Plan-Conditioned Imitation for Robust Object Retrieval under Self-Occlusion in Dense Clutter

Anonymous Authors

***Abstract*— Retrieving objects from dense clutter requires rearrangement during which the manipulator can occlude objects while moving them. Repeated arm withdrawals to restore visibility interrupt execution. We introduce TRACE, a plan-conditioned imitation framework for retrieval under self-occlusion. A single unoccluded observation initializes a digital twin, where a privileged teacher generates a fixed nominal rollout. A recurrent student combines local rollout context, partial object observations, and proprioception to select actions that can correct deviations from the prediction. Behavior cloning initializes the student; DAgger refines it with teacher labels on student-visited states. The rollout remains fixed throughout execution, so the deployed student needs neither online teacher queries nor additional simulator rollouts during pushing. On 511 simulation test scenes, TRACE achieves 90.7% success versus 43.4% for nominal replay and 96.7% for the privileged closed-loop teacher. At a matched 26,373-label budget, student-state supervision achieves 87.8% versus 66.7% for expert-only cloning, demonstrating gains beyond additional labels. On a UR5e, TRACE achieves 90.0% success versus 95.0% for the closed-loop teacher, while reducing total execution time from 192.7 s to 67.3 s. It avoids the teacher's 16.8 sensing-related arm retractions per trial during pushing, retaining a final withdrawal for graspability evaluation. Code and data will be released at: https://icra27-trace.github.io/**

I. Introduction

A target object in dense clutter may be visible yet inaccessible to a gripper. Creating grasping clearance requires rearranging neighboring objects, during which the robot's arm may block the camera's view while contact continues to move objects. We address reliable target retrieval under manipulator-induced self-occlusion while limiting interruptions for visual-state reacquisition. Applications include acquiring gears and electrical connectors from assembly kit trays [1], selecting books and glue bottles for warehouse fulfillment [2], and retrieving remote controls for people with motor impairments [3]. Such tasks motivate creating grasping space while maintaining feedback without repeated sensing interruptions. We study dense planar clutter with known object footprints and a complete initial scene observation.

Each push changes the contacts and clearances that determine which actions remain useful, making retrieval a long-horizon, contact-rich decision problem. Predictive methods reason about anticipated push outcomes using learned interaction models, tree search, or rigid-body simulation [4], [5], [6], while reactive policies choose successive manipulations from scene observations [7], [8]. During self-occlusion, hidden objects may move, making earlier observations inaccurate. Retracting the arm restores visibility but interrupts manipulation; replaying a predicted sequence cannot correct for contact uncertainty or execution error. The challenge is to combine prediction with intermittent feedback to continue making useful corrective actions.

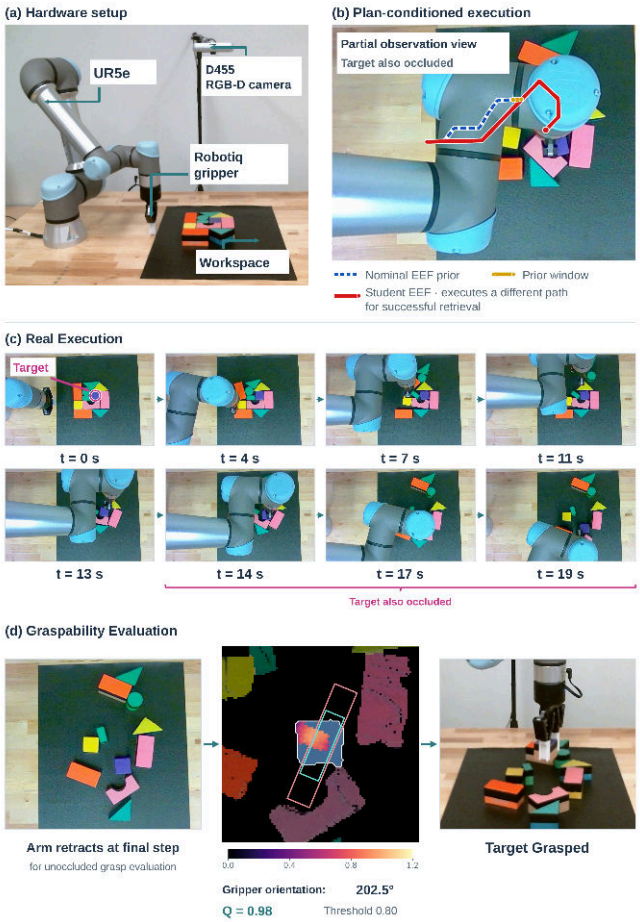

Fig. 1. **Hardware setup and plan-conditioned execution.** (a) UR5e with fixed external RGB-D sensing. (b) TRACE execution during target occlusion; the nominal EEF path and active local plan window are shown for reference. (c) Final graspability evaluation. (d) Example pushing sequence through successful retrieval.

We introduce TRACE, *Teacher Rollouts for Adaptive Closed-loop Execution*, a plan-conditioned imitation learning framework that combines a fixed predictive reference, recurrent memory, and partial visual feedback. A single unoccluded observation initializes a digital twin, where a frozen teacher trained with complete geometric state generates a scene-specific nominal rollout $\bar{\tau}$. The rollout stores predicted end-effector positions and object centers. During execution, a GRU-based student combines a local window from this rollout with visible object geometry, visibility and observation-age indicators, proprioception, and the pre-

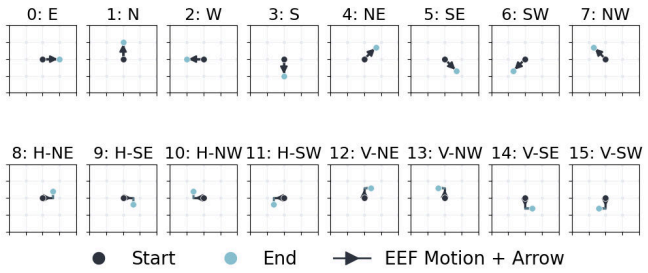

Fig. 2. **End-effector motion primitives.** The 16-action library contains four cardinal, four diagonal, and eight two-segment staircase motions, executed horizontal-first (H-XX) or vertical-first (V-XX).

vious action. The rollout supplies an expectation of scene evolution, memory carries information through observation gaps, and current detections provide evidence of execution-induced deviations. The student predicts the complete next motion primitive and can therefore depart from the nominal path. The rollout also supplies a scene-specific execution budget, after which the robot checks graspability. After the initial rollout, pushing requires neither teacher queries nor additional simulator rollouts.

