# TWINSIGHT: GUIDING HUMAN TEACHING AND POLICY ADAPTATION WITH LEARNER FORECASTS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Successful demonstrations can still leave a robot policy's weaknesses unaddressed. Without seeing how the learner would continue, human teachers may miss useful corrections, and data selection cannot recover examples that were never demonstrated. We introduce TWINSIGHT, a framework that connects human teaching and policy adaptation through learner forecasts. During human control, a learner-specific predictive model forecasts end-effector motion beyond the current action chunk, helping teachers anticipate later difficulties before committing to an approach. Persistent teaching bubbles record local differences between forecast and demonstrated motion and link them to coherent human action segments for subsequent selection and training. Across six simulated and three real-robot manipulation tasks, TWINSIGHT achieves mean success rates of 74.0% and 80.7%, exceeding the strongest evaluated baselines by 5.3 and 6.7 percentage points, respectively. On shared simulation datasets, segment selection achieves similar mean success to training on all retained data with 30.7% fewer training steps.

## 1 INTRODUCTION

End-to-end robot control has advanced rapidly. In vision-language-action (VLA) policies, camera observations and language instructions are mapped to robot actions (Kim et al., 2024). However, substantial performance drops have been observed in representative VLA policies under changes in viewpoint, object configuration or robot initial state (Fei et al., 2026). Robustness can be improved with broader training-data coverage, but demonstrations are costly to collect across diverse scenes and tasks (Khazatsky et al., 2024). Therefore, pretrained policies are often adapted to new deployment settings through fine-tuning with additional human demonstrations (Kim et al., 2025; Ye et al., 2026).

A task may be demonstrated successfully while the difficulties of a particular learner policy remain unaddressed (Gandhi et al., 2022; Chen et al., 2025a). If these difficulties are made visible, teaching can be directed toward specific improvements, potentially reducing the effort required for adaptation. Human teachers have been guided using information about learner behavior (Gandhi et al., 2022; Fang et al., 2026), and training data have been selected for their expected contribution to policy improvement (Dass et al., 2026; Agia et al., 2025). But important teaching opportunities may still be missed when later consequences of the learner's behavior are not apparent. Once collection is complete, missing examples cannot be recovered through data selection alone. *How can limited human teaching effort be timely focused on the decisions that matter most for policy adaptation?*

The choice of what to demonstrate can be informed by what the learner would do in the same situation (Gandhi et al., 2022). If its likely continuation could be seen in advance, an anticipated difficulty could be considered before the human teacher commits to an approach. Evidence used during teaching could then be retained with the demonstrated actions for adaptation. But three challenges remain: **i)** In previews of the current short-horizon action sequence, such as those used in RoboPocket (Fang et al., 2026), later task consequences may remain hidden while teaching choices are being made. **ii)** An approach decision may precede the difficulty it could prevent, making the corresponding human actions difficult to retain with sufficient preparation and continuation. **iii)** A segment's relevance to an anticipated difficulty does not establish its value for adaptation (Dass et al., 2026; Agia et al., 2025).

To address these challenges, TWINSIGHT is proposed to guide human teaching and policy adaptation through forecasts of learner behavior. From states visited under human control, future end-effector



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

(EE) motion beyond the current action sequence and episode failure are predicted with a world model. Possible later difficulties can be considered from these outputs while teaching choices remain open. Local comparisons with observed robot motion under human control are retained as visual markers called *teaching bubbles*. Coherent segments are recovered from their temporal records and ranked by the mean discrepancy of their contributing comparison intervals. Longer forecasts inform teaching choices, while recorded local comparisons supply evidence for segment selection. Across six simulated tasks, TWINSIGHT achieves 74.0% mean success versus 68.7% for HG-DAgger, and leads on all three real-robot tasks. Segment selection achieves similar mean success with 30.7% fewer training steps than training on all retained data. Our contributions are:

- A framework connects forecast-guided human teaching to the selection of human action segments for policy adaptation.
- A learner-specific world model predicts motion beyond the current action chunk and episode failure to guide teaching.
- Teaching bubbles preserve local motion comparisons and temporal context for persistent guidance and coherent segment selection.

## 2 RELATED WORK

**Human teaching and interactive learning.** Human supervision can be focused on learner needs through selective involvement and guidance during demonstration. In DAgger, HG-DAgger, and ThriftyDAgger, collection is coupled to learner execution through expert labeling or selective intervention (Ross et al., 2011; Kelly et al., 2019; Hoque et al., 2021). Human effort can also be reduced by delegating familiar portions to an assistive policy, as in PATO (Dass et al., 2023). Alongside the allocation of control, guidance has been studied to inform what a human teacher demonstrates. Robot motion feasibility is assessed through ARCap (Chen et al., 2025c), and consistency with learned behavior is assessed through compatibility guidance (Gandhi et al., 2022). In RoboPocket, the learner's predicted trajectory is projected into the scene to guide collection, while new demonstrations are incorporated through online fine-tuning (Fang et al., 2026). These previews can expose weaknesses within the current action horizon.

**Demonstration selection.** DemInf assesses data quality independently of the learner through state diversity and action predictability (Hejna et al., 2025), while S2I and STRAP select or retrieve segments (Chen et al., 2025b; Memmel et al., 2025). Demo-SCORE uses learner experience (Chen et al., 2025a). DataMIL selects demonstrations or sub-trajectories by estimating their effects on held-out imitation loss. CUPID estimates demonstration influence on policy return using evaluation rollouts (Dass et al., 2026; Agia et al., 2025). Gradient alignment with a reference objective is another selection criterion (Killamsetty et al., 2021). Selection can prioritise available examples, but cannot recover actions that were never demonstrated. TWINSIGHT links collection guidance to segment selection through teaching-bubble records, using discrepancies between forecast and demonstrated motion as a heuristic rather than directly estimating training utility.

