [53] and identify task-relevant objects by extracting CLIP [51] features for each mask region. Once a valid robot trajectory is planned, we execute it with platform-specific controllers: OSC [30] at $20\,\mathrm{Hz}$ for Franka end-effector targets, and joint-position control for the RBY-1 torso, arms, and dexterous hands, with PD control for base tracking.

B. Tasks and Baselines

1) Task Descriptions: We evaluate Human2Any on simulation and real-world manipulation tasks that require interaction-centric motion generation, constraint-aware composition, and generalization across object shapes, poses, layouts, and robot embodiments. Each task is specified by a high-level task skeleton, while the low-level object interaction motions and robot execution trajectories are generated by Human2Any.

a) PourInBowl.: The robot picks up a container and pours its contents into a target bowl. This task requires generating a precise object–object pouring motion while satisfying grasp, reachability, and collision constraints. We use this task to evaluate fine-grained interaction motion and adaptation to different container shapes, bowl poses, and surrounding layouts.

b) HangMugTree.: The robot grasps a mug and hangs it on a mug tree. Unlike tasks with a fixed target pose, the target branch is identified at test time based on robot and scene feasibility, allowing the system to avoid collisions and choose an executable hanging motion. This task evaluates precision manipulation, grasp-conditioned alignment, and constraint-aware target reasoning.

c) PrepareTable.: The robot performs a multi-stage table-setting task, such as placing a bowl on a plate and then placing a cookie box on the bowl. This task requires composing multiple interaction skills over a longer horizon. We use this task to evaluate long-horizon skill chaining and compositional generalization across different object arrangements and scene layouts.

d) SortUtensils.: In the real Franka setup, the robot places a bowl on a plate and then places a utensil on the bowl, requiring multi-stage interaction composition. This task evaluates long-horizon skill chaining and generalization across different object arrangements and scene layouts.

e) PourCup.: On the RBY-1 humanoid mobile robot, the robot performs a cup-pouring task in a larger workspace. This task evaluates whether the object-centric pouring representation can be grounded on a different robot embodiment and workspace.

f) UseRoller.: The RBY-1 robot uses a roller-like tool to interact with a target surface or object. This task evaluates tool-use behavior with a dexterous hand and mobile humanoid embodiment, requiring the learned interaction prior to be grounded through embodiment-specific grasping and motion constraints.

g) Task variants and generalization.: The simulation domains test both local robustness and broader contextual generalization. Each domain contains three variants with distinct global layouts, object arrangements, and interaction contexts; within each variant, object poses are randomized by approximately 0.1 m in translation and 30° in rotation. For each domain, we train on two variants and hold out the third for OOD evaluation, running 50 trials per variant. In real-world experiments, we vary object shapes and poses across trials and evaluate on both Franka and RBY-1, running 10 trials per task. The simulation reset ranges and real-world Franka reset settings are illustrated in Fig. 8 and Fig. 7, respectively.

2) Baseline Settings:

a) Training data.: We use strong supervised variants of the baselines whenever possible. In simulation, DP3 and Im2Flow2Act are trained with 100 robot demonstrations per task from MimicLab [54]. Although the original Im2Flow2Act trains its flow-conditioned policy from predefined random exploration data, designing such exploration rules is non-trivial for our long-horizon, contact-rich tasks. We therefore provide robot demonstrations for both its flow generation module and flow-conditioned policy, giving the baseline at least as strong supervision as task-agnostic exploration data.