Corrective actions change the states the student subsequently encounters, including configurations absent from teacher-controlled demonstrations. We address this distribution shift by initializing the student with behavior cloning and refining it using DAgger [9]. The student controls execution while the frozen teacher supplies action distributions on student-visited states. Training includes simulated self-occlusion, detection dropout, multi-step observation blackouts, and planning–execution mismatch. The same teacher thus provides both pre-execution predictive context and supervision for acting in learner-induced configurations.

The abstract summarizes the main simulation and hardware results. Our contributions are:

- We develop TRACE, which combines a one-time privileged teacher rollout with recurrent partial-observation control for object retrieval under self-occlusion.
- We quantify the benefit of privileged supervision on student-visited states using matched-label comparisons with expert-only behavior cloning, and evaluate recurrent memory and plan-context horizon.
- We demonstrate in simulation and on a real robot that TRACE improves retrieval success over nominal-plan replay and reduces execution time relative to a complete-state teacher by avoiding repeated arm withdrawals for sensing during pushing.

II. Related Work

A. Object Retrieval through Non-Prehensile Rearrangement

Push-grasp methods rearrange clutter to increase grasp accessibility [10], [11], [12], while Mechanical Search studies retrieval of targets hidden by clutter [13], [7]. Predictive methods reason about future accessibility: Visual Foresight Trees searches over outcomes predicted by a learned multi-object interaction model [4]. MORE combines Monte Carlo tree search with self-supervised learning to improve subsequent search [5], while PMBS accelerates long-horizon planning through batched rigid-body simulation [6]. TRACE instead computes a single pre-execution rollout as predictive context for a learned policy.

Visuomotor Mechanical Search learns closed-loop retrieval [7]. Kiatos et al. also execute sequential pre-grasp pushes without retracting for each observation [8]. Our distinction is rollout-conditioned recurrent feedback with explicit missing-object observations after an initially visible scene, rather than uninterrupted pushing alone. This also differs from searching for a target initially hidden by clutter [14], [13].

B. Privileged Learning under Partial Observations

Learning by Cheating distills a privileged state-based teacher into a vision-based student [15]. In manipulation, Mosbach and Behnke use memory-augmented student-teacher learning for retrieval from imperfect visual detections [16], while VIRAL uses behavior cloning and online DAgger to transfer privileged full-state humanoid policies to RGB-based control [17].

Distinct teacher states can yield the same student observation but require different actions, creating the realizability issue studied by Kim et al. [18]. TRACE combines recurrent observation history with a nominal rollout from the initial scene. Privileged state supplies training labels and pre-execution context; deployment uses partial observations and the fixed rollout.

C. Imitation on Learner States and Nominal Guidance

Behavior cloning trains on demonstrator states; DAgger addresses the resulting covariate shift by labeling learner-induced states [9]. In rearrangement, one different push changes subsequent contacts and observations. We match label budgets to isolate the benefit of learner-state coverage using a frozen teacher. Residual Reinforcement Learning adds a learned corrective action to the output of a conventional controller [19]. TRACE uses a local rollout window as input and predicts the complete next primitive, without constraining it to a nominal action plus a residual.

III. Problem Formulation

We study target retrieval from dense planar clutter under manipulator-induced self-occlusion. A robot rearranges neighboring objects through non-prehensile pushes until a designated target admits a collision-free top-down grasp. The bounded workspace $\mathcal{W} \subset \mathbb{R}^2$ contains n rigid, movable objects, including the target. A fixed external camera (Fig. 1(a)) initially observes the complete scene, from which the robot obtains an initial estimate $\hat{s}_0$. During manipulation, however, the arm and gripper may occlude objects while contact continues to change their poses.

We model online execution as a finite-horizon discounted partially observable Markov decision process $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \mathcal{O}, \mathcal{Z}, \gamma, H)$, where $\mathcal{S}$ and $\mathcal{A}$ denote the state