**Forecasting learner behavior.** World models have been used to examine possible continuations before execution, generating training experience in UniSim (Yang et al., 2024) and supporting search for successful plans in Video Language Planning (Du et al., 2024). In WorldGym and Ctrl-World, generated observations are returned to the learner for policy evaluation, with successful synthetic trajectories also used for adaptation in Ctrl-World (Quevedo et al., 2026; Guo et al., 2026). Forecasts are also used to request assistance before failure in model-based runtime monitoring and Sirius-Fleet (Liu et al., 2024a;b). In TWINSIGHT, motion and episode failure are predicted directly from an encountered state and the current action sequence. These forecasts are used for human teaching, with local motion comparisons retained for subsequent selection.

## 3 TWINSIGHT

Teaching and selection require different temporal views of the same encounter. A teacher may need to anticipate consequences beyond the current action chunk before choosing an approach. Selecting training data requires linking a learner–human difference to the actions actually demonstrated. TWINSIGHT uses longer forecasts for the first purpose and short, aligned motion comparisons for

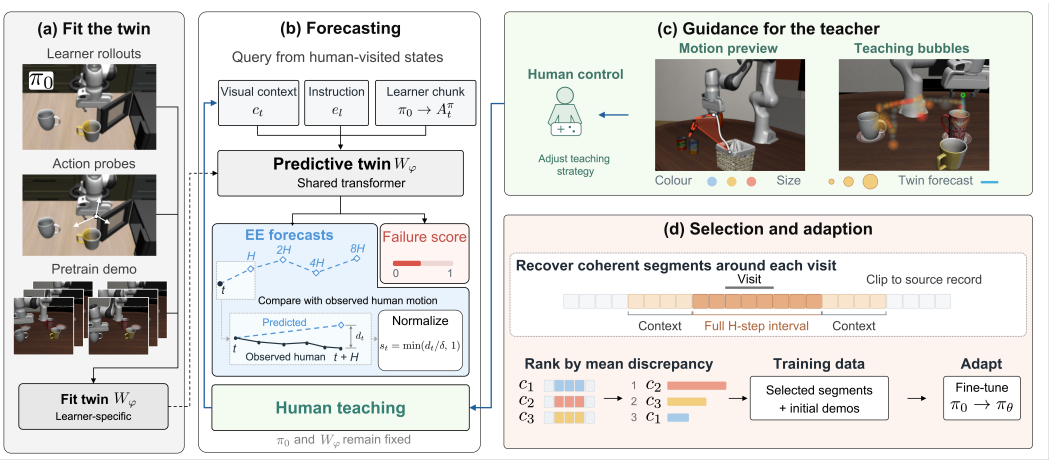



Figure 1: **TWINSIGHT overview.** (a) Learner executions train the twin, supplemented by one-chunk alternative-action and human records. (b) The twin forecasts learner EE positions and episode failure during human control. (c) Teaching bubbles retain normalized endpoint discrepancies from completed one-chunk comparisons. (d) Segments are ranked by mean interval discrepancy and selected under an action-step budget for adaptation with initial demonstrations.

the second (Figure 1). A learner-specific world model, the *predictive twin*, forecasts the learner's continuation while the human controls the robot. *Teaching bubbles* retain local endpoint discrepancies with their source actions, connecting evidence available during teaching to subsequent segment selection.

### 3.1 A PREDICTIVE TWIN OF THE LEARNER

An action chunk specifies the learner's current commands, but not how it will respond after observing their consequences. Forecasting beyond that chunk therefore requires accounting for both scene interaction and subsequent replanning. The predictive twin learns this continuation from executions of the pretrained learner $\pi_0$, whose parameters remain fixed during collection.

**Predicting learner behavior.** At demonstration step $t$, the learner maps observation $o_t$ and task instruction $\ell$ to an action chunk $A_t^\pi = \pi_0(o_t, \ell)$ containing $H$ successive robot commands. The forecast assumes that the learner executes each full chunk before observing and replanning.

Let $z_t$ denote visual features extracted from $o_t$ by a frozen encoder. EE status is represented here by the robot EE's three-dimensional position $p_t$ in a shared workspace coordinate frame. The twin $W_\phi$, with parameters $\phi$, is conditioned on visual context $c_t$ from current and recent features, an action chunk $A_t$, and instruction embedding $e_\ell$. During guided collection, $A_t = A_t^\pi$. Changes are predicted at horizons $h \in \mathcal{H}$, measured in forecast steps at the same control interval as the demonstration. The set $\mathcal{H}$ contains the one-chunk horizon $H$ and selected longer horizons:

$$W_\phi(c_t, A_t, e_\ell) = \left( \left\{ \left( \widehat{\Delta z}_{t,h}, \widehat{\Delta p}_{t,h} \right) \right\}_{h \in \mathcal{H}}, r_t \right). \tag{1}$$

For each horizon, the twin predicts visual-feature change $\widehat{\Delta z}_{t,h}$ and EE displacement $\widehat{\Delta p}_{t,h}$, giving the position forecast $\hat{p}_{t+h|t} = p_t + \widehat{\Delta p}_{t,h}$. The episode-failure score $r_t \in [0, 1]$ is trained on learner outcomes under the preparation episodes' stopping criteria. It provides an outcome cue, not the time or location of a failure. These criteria are distinct from the final policy-evaluation limits (Appendix A.4).

The twin predicts all horizons directly in one evaluation, without recursively generating future observations or actions. This gives the teacher a compact view of the learner's anticipated motion rather than a simulated scene rollout. Only EE positions and the episode-failure score are displayed. Visual-feature changes provide auxiliary training supervision and are not decoded into future images.