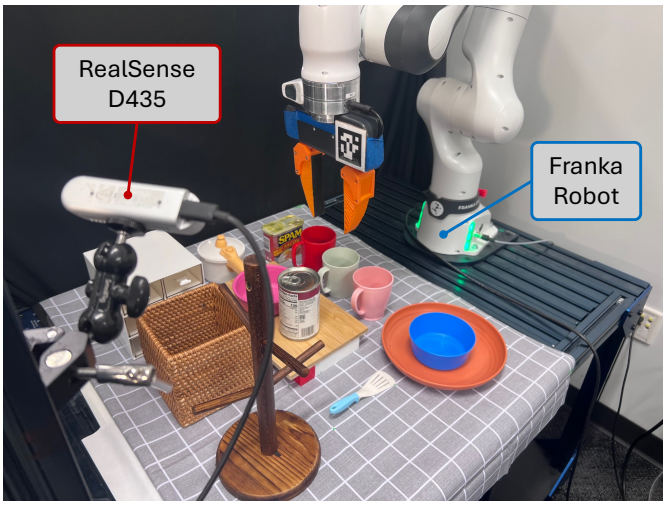


Fig. 6: **Hardware setup.** We evaluate on a Franka tabletop setup with an Intel RealSense D435 and an RBY-1 humanoid mobile robot with ROBOTERA XHAND and Meta Aria Gen 2 glasses.

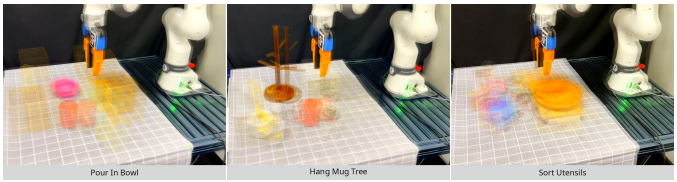


Fig. 7: **Real-world reset settings.** Object pose randomization ranges used for Franka real-world evaluation.

For real-world tasks, DP3 and Im2Flow2Act are trained with 50 real-robot teleoperation demonstrations per task. Since we do not assume a digital twin, we retrain the Im2Flow2Act flow-conditioned policy on real-world teleoperation data. Its flow generation module is trained with the same 50 teleoperation demonstrations plus 50 additional human demonstration videos. For Im2Flow2Act, we downsample the generated flow by $n = 6$ in simulation to fit GPU memory; we also tested accumulated delta end-effector actions, but this did not improve performance.

b) Baseline performance analysis.: The low performance of the baselines highlights the importance of constraint-aware composition. DP3 and Im2Flow2Act receive strong supervision, but do not explicitly reason over how predicted motions satisfy the current grasp, kinematic, collision, and scene constraints when composed into long-horizon executions. For Im2Flow2Act, even when the predicted flow represents plausible object motion, translating it into robot actions remains constraint-dependent: not every flow can be realized under the robot's current grasp, workspace, and collision constraints. Thus, locally plausible predictions can still become infeasible under diverse layouts and object poses. In contrast, Human2Any composes interaction and grasp motions under in-context feasibility constraints, enabling executable rather than merely plausible motion combinations.

Fig. 8: **Simulation reset settings.** Global layout variants and local object pose randomization ranges used in simulation.

C. Human Data Processing Pipeline

1) Trajectory Fitting and Filtering: Given RGB-D human demonstration videos, we first segment the tool and target objects and track 3D points on the tool object across time [70]. The tracked points are transformed into a common world frame (robot base frame) using calibrated camera poses. We then recover the object-centric motion by fitting a rigid transform from the first-frame tool geometry to each subsequent

frame. To reduce the effect of noisy or drifting tracks, we use RANSAC [20] to select inlier correspondences before applying Kabsch alignment [33]. This produces a raw relative trajectory $\tau_{\text{raw}}^{\text{rel}}$.

We further refine the trajectory with temporal smoothing. Specifically, we optimize the per-frame rotations and translations to fit the tracked 3D points while penalizing abrupt changes between consecutive frames. This produces a smoothed relative trajectory $\tau_{\text{smooth}}^{\text{rel}}$, which is used as the object-centric interaction motion for training the object–object interaction sampler. Alg. 2 summarizes the full fitting and filtering procedure, and Alg. 3 details the RANSAC-based inlier selection.