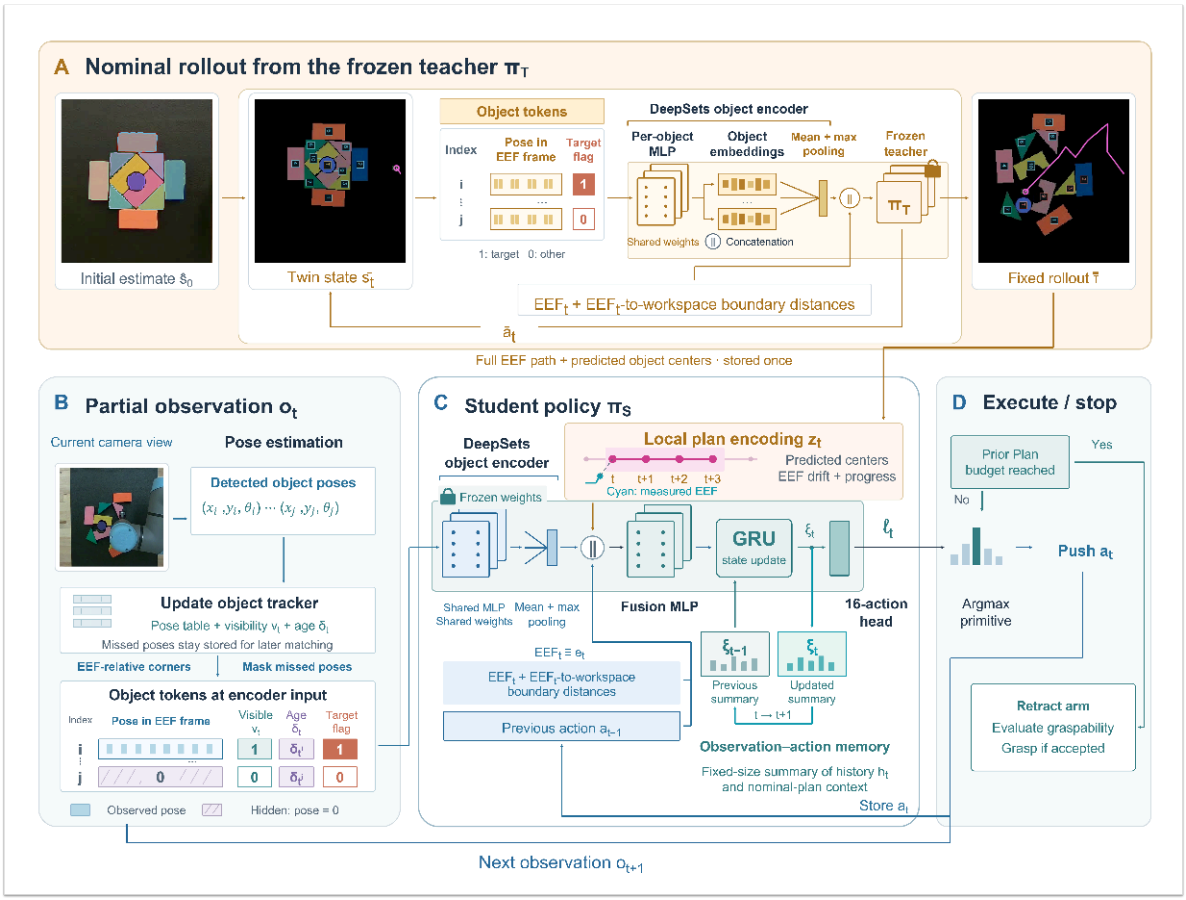

Fig. 3. **TRACE inference pipeline.** (A) A clean initial scene estimate $\hat{s}_0$ initializes a digital twin, where the frozen privileged teacher generates the fixed nominal rollout $\bar{\tau}$. (B) Current detections update object tracks; unavailable object geometry is zeroed while visibility, observation age, and target identity remain available. (C) The student combines the DeepSets scene representation, local plan context z_t, end-effector features, and previous action. A GRU updates recurrent state ξ_t, and a categorical head predicts one of the 16 motion primitives. (D) The selected primitive is executed and the next partial observation closes the loop. After panel (A), online execution requires neither teacher queries nor additional simulator rollouts.

and action spaces, $\mathcal{T}$ the transition model, $\mathcal{R}$ the reward, $\mathcal{O}$ and $\mathcal{Z}$ the observation space and observation model, $\gamma \in [0, 1]$ the discount factor, and H the decision horizon. A privileged teacher π_T is trained from complete state information, whereas the deployable policy π_S is trained from teacher supervision and acts under partial observations.

State Space. At decision step t, the latent state is $s_t = (q_t^R, e_t, q_t^1, \ldots, q_t^n)$, where q_t^R denotes the robot configuration, $e_t \in \mathbb{R}^2$ is the planar end-effector position, and $q_t^i \in SE(2)$, parameterized by $(x_t^i, y_t^i, \theta_t^i)$, is the pose of object i. The robot configuration determines the current collision-body poses used by the observation model. Object footprint geometry is known and fixed.

Action Space. The action space $\mathcal{A}$ contains the 16 discrete planar end-effector motion primitives shown in Fig. 2: four cardinal, four diagonal, and eight two-segment staircase motions. Each primitive begins at the current end-effector position and uses a fixed commanded motion scale ζ.

Transition Function. Executing $a_t \in \mathcal{A}$ produces $s_{t+1} \sim \mathcal{T}(\cdot \mid s_t, a_t)$. The transition captures free-space end-effector motion, direct gripper–object contact, and the resulting object–object interactions.

Observation. At decision step t, the robot receives $o_t = (e_t, \{\hat{q}_t^j\}_{j \in \mathcal{V}_t}) \in \mathcal{O}$, where $\mathcal{V}_t$ indexes objects detected in the current camera view and e_t remains available through proprioception. The observation model $\mathcal{Z}(o_t \mid s_t)$ describes partial observations induced by manipulator self-occlusion.

The current observation updates an object tracker initialized from $\hat{s}_0$. For each object i, it maintains visibility $v_t^i \in \{0, 1\}$ and observation age δ_t^i, which is reset to zero when the object is observed and incremented otherwise. The geometry supplied to the policy is

$$\tilde{q}_t^i = \begin{cases} \mathrm{Rel}(\hat{q}_t^i, e_t), & v_t^i = 1, \\ \mathbf{0}, & v_t^i = 0, \end{cases}$$

where $\mathrm{Rel}(\cdot)$ expresses planar geometry relative to the current end effector. With target indicator χ^i, the structured observation is

$$\tilde{o}_t = \left(e_t, \{(\tilde{f}_t^i, v_t^i, \delta_t^i, \chi^i)\}_{i=1}^n\right).$$

Reward. The reward $\mathcal{R}$ is used only to train the privileged teacher, encouraging progress toward target graspability while maintaining workspace containment. The student π_S does not optimize this reward directly; it learns from teacher supervision as described in Sec. IV-D. The teacher reward is defined in Sec. IV-A.

IV. METHOD

Given the initial scene estimate $\hat{s}_0$, we first roll out a frozen privileged teacher π_T in a digital twin to obtain a scene-specific nominal trajectory $\bar{\tau}$. A GRU-based student then executes the task from partial observations while conditioning on a local context extracted from this fixed rollout. The nominal trajectory provides predictive context rather than commands to replay, allowing the student to depart from the predicted execution as the physical scene evolves. Figure 3 summarizes the inference pipeline.

A. Learning a Privileged Retrieval Teacher

The privileged teacher observes the complete task geometry, all object poses and the current end-effector state from s_t and selects actions from the same 16 motion primitives used during deployment (Fig. 2). Each object is represented by its end-effector-relative planar geometry together with target indicator χ^i.

Both teacher and student use a DeepSets-style permutation-invariant scene encoder [20]. For object tokens $U = \{u^i\}_{i=1}^n$, we define $\mathcal{E}_\theta(U) = [\frac{1}{n}\sum_i \psi_\theta(u^i), \max_i \psi_\theta(u^i)]$, where ψ_θ is a shared object MLP and the maximum is componentwise. This pooling ensures permutation invariance; we make no universality claim for this encoder.

Let $u_t^{T,i}$ denote the privileged token of object i. The teacher scene representation is $\phi_t^T = \mathcal{E}_{\theta_T}(\{u_t^{T,i}\}_{i=1}^n)$. Let c_t contain the end-effector position and its distances to the four workspace boundaries. The teacher updates $h_t^T = \mathrm{GRU}_T(G_T([\phi_t^T, c_t]), h_{t-1}^T)$ and produces $p_t^T = \mathrm{softmax}(f_T(h_t^T)) = \pi_T(\cdot \mid s_t, h_{t-1}^T)$. A separate value head supports PPO training [21].