**Training the twin.** The supervision changes with horizon. Over one chunk, the target concerns the consequences of the supplied commands. Learner executions can therefore be supplemented

by alternative-action trials for visual prediction and existing human records for visual and position prediction. Beyond that chunk, the target also depends on which actions $\pi_0$ chooses next. All longer-horizon targets and failure labels therefore come exclusively from separate learner executions under the same full-chunk execution and replanning schedule. Each input action chunk is paired with its recorded transition. Before guided collection, the twin is fitted with the objective

$$\mathcal{L}_{\text{pred}} = \sum_{h \in \mathcal{H}} \left( \mathcal{L}_h^{\text{vis}} + \lambda_p \mathcal{L}_h^{\text{pos}} \right) + \lambda_r \mathcal{L}^{\text{fail}}. \tag{2}$$

For horizon $h$, $\mathcal{L}_h^{\text{vis}}$ and $\mathcal{L}_h^{\text{pos}}$ are squared-error losses against observed changes $z_{t+h} - z_t$ and $p_{t+h} - p_t$, averaged over available targets. The binary cross-entropy $\mathcal{L}^{\text{fail}}$ is computed against learner episode outcomes, with failure encoded as one and success as zero. Position and failure supervision are balanced against visual prediction by nonnegative weights $\lambda_p$ and $\lambda_r$.

An auxiliary head reconstructs the supplied learner action chunk as $\hat{A}_t$ from the shared representation. It has access to the input chunk and is used only during training. Its squared-error loss $\mathcal{L}_{\text{act}}$ is evaluated on learner executions and added with weight $\lambda_a \geq 0$, giving $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{pred}} + \lambda_a \mathcal{L}_{\text{act}}$. Since human control can visit states outside the learner's own trajectories, we assess motion forecasts against learner continuations from human-visited states (Appendix B.2).

### 3.2 Teaching bubbles from aligned motion forecasts

Longer forecasts can inform an approach before its consequences unfold, but a later endpoint difference also incorporates subsequent choices by both controllers. For selection, we instead compare endpoints over one action chunk from a shared starting state. This anchors each discrepancy to a specific interval of demonstrated actions, without treating it as evidence of which action caused or prevented a failure.

Comparisons are formed over successive nonoverlapping intervals of $H$ demonstration steps. At an interval's start $t$, the forecast $\hat{p}_{t+H|t}$ is retained, conditioned on the learner's proposed actions while control is retained by the human teacher. Once the EE position under human control, $p_{t+H}^{\text{hum}}$, is observed, discrepancy is computed as

$$d_t = \left\| \hat{p}_{t+H|t} - p_{t+H}^{\text{hum}} \right\|_2, \qquad s_t = \min\left( \frac{d_t}{\delta}, 1 \right), \tag{3}$$

where $d_t$ is the Euclidean endpoint distance and $\delta > 0$ is a fixed distance scale in the same units. The normalized score $s_t \in [0, 1]$ is zero for matching positions and saturates at distance $\delta$. It becomes available after both inference and the human interval are complete, and is retained with that interval's starting state and actions. Thus, the comparison is retrospective even though the displayed forecasts are prospective. The observed human continuation may itself have been influenced by earlier guidance.

**Persistent teaching guidance.** A live forecast changes as the teacher moves, so earlier comparisons would otherwise disappear from view. Teaching bubbles preserve them spatially while retaining the temporal records needed to recover training segments (Figure 2). Each interval's score is assigned uniformly along its recorded EE path. Path samples are assigned to the closest bubble whose bounded spatial neighborhood contains them, with a new bubble created otherwise. Bubble centres are updated by duration-weighted averaging. Let $\mathcal{B}$ denote the collection of bubbles, each associated with one source demonstration. For bubble $b \in \mathcal{B}$, $w_{tb} \geq 0$ denotes the duration assigned from the interval starting at $t$, with assignments across bubbles summing to that interval's duration. Accumulated and mean discrepancy are computed over completed, scored intervals as

$$M_b = \sum_t w_{tb} s_t, \qquad \bar{s}_b = \frac{M_b}{\sum_t w_{tb}}. \tag{4}$$

The sums cover the bubble's source demonstration, and the mean is defined for positive total assigned duration. Each path contribution retains its timestamp and originating comparison interval.

Bubble colour is determined by $\bar{s}_b$, while size is determined by $M_b$ relative to other regions, distinguishing disagreement magnitude from its accumulation through duration or revisits. Task relevance is interpreted by the human teacher using the longer forecasts and episode-failure score. For example,

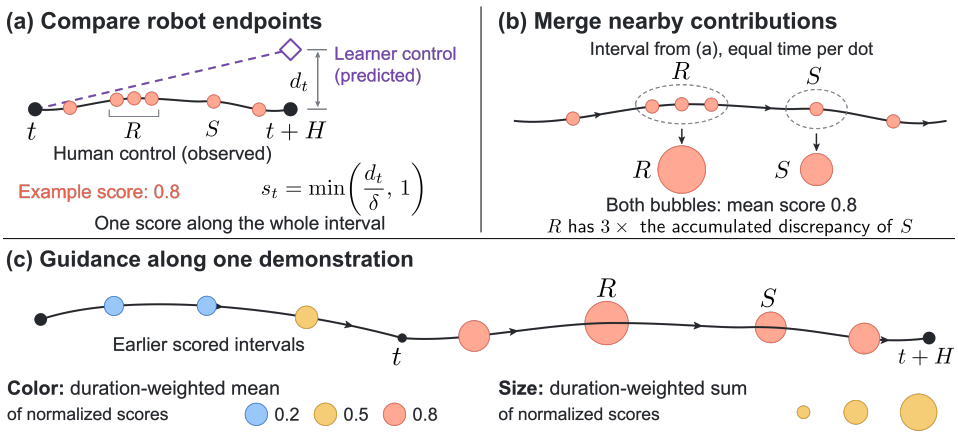



Figure 2: **Teaching bubbles.** (a) Forecast learner and observed human EE endpoints are compared over the same interval from a shared state. Normalized discrepancy is assigned along the human path. (b) Nearby contributions merge. (c) Successive comparisons persist. Colour encodes duration-weighted mean discrepancy, size its accumulation. Values and geometry are illustrative.

forecasting later EE positions near an obstacle could prompt a wider demonstrated approach, retained with its local discrepancy for subsequent segment selection. Discrepancy may also reflect forecast error, execution speed, or valid alternative strategies, while matching endpoints can conceal differences in path, orientation, or grasp.