Algorithm 2 Object-Centric Trajectory Extraction from Human Video Tracks

Require: Tracked 3D points $\mathbf{P}^{\text{cam}} \in \mathbb{R}^{H\times M\times 3}$, transform $\mathbf{T}_{\text{cam}}^{\text{world}}$
Ensure: Raw and smoothed relative trajectories $\tau_{\text{raw}}^{\text{rel}}, \tau_{\text{smooth}}^{\text{rel}}$

Transform points to the world frame: $\mathbf{P} \leftarrow \mathbf{T}_{\text{cam}}^{\text{world}}\mathbf{P}^{\text{cam}}$
Use first-frame points $\mathbf{P}_0 = \{\mathbf{p}_{0,m}\}_{m=1}^{M}$ as canonical geometry
Initialize empty raw trajectory $\tau_{\text{raw}}^{\text{rel}}$
for $t = 0, \ldots, H-1$ **do**
  $\mathcal{I}_t \leftarrow \textsc{RansacInliers}(\mathbf{P}_0, \mathbf{P}_t)$
  Estimate $(\mathbf{R}_t, \mathbf{t}_t)$ from inlier correspondences using Kabsch alignment [33]
  Append $\mathbf{T}_t^{\text{raw}} = \begin{bmatrix}\mathbf{R}_t & \mathbf{t}_t \\ \mathbf{0}^\top & 1\end{bmatrix}$ to $\tau_{\text{raw}}^{\text{rel}}$
Refine $\tau_{\text{raw}}^{\text{rel}}$ by optimizing poses:

$$\min_{\{\mathbf{R}_t,\mathbf{t}_t\}_{t=0}^{H-1}} \sum_{t=0}^{H-1}\sum_{m=1}^{M} \|\mathbf{R}_t\mathbf{p}_{0,m} + \mathbf{t}_t - \mathbf{p}_{t,m}\|_2^2 + \lambda_R \sum_{t=1}^{H-1} d_R(\mathbf{R}_t, \mathbf{R}_{t-1})^2 + \lambda_T \sum_{t=1}^{H-1} \|\mathbf{t}_t - \mathbf{t}_{t-1}\|_2^2.$$

Here, $d_R(\cdot,\cdot)$ is the rotation distance, and λ_R, λ_T control temporal smoothness.
Convert optimized poses into $\tau_{\text{smooth}}^{\text{rel}} = \{\mathbf{T}_t^{\text{smooth}}\}_{t=0}^{H-1}$
return $\tau_{\text{raw}}^{\text{rel}}, \tau_{\text{smooth}}^{\text{rel}}$

Algorithm 3 RANSAC Inlier Selection for Rigid Alignment

Require: Corresponding point sets $\mathbf{P}_0 = \{\mathbf{p}_{0,m}\}_{m=1}^{M}$ and $\mathbf{P}_t = \{\mathbf{p}_{t,m}\}_{m=1}^{M}$, batch size B, number of iterations N_{iter}, inlier threshold ϵ
Ensure: Inlier set $\mathcal{I}^\star$

Initialize best inlier set $\mathcal{I}^\star \leftarrow \emptyset$
for $j = 1, \ldots, N_{\text{iter}}$ **do**
  Randomly sample an index subset $\mathcal{B}_j \subseteq \{1, \ldots, M\}$ with $|\mathcal{B}_j| = B$
  Estimate $(\mathbf{R}_j, \mathbf{t}_j)$ from $\{(\mathbf{p}_{0,m}, \mathbf{p}_{t,m})\}_{m\in\mathcal{B}_j}$ using Kabsch alignment [33]
  Compute residuals for all correspondences: $e_m = \|\mathbf{R}_j\mathbf{p}_{0,m} + \mathbf{t}_j - \mathbf{p}_{t,m}\|_2$
  Set $\mathcal{I}_j = \{m \mid e_m < \epsilon\}$
  if $|\mathcal{I}_j| > |\mathcal{I}^\star|$ **then**
    $\mathcal{I}^\star \leftarrow \mathcal{I}_j$
return $\mathcal{I}^\star$

2) Data Visualization and Analysis: Fig. 9 visualizes intermediate outputs from the human data processing pipeline. We show RGB observations with depth overlays, tracked 3D points, and the recovered object-centric trajectories. The visualizations illustrate that the pipeline extracts consistent object motion across diverse demonstrations and filters noisy tracks into smooth relative trajectories suitable for training object–object interaction samplers.