The teacher is optimized using a graspability-based reward. Let $g(s) \in [0,1]$ denote graspability confidence predicted by the Grasp Network adopted from PMBS [6], and define $\Phi(s) = 2g(s)$. The reward is $r_t = 10$ when $g(s_{t+1}) > 0.9$ and otherwise

$$r_t = \eta_t\,[\gamma\Phi(s_{t+1}) - \Phi(s_t)] - 0.1 - f_t, \tag{1}$$

where $\gamma = 0.99$, $\eta_t = 0$ on the first transition after reset and 1 thereafter, and f_t penalizes workspace violations. The potential difference follows reward shaping [22]; the initial-step gate and terminal reward mean policy invariance is not assumed here. The step cost favors shorter solutions. After training, the teacher is frozen and discrete actions are selected as $a_t^T = \arg\max_{a\in\mathcal{A}} p_t^T(a)$.

B. Nominal Rollout and Local Plan Context

Before execution, $\hat{s}_0$ initializes the digital twin and the frozen teacher is rolled out once (Fig. 3(A)). We retain the nominal trajectory $\bar{\tau} = \{(\bar{e}_j, \bar{P}_j)\}_{j=0}^{N-1}$, where $\bar{P}_j = \{\bar{p}_j^i\}_{i=1}^n$ contains the predicted object centers. The rollout remains fixed throughout the corresponding execution episode. At student decision t, let $j_k = \min(t + k, N-1)$ for $k \in \{0,1,2,3\}$. The local plan context is $z_t = [\{\bar{e}_{j_k} - e_t\}_{k=0}^3, \rho_t, \beta_t, \{\bar{p}_{j_0}^i - e_t\}_{i=1}^n]$, where $\rho_t = \min(1, t/\max(1, N-1))$ denotes rollout progress and $\beta_t = \mathbf{1}[t \geq N]$ indicates nominal-rollout exhaustion. Requested indices beyond the stored rollout reuse its final sample.

Algorithm 1 TRACE Student Training

Require: Training scenes $\mathcal{S}_{\text{train}}$, frozen teacher π_T, DAgger rounds K

$\mathcal{D}_0 \leftarrow \varnothing$
for each scene in $\mathcal{S}_{\text{train}}$ **do**
  $\bar{\tau} \leftarrow \textsc{NominalRollout}(\pi_T)$
  for each teacher-visited state s_t **do**
    $\tilde{o}_t \leftarrow \textsc{PartialObs}(s_t)$
    $x_t \leftarrow [\phi_t^S, c_t, z_t, \mathrm{onehot}(a_{t-1})]$
    $\mathcal{D}_0 \leftarrow \mathcal{D}_0 \cup \{(x_t, p_t^T)\}$
  end for
end for
$\pi_S^0 \leftarrow \textsc{FitStudent}(\mathcal{D}_0)$
for $k = 1, \ldots, K$ **do**
  $\mathcal{B}_k \leftarrow \varnothing$
  for each scene in $\mathcal{S}_{\text{train}}$ **do**
    $\bar{\tau} \leftarrow \textsc{NominalRollout}(\pi_T)$
    for each state s_t visited by π_S^{k-1} **do**
      $\tilde{o}_t \leftarrow \textsc{PartialObs}(s_t)$
      $x_t \leftarrow [\phi_t^S, c_t, z_t, \mathrm{onehot}(a_{t-1})]$
      $p_t^T \leftarrow \pi_T(\cdot \mid s_t, h_{t-1}^T)$
      $\mathcal{B}_k \leftarrow \mathcal{B}_k \cup \{(x_t, p_t^T)\}$
    end for
  end for
  $\mathcal{D}_k \leftarrow \mathcal{D}_{k-1} \cup \mathcal{B}_k$
  $\pi_S^k \leftarrow \textsc{FitStudent}(\mathcal{D}_k)$ ▷ Initializes a new student and trains it on $\mathcal{D}_k$ using Eq. (2)
end for
return student selected on held-out validation scenes

function $\textsc{PartialObs}(s_t)$
  $\bar{v}_t^i \leftarrow \mathbf{1}[F_t^i \cap M_t^R = \varnothing]$
  $d_t^i \sim \mathrm{Bernoulli}(p_{\text{drop}}), \quad \kappa_t = \mathbf{1}[t \in \mathcal{B}_{\text{out}}]$
  $v_t^i \leftarrow (1-\kappa_t)\,\bar{v}_t^i(1-d_t^i), \quad \forall i$
  $\delta_t^i \leftarrow (1-v_t^i)(\delta_{t-1}^i + 1), \quad \tilde{q}_t^i \leftarrow v_t^i\,\mathrm{Rel}(\hat{q}_t^i, e_t)$
  return $\tilde{o}_t = (e_t, \{(\tilde{q}_t^i, v_t^i, \delta_t^i, \chi^i)\}_{i=1}^n)$
end function

All nominal positions are expressed relative to the current measured end-effector position. Thus, $\bar{\tau}$ remains fixed while z_t changes with physical execution. The student receives geometric plan context rather than the teacher's nominal primitive identities.

C. Plan-Conditioned Recurrent Student

At each decision, the structured observation $\tilde{o}_t$ defined in Sec. III provides one token for every tracked object (Fig. 3(B)). Visible objects contribute their current end-effector-relative geometry; unavailable geometry is zeroed, while visibility, observation age, and target identity remain encoded. For object token $u_t^{S,i}$, the student scene representation is $\phi_t^S = \mathcal{E}_{\theta_S}(\{u_t^{S,i}\}_{i=1}^n)$, using the same mean–max aggregation as the teacher but separate parameters. The instantaneous network input is $x_t = [\phi_t^S, c_t, z_t, \mathrm{onehot}(a_{t-1})]$. A fusion MLP produces $u_t = G_S(x_t)$, and the recurrent state updates as $\xi_t = \mathrm{GRU}_S(u_t, \xi_{t-1})$ (Fig. 3(C)). The recurrent state allows information from earlier observations and actions to influence decisions when current object geometry is unavailable. It is not trained to explicitly reconstruct the hidden scene; instead, it provides task-relevant temporal context for action selection. The categorical action head produces $p_t^S = \mathrm{softmax}(f_S(\xi_t)) = \pi_S(\cdot \mid x_t, \xi_{t-1})$, and the executed primitive is $a_t^S = \arg\max_{a\in\mathcal{A}} p_t^S(a)$.