### 3.3 Selecting human action segments for training

A spatial discrepancy does not by itself define a useful training example. The relevant human response can include preparation before the comparison and continuation after it. We therefore recover temporally coherent segments around bubble visits and rank them by mean discrepancy. This ranking is a heuristic for prioritising learner–human differences, not an estimate of a segment's effect on policy improvement.

**Constructing candidate segments.** Within each human-controlled record, a *visit* is a maximal consecutive sequence of path samples assigned to the same bubble. Therefore, returning to a bubble creates a separate visit. To construct a candidate segment $C$, a visit is first expanded to cover every complete $H$-step comparison interval that contributes path samples to it. Fixed windows of preceding and subsequent actions are then added, clipped to the boundaries of the same uninterrupted human-controlled record. These context windows retain the actions leading into and out of the local discrepancy, such as the approach and subsequent motion in the obstacle example above. A visit must include samples from at least one complete comparison interval to form a candidate.

**Prioritising segments.** For a candidate $C$, let $I_C$ denote the set of starting steps of the complete comparison intervals that contribute path samples to its original visit. Using the normalised discrepancy $s_t$ from Section 3.2, the candidate score is

$$q(C) = \frac{1}{|I_C|} \sum_{t \in I_C} s_t. \tag{5}$$

Each comparison interval contributes once. Since every interval spans $H$ steps, averaging avoids giving longer visits a higher score simply because they contain more comparisons. The added context is used for training but does not affect the score.

Candidates with positive scores are considered in descending score order. A candidate is skipped if it overlaps a previously selected segment from the same record or would exceed the remaining budget. The budget $B$ counts unique human action steps in the selected later segments, including context. Initial demonstrations are outside this budget but are included in the total training-data and human-effort accounting.

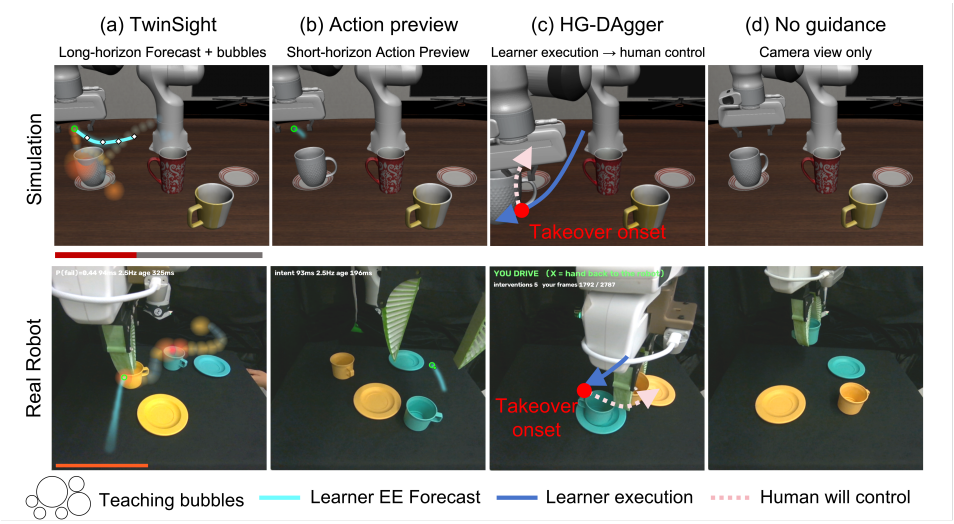



Figure 3: **Teaching guidance in simulation and FR3.**

# 4 EXPERIMENTS

## 4.1 EXPERIMENTAL SETUP

The key experimental settings are summarised below, with further details provided in Appendix A.

**Baselines & Benchmarks.** In simulation, TWINSIGHT is compared with four demonstration-collection baselines. In *No guidance*, demonstrations are provided through ordinary teleoperation without predictive displays. In *Curriculum learning*, component subtasks are demonstrated before the full task. In *HG-DAgger*, human takeovers are collected during learner execution (Kelly et al., 2019). In the *RoboPocket-style Action Preview* condition, the learner's current action sequence is displayed during demonstration (Fang et al., 2026). For real-robot evaluation, No guidance, HG-DAgger, and RoboPocket are used as baselines. Segment-selection comparisons are defined in Appendix B.1.

Evaluations are conducted on six customised manipulation tasks in MuJoCo (Todorov et al., 2012), using scenes and objects from LIBERO-90 (Liu et al., 2023). These tasks cover spatial composition, stacking, multi-object placement, drawer manipulation, loaded transport, and long-horizon composition. Real-robot evaluation is conducted on a Franka Research 3 (FR3) robot for colour-matched placement, object collection followed by container closure, and block stacking. Demonstrations are collected through teleoperation on both platforms, with simulation control configured at 20 Hz. Task instructions and initial-state settings are provided in Appendix A.1 and Appendix A.1.2.

**Experimental protocol.** In simulation, five separate groups of five participants are used, one group for each comparison condition, including TWINSIGHT. All six tasks are completed by every participant in their assigned condition. On the FR3, all three tasks are completed under all four conditions by the same three participants. Each participant retains 20 records per task in simulation and 50 per task and condition on the FR3. A record is a complete demonstration or human action segment. Datasets and adapted learners are kept separate across participants. For HG-DAgger and TWINSIGHT, the first five simulation records and first ten FR3 records are complete demonstrations, followed by human action segments retained under each method's rule. During collection, the learner is held fixed in TWINSIGHT, No guidance, and the simulation action-preview condition. The forecasting model is also held fixed in TWINSIGHT. In TWINSIGHT, teachers review each attempt before retaining data. Retained records follow collection order.

**Learner and training.** The StarVLA framework (StarVLA, 2026), using a Qwen3-VL-4B-Instruct backbone (Qwen, 2025) and an OFT action head, is deployed on both platforms. Each adaptation run is initialised from the LIBERO-4in1 checkpoint and trained using LoRA (Hu et al., 2022). Adaptation uses up to four passes over eligible training examples at batch size eight, with a cap of 6,250 updates. Actual training budgets therefore depend on the retained supervision. Training and inference are

each performed on one NVIDIA GeForce RTX 5090 with 32 GB of memory. The pass budget, example-counting rule, and optimisation settings are specified in Appendix A.3.