D. Simulation Data Generation

We generate robot-specific agent–object training data in simulation. For the Franka setup, we sample grasp candidates in the object local frame using geometric heuristics, execute the corresponding grasp motions in simulation, and retain successful executions as positive samples, following prior work [15, 46]. We further augment the dataset by applying random rigid transformations to both the object and associated grasp trajectories, improving robustness to object pose variation.

For the RBY-1 dexterous hand, we use DRO [67] to generate candidate dexterous grasps. We curate the dataset by filtering out candidates that can stably hold the object in the air but still contain undesirable collisions or penetrations between the robot hand and the object. The resulting simulation datasets provide embodiment-specific agent–object priors for both the Franka gripper and the RBY-1 dexterous hand, without requiring real-world robot demonstrations.

E. Model and Training Details

We implement all trajectory samplers using the CleanDiffuser codebase [12]. Each sampler is trained as a conditional DDPM over pose trajectories. The model predicts 6-DoF pose actions with action dimension 6 and horizon $H = 16$, conditioned on point-cloud observations. We use a point-transformer condition encoder with feature dimension 256 and a U-Net structure [75] for denoising. During sampling, we use 20 reverse diffusion steps. All models are trained with batch size 512 for 60,000 gradient steps using Adam with learning rate 10^{-4}.

F. Additional Results

1) Real-world Rollouts: Fig. 10 - Fig. 14 show representative real-world executions of Human2Any on both the Franka tabletop setup and the RBY-1 humanoid mobile robot. These rollouts cover interaction-centric skills such as pouring, hanging, table setting, and tool use, demonstrating that human-derived object–object interaction priors can be composed with robot-specific constraints to produce executable motions across different embodiments and scene contexts.

2) Simulation Rollouts: Fig. 17 - Fig. 15 show representative simulation executions across our task domains. These rollouts illustrate how Human2Any composes learned interaction priors with robot- and scene-specific constraints to generate executable trajectories for fine-grained manipulation, long-horizon skill chaining, and diverse object arrangements.