D. Learning from Privileged Supervision

We train the student in two stages using supervision from the frozen teacher. For each training scene, the teacher first

generates the fixed nominal rollout $\bar{\tau}$ from the nominal scene. Student trajectories are then collected in the corresponding execution scene, which may be perturbed while $\bar{\tau}$ remains unchanged. During data collection, we simulate manipulator-induced self-occlusion using the current robot geometry. For collision body b, let V^b denote its collision-mesh vertices and T_t^b its current pose. Its planar occlusion region is $M_t^b = \mathrm{Rect}_\epsilon(\mathrm{Hull}(\Pi_{xy}(T_t^b V^b)))$, and $M_t^R = \bigcup_b M_t^b$. Here, Π_{xy} projects transformed collision geometry onto the workspace plane and Rect_ϵ is a minimum-area enclosing rectangle padded by $\epsilon = 10\,\mathrm{mm}$. With object footprint F_t^i, self-occlusion visibility is $\bar{v}_t^i = \mathbf{1}[F_t^i \cap M_t^R = \varnothing]$. This provides a conservative planar approximation of manipulator self-occlusion rather than an exact rendering of the camera silhouette. Training additionally applies independent token dropout $d_t^i \sim \mathrm{Bernoulli}(p_{\mathrm{drop}})$ and scheduled multi-decision blackouts with indicator $\kappa_t = \mathbf{1}[t \in \mathcal{B}_{\mathrm{out}}]$. The resulting visibility is $v_t^i = (1-\kappa_t)\bar{v}_t^i(1-d_t^i)$. Unavailable object geometry is zeroed, while observation age δ_t^i is reset when $v_t^i = 1$ and incremented otherwise. Multi-step blackouts expose the GRU to consecutive decisions without current object geometry, encouraging the policy to exploit temporal context together with proprioception and nominal-plan context. We denote the resulting structured observation by $\tilde{o}_t = \Omega(s_t)$. Algorithm 1 summarizes the complete student-training procedure.

Behavior cloning. The initial dataset $\mathcal{D}_0$ is collected from teacher-controlled trajectories. At every visited state, the teacher acts from privileged state information, while the student input is constructed from $\tilde{o}_t$, the fixed-rollout context z_t, end-effector features c_t, and the previous action. Each recurrent sequence therefore pairs the information available to the student with the teacher's categorical action distribution p_t^T. Training on $\mathcal{D}_0$ yields the behavior-cloned policy π_S^0.

DAgger. Behavior cloning exposes the student only to states visited under teacher control. At DAgger round k, the current student π_S^{k-1} instead controls execution, while the frozen teacher labels every student-visited state with p_t^T. The student's selected primitive remains the executed action; the teacher output is used only as the supervised target. The teacher recurrent state h_t^T is advanced along the same student-induced state sequence. Let $\mathcal{B}_k$ denote the recurrent sequences collected at round k. We aggregate $\mathcal{D}_k = \mathcal{D}_{k-1} \cup \mathcal{B}_k$ and fit the next student on $\mathcal{D}_k$. Thus, DAgger preserves the same privileged supervision used for behavior cloning while extending it to states induced by the student's own actions. Across behavior cloning and DAgger, the student minimizes a recovery-weighted forward-KL objective,

$$\mathcal{L}_{\mathrm{IL}} = \frac{\sum_t m_t w_t D_{\mathrm{KL}}\left(p_t^T \| p_t^S\right)}{\sum_t m_t w_t}, \tag{2}$$

where the sums extend over the sampled recurrent sequences. Here, p_t^T and p_t^S are the teacher and student categorical distributions over the 16 motion primitives. The mask m_t selects valid action-supervision steps, excluding padding, terminal observations, invalid states, and states already satisfying the graspability criterion; occluded observations remain eligible for supervision. The recovery weight $w_t \in [1, 3]$ increases the contribution of decisions for which recovery margin is limited, based on progress toward the nominal horizon, remaining step slack, and remaining travel slack. These quantities depend only on the recorded execution and nominal rollout, not on future success labels. Normalization by $\sum_t m_t w_t$ prevents the weighting magnitude from rescaling the overall loss. Matching the full teacher distribution preserves relative preferences among corrective actions beyond the argmax.

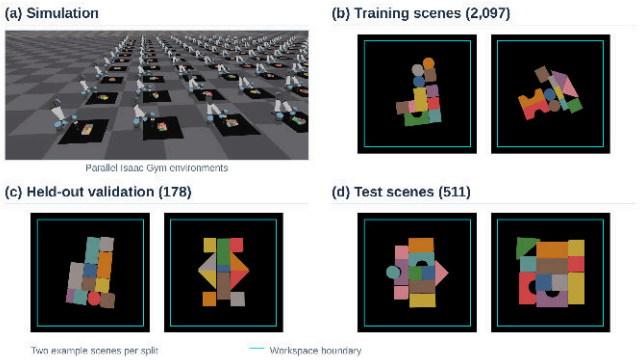

Fig. 4. **Simulation setup and dataset.** (a) Parallel Isaac Gym environments. (b)-(d) Representative training, validation, and test scenes (2,097/178/511); validation is used for model selection and the disjoint test set only for final evaluation.

V. Experiments

We evaluate four questions: whether plan-conditioned closed-loop execution improves over nominal replay and existing retrieval methods; whether supervision on student-visited states provides benefit beyond additional expert data; how recurrent memory and the plan-context horizon affect performance under partial observation; and whether these advantages transfer to physical retrieval without repeated complete-scene reacquisition.

A. Experimental Protocol

Environment and scenes. Experiments use Isaac Gym [23] with eleven movable objects in a $0.448 \times 0.448\,\mathrm{m}$ workspace and the 16 primitives in Fig. 2. We use 2,097 training, 178 held-out validation, and 511 independently generated test scenes; validation is used for model selection and the test set only for final evaluation (Fig. 4).

Planning–execution mismatch. For each episode, $\bar{\tau}$ is generated from the initial planning scene, after which every object in the corresponding execution scene is perturbed independently by up to $\pm 15\,\mathrm{mm}$ in each planar direction and $\pm 10^\circ$ in yaw. Thus, the nominal rollout predicts a nearby scene rather than revealing the executed state. Paired policies receive identical perturbations.