**Evaluation metrics.** Each participant's learner is evaluated over 50 runs per task from held-out initial configurations, with identical simulation states across conditions. Success requires full task completion within the execution limit. Per-task rates are averaged across participants, then weighted equally in the overall mean. Teaching efficiency uses human-controlled steps from retained and discarded attempts and collection time, including resets, review, planning, learner execution, and waiting. Training steps count unique supervised human action steps, and eligible training examples and optimisation updates are counted separately. Selection and adaptation time combines offline segment selection and final learner adaptation. Forecast accuracy uses the difference between forecast and observed gripper positions at each horizon (endpoint error). *Appropriate changes* is the percentage of assessed demonstration revisions addressing an identified learner difficulty, and *useful examples* is the percentage of retained later segments that address an evidenced difficulty, meet its local completion criterion, and preserve coherent surrounding actions. Success-rate averaging and effort accounting appear in Appendix A.5, with forecast validation and teaching assessments in Appendices B.2 and B.3.

### 4.2 Experimental results

We evaluate how TwinSight guides demonstrations and whether selecting segments after collection preserves task performance with less training data. Figure 3 compares RoboPocket's short action previews, HG-DAgger's human takeovers, and TwinSight's longer forecasts and persistent teaching bubbles.

Table 1: **Simulation success (%)** after 20 collection records per participant and task.

| Method | Spatial | Stacking | Multi-object | Drawer | Transport | Long horizon | Mean |
|---|---|---|---|---|---|---|---|
| No guidance | 62 | 58 | 66 | 64 | 58 | 40 | 58.0 |
| Curriculum | 60 | 60 | 64 | 62 | 60 | 40 | 57.7 |
| HG-DAgger | 70 | 72 | **76** | 74 | 66 | 54 | 68.7 |
| Action preview | 68 | 64 | 72 | 70 | 62 | 50 | 64.3 |
| TwinSight | **78** | **76** | 74 | **80** | **74** | **62** | **74.0** |

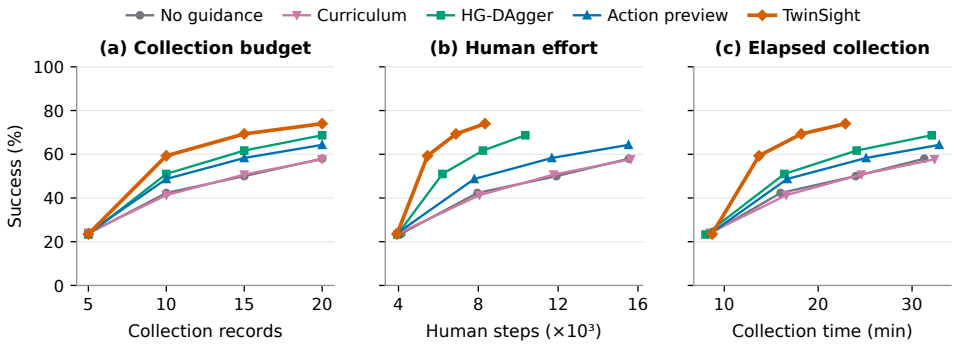



Figure 4: **Simulation success** versus collection records, human-controlled steps and collection time.

**Simulation performance.** TwinSight achieves the highest mean success rate of 74.0%, leading on five of six tasks (Table 1). Its larger gains over Action preview on stacking, loaded transport, and long-horizon composition are consistent with the value of anticipating later learner motion. These tasks require early actions to prepare for subsequent manipulation, so a plausible immediate action may conceal difficulties arising beyond the current chunk. Longer forecasts allow the teacher to consider these difficulties while approach decisions remain open. The learning curves (Figure 4) are also consistent with the benefit of supervision directed at learner-specific difficulties.

**Forecast accuracy and guidance ablation.** In the experiment, TwinSight world-model forecasts are generated at horizons $H$, $2H$, $4H$ and $8H$. With $H = 8$ actions at 20 Hz, these correspond to

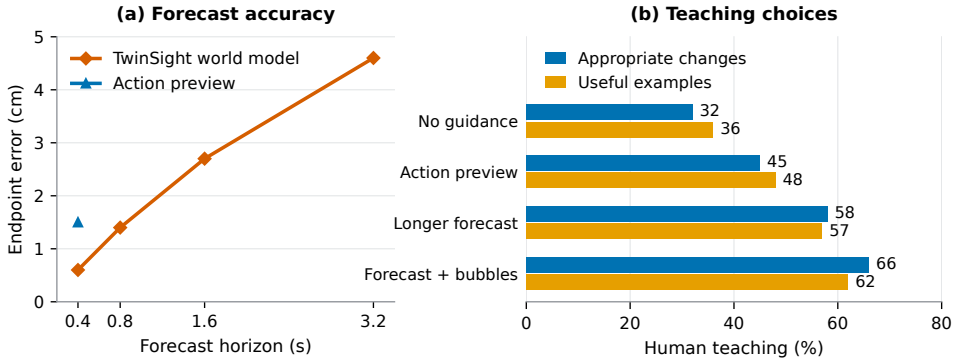



Figure 5: **Simulation forecasts and guidance ablation.** (a) Gripper endpoint error, with Action preview compared at 0.4 s. (b) Appropriate changes and useful examples (%, Appendix B.3) under four guidance conditions. Each uses five full demonstrations and 15 segment records, with all retained data used for adaptation and selection disabled.

0.4, 0.8, 1.6 and 3.2 s, respectively. At the shared 0.4 s horizon, the twin has lower endpoint error than Action preview, approximately 0.6 versus 1.5 cm (Figure 5a). This accuracy matters directly for teaching bubbles, since forecast error contributes to the discrepancy between predicted learner motion and observed human motion. Error increases with the forecast horizon.