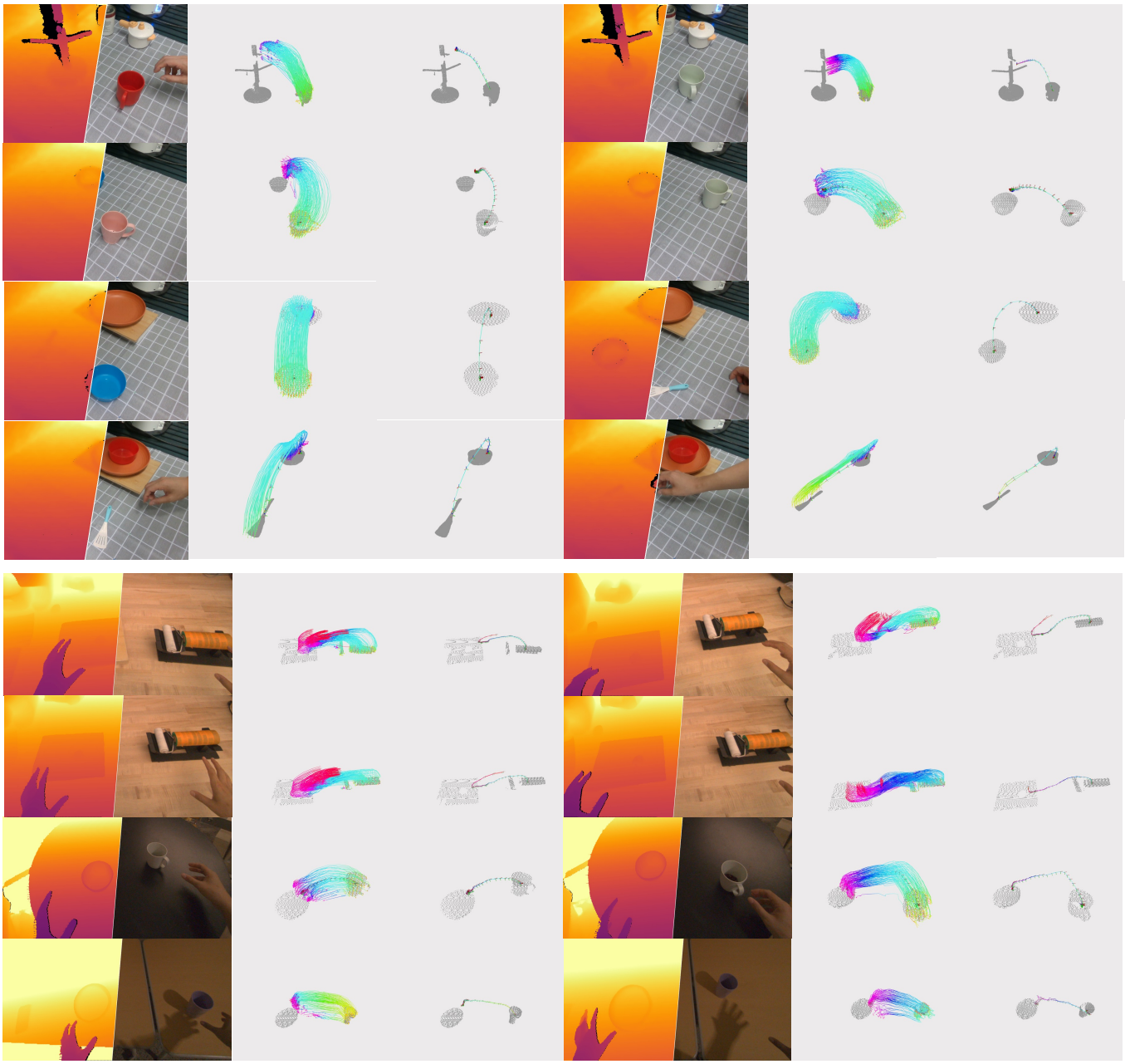


Fig. 9: **Human data processing.** We visualize RGB observations with depth overlays, 3D point tracks, and recovered object-centric trajectories from human demonstrations.

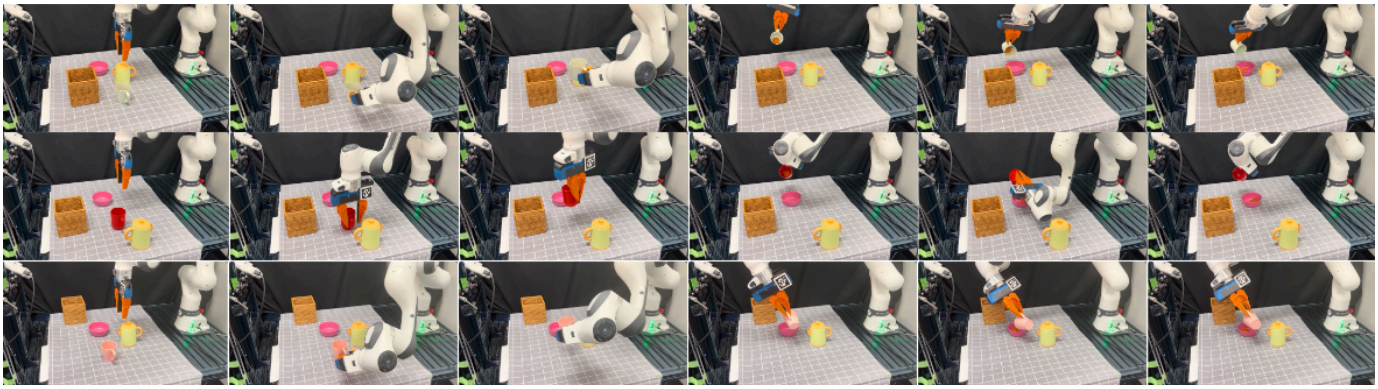


Fig. 10: **Real-world rollouts on PourInBowl tasks.** The robot avoids the kettle, grasps the cup by its handle, and rotates it during the transfer motion to align the cup for pouring into the bowl.