Partial-observation condition. Manipulator self-occlusion follows the geometric model in Sec. IV-D. During training, otherwise visible object tokens are additionally dropped with probability $p_{\mathrm{drop}} = 10\%$, and scheduled blackouts remove all object geometry for 5 consecutive decisions. Visible objects use exact simulator geometry so that these experiments

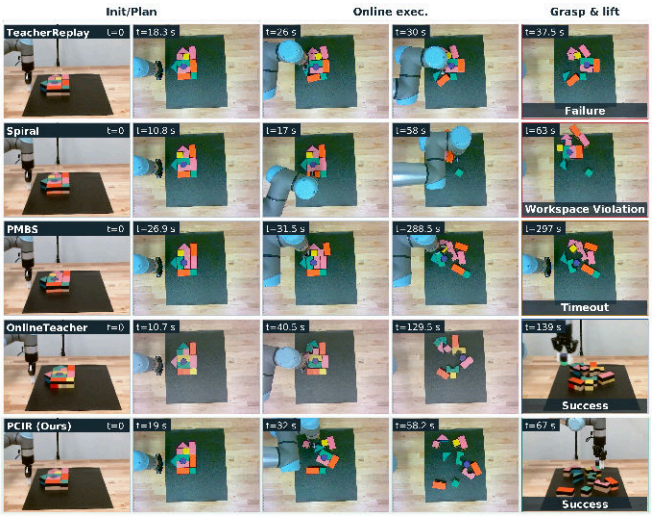

Fig. 5. **Qualitative real-robot comparison.** Representative outcomes: TEACHER REPLAY ends ungraspable, Spiral causes a workspace exit, PMBS reaches its 15-push limit, ONLINE TEACHER succeeds after repeated complete-scene reacquisition, and TRACE succeeds without online arm retraction. Timestamps are in seconds.

isolate missing observations from pose-estimation error. At test time, the same self-occlusion model is active, together with $p_{\text{drop}} = 10\%$ token dropout at every decision and one 5-decision blackout whose onset is sampled uniformly over the 120-decision horizon. Paired policies receive identical corruption draws.

Training. The teacher is trained with PPO using Eq. (1) and then frozen. The student is initialized by behavior cloning and refined with DAgger (Sec. IV-D). Each student fit uses 10,000 optimizer updates with minibatches of 32 recurrent sequences and minimizes Eq. (2). Checkpoints and the DAgger round are selected using the 178 held-out validation scenes. The 511 test scenes are not used for model selection.

Metrics and statistics. An episode succeeds when predicted graspability exceeds 0.9 while all objects remain inside the workspace. We separately report out-of-workspace (OOW) and *Budget* failures, the latter indicating that a method reached its execution limit before success. TRACE and TRACE-BC use the scene-specific teacher-relative step/travel budget, without additional travel tolerance in simulation; TEACHER REPLAY executes the recorded nominal sequence to completion. The remaining methods use their predefined timeout or action limits. Success confidence intervals use 2,000 stratified scene-bootstrap resamples; paired comparisons use identical scenes and initial perturbations.

Reference policies and baselines. TEACHER REPLAY executes the teacher-generated nominal sequence without online object feedback. TRACE-BC uses the same recurrent plan-conditioned architecture as TRACE but is trained only by behavior cloning. ONLINE TEACHER executes the privileged teacher from complete state. We additionally compare against PMBS and Serial MCTS [6], which require complete scene state, and the target-centric Spiral and Straight-Line heuristics. The information available to each method is summarized in Table I.

TABLE I

SIMULATION COMPARISON ON THE 511-SCENE TEST SET. OOW DENOTES A WORKSPACE VIOLATION; BUDGET DENOTES THE METHOD-SPECIFIC EXECUTION LIMIT. BRACKETS ARE 95% STRATIFIED SCENE-BOOTSTRAP CIS FROM 2,000 RESAMPLES.

Method	Online state	Success (%)	95% CI	OOW (%)	Budget (%)
Nominal-rollout methods					
TEACHER REPLAY	None	43.4	[39.1,48.0]	6.3	50.3
TRACE-BC	Partial	66.7	[63.7,69.8]	15.3	18.0
TRACE	Partial	**90.7**	[88.9,92.6]	7.2	2.1
Planning and heuristic baselines					
PMBS	Complete	88.1	[85.1,90.8]	—	11.9
Serial MCTS	Complete	44.0	[40.1,48.1]	—	56.0
Spiral	Target position	80.2	[76.9,83.8]	19.6	0.2
Straight Line	Target position	0.4	[0.0,1.0]	0.0	99.6
Privileged reference					
ONLINE TEACHER	Complete	**96.7**	[95.1,98.0]	3.3	0.0

TABLE II

EFFECT OF STUDENT-STATE SUPERVISION ON THE 511-SCENE TEST SET. RESULTS AVERAGE THREE SEEDS; ALL FITS USE 10,000 UPDATES. Δ IS RELATIVE TO THE PRECEDING ROUND IN THE AGGREGATION BLOCK AND EXPERT BC IN THE MATCHED-BUDGET BLOCKS; BRACKETS ARE PAIRED 95% CIS.

Training data	Labels	Success (%)	Δ (pp, 95% CI)
DAgger aggregation			
Expert BC	26,373	66.7	–
DAgger R1	57,125	87.1	$+20.4$ $[17.5, 23.2]$
DAgger R2	84,429	89.0	$+2.0$ $[-0.1, 4.0]$
TRACE (DAgger R3)	111,555	**90.7**	$+1.7$ $[-0.2, 3.5]$
Matched label budget			
Expert BC	26,373	66.7	–
Student-state BC	26,373	**87.8**	$\mathbf{+21.1}$ $[18.3, 23.9]$
Matched full-data budget			
Expert BC, full data	111,555	78.8	–
TRACE (DAgger R3)	111,555	**90.7**	$\mathbf{+11.9}$ $[9.5, 14.4]$

B. Plan-Conditioned Execution and Reference Policies

We compare plan-conditioned closed-loop execution against open-loop replay of the same nominal rollout and other retrieval references. Table I compares TRACE with its internal references, planning-based baselines, target-centric heuristics, and the privileged ONLINE TEACHER.