The guidance ablation (Figure 5b) uses the same collection-record budget across conditions, with selection disabled and all retained data used for adaptation. The longer-forecast condition improves both appropriate changes and useful examples over Action preview. Adding teaching bubbles yields further gains of eight and five percentage points, respectively. The additional benefit of bubbles therefore concerns their use during teaching, independently of subsequent segment selection.

Table 2: **Selection on shared TWINSIGHT datasets.** For each participant and task, the four selectors use at most 60% of later human action steps, including context. All retained and All candidate use larger sets.

| Selection | Training steps | Updates | Selection + adaptation (min) | Success (%) |
|---|---|---|---|---|
| All retained | 7,255.8 | 3,558.1 | 24.3 | 73.7 |
| All candidate | 6,579.5 | 3,223.5 | 22.0 | 73.0 |
| Random segments | 4,992.6 | 2,432.3 | 16.8 | 69.0 |
| Policy-only | 5,036.8 | 2,453.6 | 18.1 | 71.0 |
| Performance-aware | 4,978.2 | 2,424.0 | 26.7 | 73.3 |
| TWINSIGHT selection | 5,014.4 | 2,442.4 | 28.7 | 74.0 |

**Segment selection and training efficiency.** The selection comparisons use the same datasets collected with TWINSIGHT guidance, separating the contribution of selection from that of collection. Policy-only selection ranks the same candidate segments by the frozen learner's mean imitation loss on human action targets, prioritising higher-loss segments within the shared training budget. At matched training budgets, TWINSIGHT outperforms Random segments and Policy-only selection, with a smaller advantage over Performance-aware selection (Table 2). This ordering persists across the tested budgets (Figure 6). In TWINSIGHT, candidates are prioritised by their mean recorded motion discrepancy, as defined in Section 3.3.

Relative to All retained, TWINSIGHT achieves similar mean success with 30.7% fewer unique training steps and 31.2% fewer optimisation updates. However, total selection plus adaptation time increases from 24.3 to 28.7 minutes. The measured efficiency gain therefore concerns the amount of data used for final adaptation and the resulting update count, while the cost of identifying those examples remains a computational trade-off.

**Real-robot evaluation.** To verify its applicability to physical manipulation, TWINSIGHT is evaluated on the FR3 using wrist and front camera views (Appendix A.2). Both cameras are calibrated to

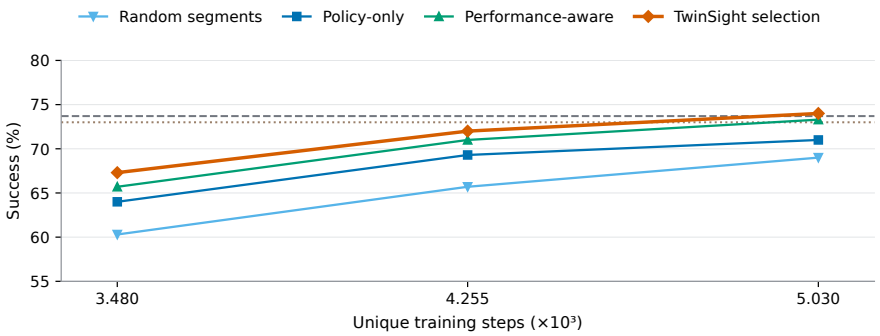



Figure 6: **Selection budgets on shared TWINSIGHT datasets.** All runs include five initial demonstrations. Budgets cover 20%, 40% or 60% of later human action steps. All retained (dashed) and All candidate (dotted) use larger sets (Table 2).

project forecast EE positions and teaching bubbles onto their corresponding image frames, providing visual guidance during demonstration (Figure 3). TWINSIGHT achieves the highest success rate on all three tasks (Table 3), with an advantage also observed at intermediate collection budgets (Figure 7). These results support the applicability of the framework to real-robot teaching and policy adaptation, with guidance displayed directly in the camera views.

Table 3: **FR3 success (%)** after 50 collection records per participant and task, with the same participants across conditions.

| Method | Colour-based placement | Collection and closure | Stacking | Mean |
|---|---|---|---|---|
| No guidance | 72 | 54 | 58 | 61.3 |
| HG-DAgger | 78 | 68 | 74 | 73.3 |
| RoboPocket | 80 | 70 | 72 | 74.0 |
| TWINSIGHT | **86** | **80** | **76** | **80.7** |

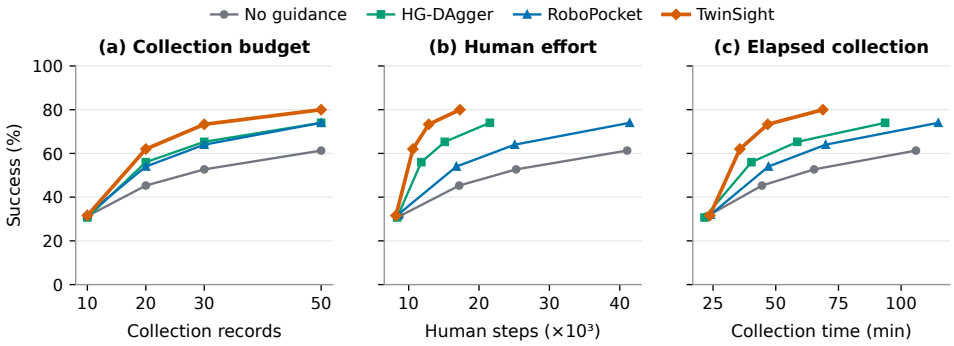



Figure 7: **FR3 success** versus collection records, human-controlled steps and collection time.

## 5 CONCLUSION

In this work, TWINSIGHT combines learner forecasts and persistent teaching bubbles to improve both teaching efficiency and policy performance. Across six simulated and three real-robot manipulation tasks, it achieves higher mean task success with fewer human-controlled steps and less collection time than the evaluated baselines. Ablation studies further support the complementary roles of longer forecasts and persistent teaching bubbles. The current study focuses on teaching with a fixed learner and predictive twin, followed by policy adaptation. Future work will explore repeated cycles of teaching and adaptation, updating the twin alongside the learner, and extend evaluation to a wider range of policies, tasks, and robot platforms.