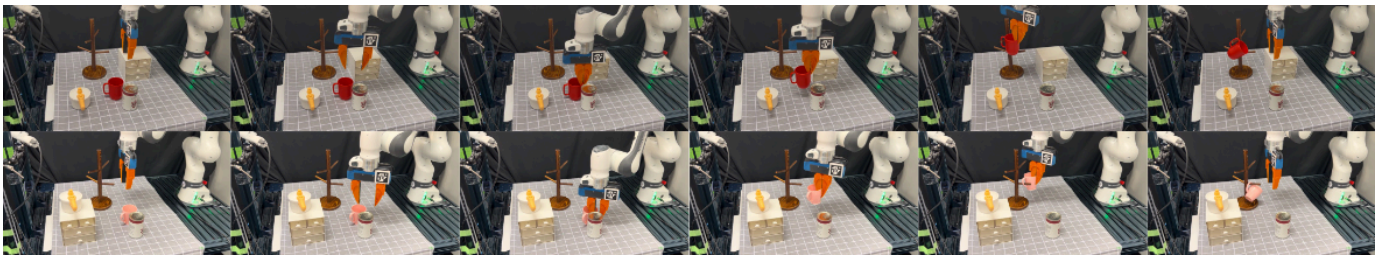


Fig. 11: **Real-world rollouts on HangMugTree tasks.** The robot aligns its grasp to avoid non-target objects and reorients the mug on the fly to accurately place the handle onto the mug tree.

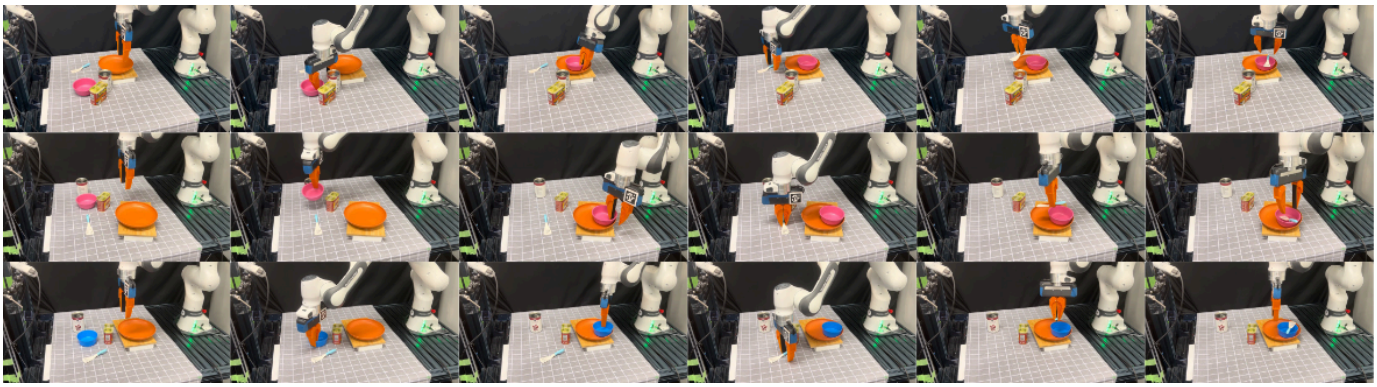


Fig. 12: **Real-world rollouts on SortUtensils tasks.** The robot sequentially places a bowl onto a plate and a utensil onto the bowl, demonstrating long-horizon skill composition across varying object arrangements and scene layouts.