TEACHER REPLAY succeeds in only 43.4% of scenes, whereas TRACE reaches 90.7% using the same nominal prediction together with partial closed-loop observations. TRACE approaches ONLINE TEACHER (96.7%) and also exceeds PMBS (88.1%) despite using less online state information. The lower TRACE-BC performance (66.7%) motivates the student-state supervision study in Sec. V-C.

TABLE III

PLAN-CONTEXT HORIZON AND RECURRENT-MEMORY ABLATIONS ON THE 511-SCENE TEST SET (THREE SEEDS). Δ AND 95% CIS ARE PAIRED AGAINST TRACE ($K = 4$); BUDGET IS THE TEACHER-RELATIVE STEP/TRAVEL LIMIT. STEPS IS THE MEAN OVER SUCCESSFUL EPISODES.

Variant	Success (%)	Δ vs. TRACE (pp, 95% CI)	OOW (%)	Budget (%)	Steps
TRACE ($K = 4$)	**90.7**	—	7.2	2.1	14.8
w/o GRU	85.8	$\mathbf{-4.9}$ $[-6.5, -2.4]$	8.0	**6.2**	15.0
$K = 1$	88.5	-2.2 $[-3.7, +0.2]$	8.4	3.1	14.6
$K = 2$	89.9	-0.8 $[-2.1, +1.4]$	7.8	2.3	14.7
$K = 8$	89.3	-1.4 $[-2.7, +0.8]$	7.7	3.0	14.7
Full plan	90.0	-0.7 $[-2.0, +1.6]$	7.3	2.6	14.4

C. Effect of Student-State Supervision

We test whether DAgger helps because supervision is collected on student-visited states rather than simply because aggregation provides more labels. Expert BC uses 2,097 teacher-controlled trajectories containing 26,373 valid labels; three DAgger rounds then progressively add teacher labels on states induced by the current student. Most of the gain appears after the first aggregation round, with later rounds giving smaller improvements. More importantly, the matched-label control improves success from 66.7% to 87.8% at the same 26,373-label and optimization budgets. Even after increasing Expert BC to the full 111,555-label budget, it reaches only 78.8%, compared with 90.7% for DAgger R3. Together, these controls show that student-state coverage, rather than additional supervision alone, accounts for most of the DAgger advantage.

D. Plan-Context Horizon and Recurrent Memory

Among plan-conditioned policies, we ablate recurrent memory and the nominal-plan context horizon while holding the remaining protocol fixed. We do not treat a plan-free controller as a matched ablation because removing $\bar{\tau}$ also removes the scene-specific execution horizon and therefore introduces a separate termination-policy design choice. Removing the GRU primarily increases execution-budget exhaustion, while the OOW rate remains nearly unchanged, supporting the role of recurrence in carrying useful temporal context through missing observations. In contrast, performance is similar across the tested plan horizons, and the full nominal rollout provides no observed benefit over $K = 4$. We therefore retain $K = 4$ as a compact local context.

E. Real-Robot Evaluation

We evaluate whether the simulation-trained policy transfers to physical retrieval and whether partial-observation execution reduces the overhead of obtaining complete scene state. The system uses a UR5e, Robotiq parallel-jaw gripper, and fixed Intel RealSense D455 (Fig. 1). A clean initial observation initializes the digital twin and generates $\bar{\tau}$ once before manipulation.

Protocol. The benchmark contains 20 unseen clutter arrangements with two trials per scene and method. Success requires the target to become graspable, be successfully grasped and lifted, and remain within workspace constraints. We report success, OOW failures, initialization/planning time, online execution time, grasp-and-lift time, total time, and deliberate arm retractions used for visual-state reacquisition. For PMBS and ONLINE TEACHER, these retractions restore complete scene state; for Spiral, they are required only when the target pose is occluded. The fixed camera continues to provide partial observations throughout execution without requiring arm motion. In contrast, methods that require the *complete* scene state must retract the arm to restore visibility; we count each such arm-retraction event as a complete-scene reacquisition. The final arm retraction used by TRACE for terminal graspability evaluation is not counted as an online reacquisition. The TRACE policy has no learned stop action; pushing terminates at the scene-specific teacher-relative step/travel budget with an additional 5 cm travel tolerance. Other methods use their predefined action or timeout limits. Table IV reports success and OOW; remaining unsuccessful trials terminate at the corresponding method-specific execution limit.

References. TEACHER REPLAY uses only the initial nominal plan. Spiral requires the target pose at each decision: while the target remains visible, this pose is updated directly from the fixed camera without arm retraction; when the target is occluded, the arm retracts to reacquire it. PMBS and ONLINE TEACHER require complete scene state and therefore retract the arm whenever full visibility must be restored.

Results. Figure 5 illustrates representative execution behaviors and failure modes across the evaluated methods. TRACE reaches 90.0% hardware success without any arm retraction for state reacquisition during pushing, compared with 47.5% for TEACHER REPLAY. Spiral has the lowest total time at 44.4 s and requires only 2.8 retraction-based reacquisitions per trial on average, but reaches 77.5% success and incurs OOW failures in 17.5% of trials. Spiral shows that target-only feedback can be obtained with relatively little reacquisition overhead, but this heuristic remains less reliable than TRACE and incurs substantially more OOW failures. ONLINE TEACHER reaches 95.0% success, but requires 16.8 arm retractions per trial to recover complete scene state and 192.7 s total execution time, versus 67.3 s for TRACE. PMBS similarly requires complete-state reacquisition and reaches 85.0% success with 141.1 s total time. Overall, TRACE retains much of the reliability of privileged closed-loop execution while avoiding the repeated arm-withdrawal overhead needed to restore complete scene visibility.

VI. CONCLUSIONS

We presented TRACE, which uses a one-time privileged nominal rollout as predictive context for recurrent retrieval under self-occlusion. The design retains much of privileged closed-loop performance while avoiding repeated complete-scene reacquisition; on hardware, the rollout also provides a scene-specific execution horizon without a learned stop action. Nominal plans thus support feedback under partial

TABLE IV

REAL-ROBOT EVALUATION ON 20 SCENES WITH TWO TRIALS EACH. OOW DENOTES A WORKSPACE VIOLATION. ARM RETRACTIONS COUNT VISUAL-STATE REACQUISITION; TRACE'S FINAL GRASPABILITY-CHECK RETRACTION IS EXCLUDED. OTHER UNSUCCESSFUL TRIALS REACHED THE METHOD-SPECIFIC EXECUTION LIMIT.