### AI USE STATEMENT

Generative AI assisted with manuscript drafting and editing and with discussions of the method and experimental design. The authors take responsibility for the scientific content and final manuscript.

### ETHICS STATEMENT

The participant study received ethics approval. Identifying details are withheld for anonymous review and will be provided upon acceptance. FR3 experiments are supervised by a dedicated safety operator. The predictive feedback supports teaching decisions and does not replace the robot's safety procedures.

### REPRODUCIBILITY STATEMENT

Section 3 defines the predictive model, training objective, and feedback construction. Section 4 describes the experimental platforms, baselines, collection budgets, and evaluation protocol. Appendix A.2 provides the real-robot setup and camera projection details. We will release the code, configurations, and task definitions.

## REFERENCES

Christopher Agia, Rohan Sinha, Jingyun Yang, Rika Antonova, Marco Pavone, Haruki Nishimura, Masha Itkina, and Jeannette Bohg. CUPID: curating data your robot loves with influence functions. *CoRR*, abs/2506.19121, 2025. doi: 10.48550/ARXIV.2506.19121.

Annie S. Chen, Alec M. Lessing, Yuejiang Liu, and Chelsea Finn. Curating demonstrations using online experience. *CoRR*, abs/2503.03707, 2025a. doi: 10.48550/ARXIV.2503.03707.

Jingjing Chen, Hongjie Fang, Haoshu Fang, and Cewu Lu. Towards effective utilization of mixed-quality demonstrations in robotic manipulation via segment-level selection and optimization. In *IEEE International Conference on Robotics and Automation, ICRA 2025, Atlanta, GA, USA, May 19-23, 2025*, pp. 16884–16891. IEEE, 2025b. doi: 10.1109/ICRA55743.2025.11128787.

Sirui Chen, Chen Wang, Kaden Nguyen, Li Fei-Fei, and C. Karen Liu. Arcap: Collecting high-quality human demonstrations for robot learning with augmented reality feedback. In *IEEE International Conference on Robotics and Automation, ICRA 2025, Atlanta, GA, USA, May 19-23, 2025*, pp. 8291–8298. IEEE, 2025c. doi: 10.1109/ICRA55743.2025.11128717.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

Shivin Dass, Karl Pertsch, Hejia Zhang, Youngwoon Lee, Joseph J. Lim, and Stefanos Nikolaidis. PATO: policy assisted teleoperation for scalable robot data collection. In Kostas E. Bekris, Kris Hauser, Sylvia L. Herbert, and Jingjin Yu (eds.), *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, 2023. doi: 10.15607/RSS.2023.XIX.013.

Shivin Dass, Alaa Khaddaj, Logan Engstrom, Aleksander Madry, Andrew Ilyas, and Roberto Martín-Martín. Datamil: Selecting data for robot imitation learning with datamodels. In Carl Vondrick, Bharath Hariharan, Colin Raffel, Lerrel Pinto, Diyi Yang, and Aleksandra Faust (eds.), *The Fourteenth International Conference on Learning Representations, ICLR 2026, Rio de Janeiro, Brazil, April 23-27, 2026*. proceedings.iclr.cc, 2026.

Yilun Du, Sherry Yang, Pete Florence, Fei Xia, Ayzaan Wahid, Pierre Sermanet, Tianhe Yu, Pieter Abbeel, Joshua B Tenenbaum, Leslie Kaelbling, et al. Video language planning. In *International Conference on Learning Representations*, volume 2024, pp. 31138–31155, 2024.

Junjie Fang, Wendi Chen, Han Xue, Fangyuan Zhou, Tian Le, Yi Wang, Yuting Zhang, Jun Lv, Chuan Wen, and Cewu Lu. Robopocket: Improve robot policies instantly with your phone. *CoRR*, abs/2603.05504, 2026. doi: 10.48550/ARXIV.2603.05504.

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, and Jikun Cai et al. Libero-plus: A progressive robustness benchmark for visual-language-action models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2026, Denver, CO, USA, June 3-7, 2026*, pp. 38574–38583. Computer Vision Foundation, 2026.

Kanishk Gandhi, Siddharth Karamcheti, Madeline Liao, and Dorsa Sadigh. Eliciting compatible demonstrations for multi-human imitation learning. In Karen Liu, Dana Kulic, and Jeffrey Ichnowski (eds.), *Conference on Robot Learning, CoRL 2022, 14-18 December 2022, Auckland, New Zealand*, volume 205 of *Proceedings of Machine Learning Research*, pp. 1981–1991. PMLR, 2022.

Yanjiang Guo, Lucy Xiaoyang Shi, Jianyu Chen, and Chelsea Finn. Ctrl-world: A controllable generative world model for robot manipulation. In Carl Vondrick, Bharath Hariharan, Colin Raffel, Lerrel Pinto, Diyi Yang, and Aleksandra Faust (eds.), *The Fourteenth International Conference on Learning Representations, ICLR 2026, Rio de Janeiro, Brazil, April 23-27, 2026*. proceedings.iclr.cc, 2026.

Joey Hejna, Suvir Mirchandani, Ashwin Balakrishna, Annie Xie, Ayzaan Wahid, Jonathan Tompson, Pannag Sanketi, Dhruv Shah, Coline Devin, and Dorsa Sadigh. Robot data curation with mutual information estimators. *CoRR*, abs/2502.08623, 2025. doi: 10.48550/ARXIV.2502.08623.