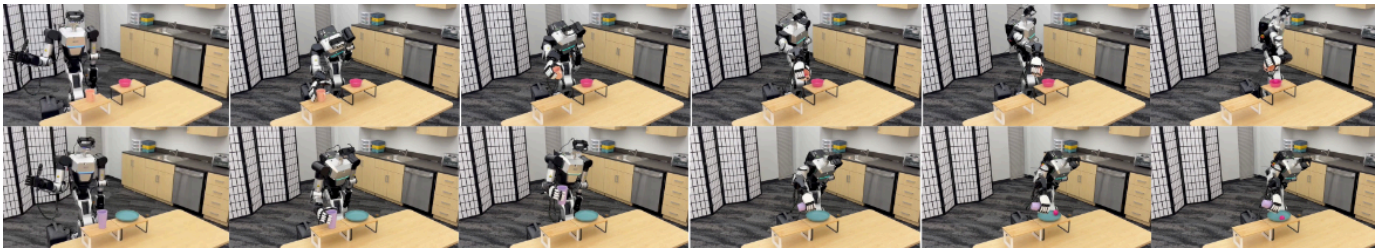


Fig. 13: **Real-world rollouts on PourCup tasks.** The robot firmly grasps the cup and leverages whole-body motion to pour the pink ball onto the target container.

Fig. 14: **Real-world rollouts on UseRoller tasks.** The robot grasps the handle of the roller precisely and move to press the target dough.

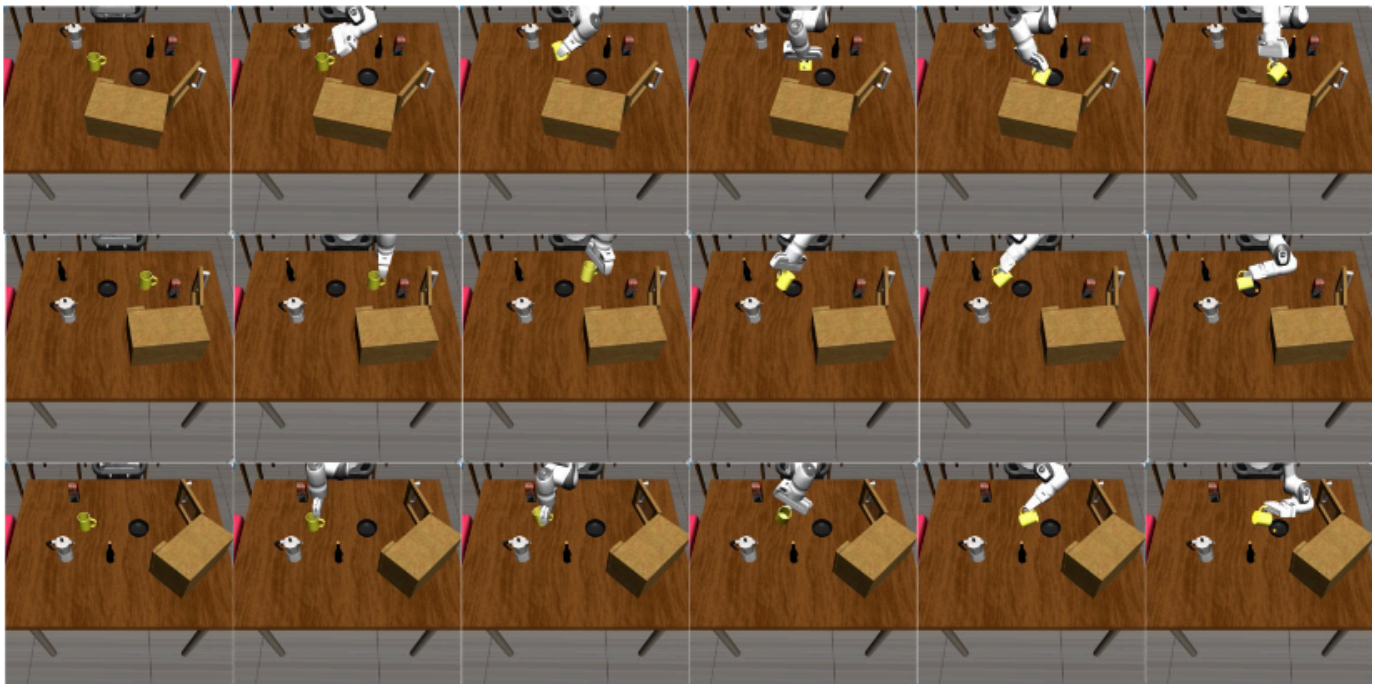


Fig. 15: **Simulation rollouts.** Qualitative POURINBOWL executions showing precise pouring interactions under varied object poses and scene layouts.