Method	Execution	Online information	Success (%)	OOW (%)	Init./plan (s)	Online exec. (s)	Grasp+lift (s)	Total (s)	Arm retractions
TEACHER REPLAY	Open loop	Initial plan only	47.5	**0.0**	27.5	11.6	10.4	49.5	**0.0**
Spiral	Closed loop	Target pose	77.5	17.5	**6.3**	29.2	8.9	**44.4**	2.8
PMBS	Closed loop	Complete scene	85.0	10.0	124.9	**7.4**	**8.8**	141.1	3.9
ONLINE TEACHER	Closed loop	Complete scene	**95.0**	**0.0**	10.4	173.3	9.0	192.7	16.8
TRACE (Ours)	Closed loop	Partial scene + $\bar{\tau}$	90.0	**0.0**	30.4	24.8	12.1	67.3	**0.0**

observability. Matched-label comparisons show that supervision on student-visited states improves retrieval reliability beyond adding teacher demonstrations alone. Architecture ablations further support combining recurrent memory with local rollout context to respond to deviations during contact. The current evidence is limited to planar clutter with known object footprints and an initially visible scene. Uncertain initial estimates and unfamiliar geometries require further evaluation. An important next step is detecting when the nominal rollout becomes uninformative and selectively acquiring a new view, balancing sensing overhead against the risk of continued execution. Our implementation uses discrete motion primitives and a teacher-relative budget; future work will study continuous end-effector actions, learned termination, and object retrieval from non-planar (3D) clutter.

ACKNOWLEDGEMENT

OpenAI's ChatGPT was used to assist with improving the text, and presentation of the paper.

REFERENCES

[1] National Institute of Standards and Technology, "Manufacturing track: Task 1—board assembly/disassembly," Robotic Grasping and Manipulation Competition, IROS, 2017. [Online]. Available: https://www.nist.gov/document/assemblydisassembly-task

[2] K.-T. Yu, N. Fazeli, N. Chavan-Dafle, O. Taylor, E. Donlon, G. Diaz Lankenau, and A. Rodriguez, "A summary of team MIT's approach to the Amazon Picking Challenge 2015," *arXiv preprint arXiv:1604.03639*, 2016.

[3] C.-H. King, T. L. Chen, Z. Fan, J. D. Glass, and C. C. Kemp, "Dusty: An assistive mobile manipulator that retrieves dropped objects for people with motor impairments," *Disability and Rehabilitation: Assistive Technology*, vol. 7, no. 2, pp. 168–179, 2012.

[4] B. Huang, S. D. Han, J. Yu, and A. Boularias, "Visual foresight trees for object retrieval from clutter with nonprehensile rearrangement," *IEEE Robotics and Automation Letters*, vol. 7, no. 1, pp. 231–238, 2022.

[5] B. Huang, T. Guo, A. Boularias, and J. Yu, "Interleaving Monte Carlo tree search and self-supervised learning for object retrieval in clutter," in *2022 IEEE International Conference on Robotics and Automation (ICRA)*, 2022, pp. 625–632.

[6] B. Huang, A. Boularias, and J. Yu, "Parallel Monte Carlo tree search with batched rigid-body simulations for speeding up long-horizon episodic robot planning," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 1153–1160.

[7] A. Kurenkov, J. Taglic, R. Kulkarni, M. Dominguez-Kuhne, A. Garg, R. Martín-Martín, and S. Savarese, "Visuomotor mechanical search: Learning to retrieve target objects in clutter," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 8408–8414.

[8] M. Kiatos, L. Koutras, I. Sarantopoulos, and Z. Doulgeri, "Learning a pre-grasp manipulation policy to effectively retrieve a target in dense clutter," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024, pp. 7543–7549.

[9] S. Ross, G. J. Gordon, and J. A. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, vol. 15. PMLR, 2011, pp. 627–635.

[10] A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, and T. Funkhouser, "Learning synergies between pushing and grasping with self-supervised deep reinforcement learning," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4238–4245.

[11] K. Xu, H. Yu, Q. Lai, Y. Wang, and R. Xiong, "Efficient learning of goal-oriented push-grasping synergy in clutter," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6337–6344, 2021.

[12] L. Berscheid, P. Meißner, and T. Kröger, "Robot learning of shifting objects for grasping in cluttered environments," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 612–618.

[13] M. Danielczuk, A. Kurenkov, A. Balakrishna, M. Matl, D. Wang, R. Martín-Martín, A. Garg, S. Savarese, and K. Goldberg, "Mechanical search: Multi-step retrieval of a target object occluded by clutter," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 1614–1621.

[14] Y. Xiao, S. Katt, A. ten Pas, S. Chen, and C. Amato, "Online planning for target object search in clutter under partial observability," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 8241–8247.

[15] D. Chen, B. Zhou, V. Koltun, and P. Krähenbühl, "Learning by cheating," in *Proceedings of the Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, vol. 100. PMLR, 2020, pp. 66–75.

[16] M. Mosbach and S. Behnke, "Prompt-responsive object retrieval with memory-augmented student-teacher learning," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 2025, pp. 4551–4557.

[17] T. He, Z. Wang, H. Xue, Q. Ben, Z. Luo, W. Xiao, Y. Yuan, X. Da, F. Castañeda, S. Sastry, C. Liu, G. Shi, L. Fan, and Y. Zhu, "VIRAL: Visual sim-to-real at scale for humanoid loco-manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026, pp. 13 430–13 441.

[18] Y. Kim, N. Chin, A. Vasudev, and S. Choudhury, "Distilling realizable students from unrealizable teachers," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025, pp. 8103–8110.

[19] T. Johannink, S. Bahl, A. Nair, J. Luo, A. Kumar, M. Loskyll, J. A. Ojea, E. Solowjow, and S. Levine, "Residual reinforcement learning for robot control," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 6023–6029.

[20] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Póczos, R. Salakhutdinov, and A. J. Smola, "Deep sets," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 3391–3401.

[21] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[22] A. Y. Ng, D. Harada, and S. Russell, "Policy invariance under reward transformations: Theory and application to reward shaping," in *Proceedings of the 16th International Conference on Machine Learning (ICML)*, 1999, pp. 278–287.

[23] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, and G. State, "Isaac Gym: High performance GPU-based physics simulation for robot learning," *arXiv preprint arXiv:2108.10470*, 2021.