Fig. 16: **Simulation rollouts.** Qualitative HANGMUGTREE executions showing fine-grained interaction motion and precise mug–branch alignment.

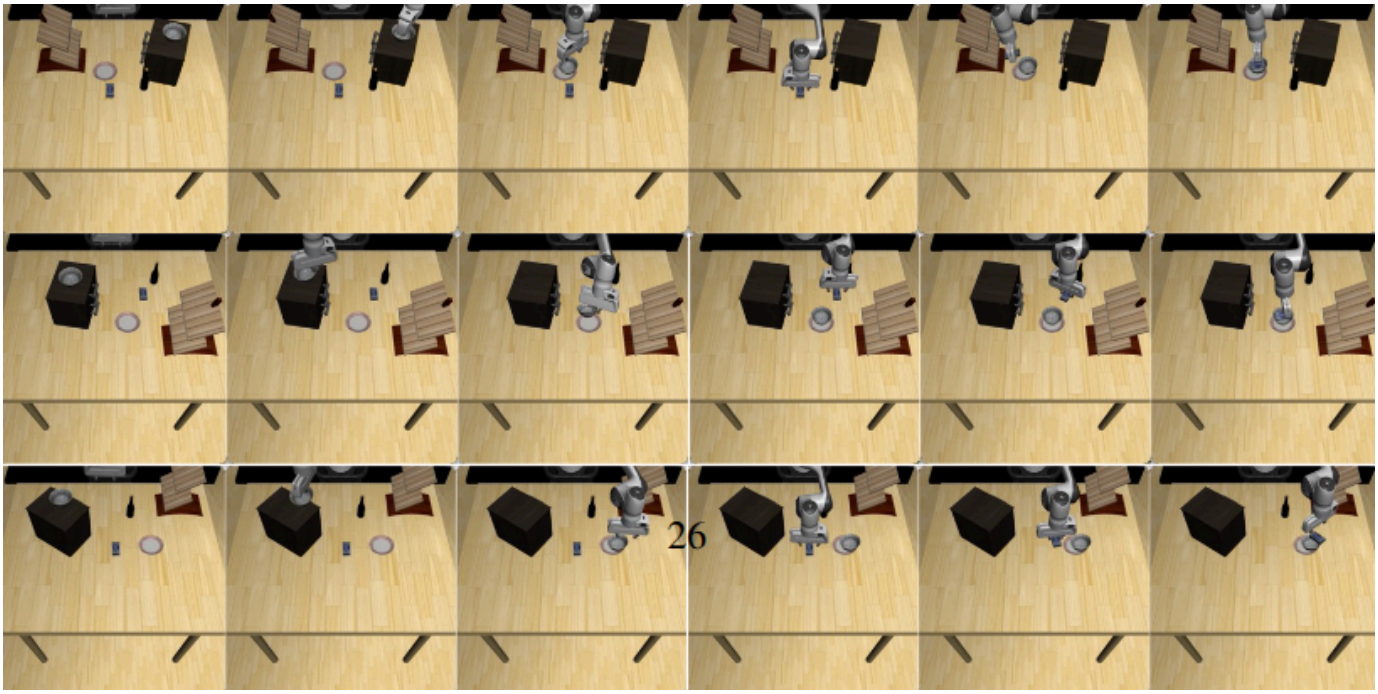


Fig. 17: **Simulation rollouts.** Qualitative executions of Human2Any on PREPARETABLE, demonstrating long-horizon skill chaining through multi-stage interaction composition.