Ryan Hoque, Ashwin Balakrishna, Ellen R. Novoseller, Albert Wilcox, Daniel S. Brown, and Ken Goldberg. Thriftydagger: Budget-aware novelty and risk gating for interactive imitation learning. In Aleksandra Faust, David Hsu, and Gerhard Neumann (eds.), *Conference on Robot Learning, 8-11 November 2021, London, UK*, volume 164 of *Proceedings of Machine Learning Research*, pp. 598–608. PMLR, 2021.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

Michael Kelly, Chelsea Sidrane, Katherine Rose Driggs-Campbell, and Mykel J. Kochenderfer. Hg-dagger: Interactive imitation learning with human experts. In *International Conference on Robotics and Automation, ICRA 2019, Montreal, QC, Canada, May 20-24, 2019*, pp. 8077–8083. IEEE, 2019. doi: 10.1109/ICRA.2019.8793698.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, and Sudeep Dasari et al. DROID: A large-scale in-the-wild robot manipulation dataset. In Dana Kulic, Gentiane Venture, Kostas E. Bekris, and Enrique Coronado (eds.), *Robotics: Science and Systems XX, Delft, The Netherlands, July 15-19, 2024*, 2024. doi: 10.15607/RSS.2024.XX.120.

KrishnaTeja Killamsetty, Durga Sivasubramanian, Ganesh Ramakrishnan, and Rishabh K. Iyer. GLISTER: generalization based data subset selection for efficient and robust learning. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pp. 8110–8118. AAAI Press, 2021. doi: 10.1609/AAAI.V35I9.16988.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, and Ashwin Balakrishna et al. Openvla: An open-source vision-language-action model. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard (eds.), *Conference on Robot Learning, 6-9 November 2024, Munich, Germany*, volume 270 of *Proceedings of Machine Learning Research*, pp. 2679–2713. PMLR, 2024.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. *CoRR*, abs/2502.19645, 2025. doi: 10.48550/ARXIV.2502.19645.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: benchmarking knowledge transfer for lifelong robot learning. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

Huihan Liu, Shivin Dass, Roberto Martín-Martín, and Yuke Zhu. Model-based runtime monitoring with interactive imitation learning. In *IEEE International Conference on Robotics and Automation, ICRA 2024, Yokohama, Japan, May 13-17, 2024*, pp. 4154–4161. IEEE, 2024a. doi: 10.1109/ICRA57147.2024.10611038.

Huihan Liu, Yu Zhang, Vaarij Betala, Evan Zhang, James Liu, Crystal Ding, and Yuke Zhu. Multi-task interactive robot fleet learning with visual world models. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard (eds.), *Conference on Robot Learning, 6-9 November 2024, Munich, Germany*, volume 270 of *Proceedings of Machine Learning Research*, pp. 4286–4313. PMLR, 2024b.

Marius Memmel, Jacob Berg, Bingqing Chen, Abhishek Gupta, and Jonathan Francis. STRAP: robot sub-trajectory retrieval for augmented policy learning. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, and Marc Szafraniec et al. Dinov2: Learning robust visual features without supervision. *Trans. Mach. Learn. Res.*, 2024, 2024.

Julian Quevedo, Ansh Kumar Sharma, Yixiang Sun, Varad Suryavanshi, Percy Liang, and Sherry Yang. Worldgym: World model as an environment for policy evaluation. In *International Conference on Learning Representations*, volume 2026, pp. 78932–78957, 2026.

Qwen. Qwen3-vl technical report. *CoRR*, abs/2511.21631, 2025. doi: 10.48550/ARXIV.2511.21631.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021.

Stéphane Ross, Geoffrey J. Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In Geoffrey J. Gordon, David B. Dunson, and Miroslav Dudík (eds.), *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2011, Fort Lauderdale, USA, April 11-13, 2011*, volume 15 of *JMLR Proceedings*, pp. 627–635. JMLR.org, 2011.

StarVLA. Starvla: A lego-like codebase for vision-language-action model developing. *CoRR*, abs/2604.05014, 2026. doi: 10.48550/ARXIV.2604.05014.

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, IROS 2012, Vilamoura, Algarve, Portugal, October 7-12, 2012*, pp. 5026–5033. IEEE, 2012. doi: 10.1109/IROS.2012.6386109.

Sherry Yang, Yilun Du, Seyed Kamyar Seyed Ghasemipour, Jonathan Tompson, Leslie Pack Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, and Sihyun Yu et al. World action models are zero-shot policies. *CoRR*, abs/2602.15922, 2026. doi: 10.48550/ARXIV.2602.15922.

# A EXPERIMENTAL SETUP DETAILS

## A.1 TASKS AND INITIAL CONFIGURATIONS

### A.1.1 TASK INSTRUCTIONS AND SIMULATION PLATFORM

The six simulation tasks use customised manipulation goals in MuJoCo, with objects and environments drawn from the LIBERO-90 scenes listed in Table 4. The table also gives the FR3 task instructions. Spatial terms such as left and right are defined from the robot's perspective. The simulation learner receives the instruction and two $256 \times 256$ RGB images from the front and wrist cameras. Demonstrations are collected through teleoperation at 20 Hz. Task-specific execution limits range from 900 to 1,300 control steps and are held constant across conditions.

Table 4: Task instructions and source LIBERO scenes for simulation, followed by FR3 instructions.

| Task | Instruction |
|---|---|
| *Simulation* | |
| Spatial composition<br>`LIVING_ROOM_SCENE5` | Put the white mug on the right plate and the red mug on the left plate. |
| Stacking<br>`KITCHEN_SCENE2` | Put the front black bowl on the plate, then stack the back black bowl on it. |
| Multi-object placement<br>`LIVING_ROOM_SCENE3` | Put tomato sauce and ketchup in the tray. |
| Drawer manipulation<br>`KITCHEN_SCENE4` | Put the wine bottle in the bottom drawer and close the drawer. |
| Loaded transport<br>`KITCHEN_SCENE9` | Put the white bowl on the frying pan and put the frying pan on the cabinet shelf. |
| Long-horizon composition<br>`KITCHEN_SCENE5` | Put the black bowl on the plate, put the ketchup in the top drawer, and close the drawer. |
| *FR3* | |
| Colour-based placement | Put the blue cup on the blue plate and the yellow cup on the yellow plate. |
| Collection and closure | Put all the fruit in the basket and close the basket. |
| Stacking | Stack the three wooden blocks. |

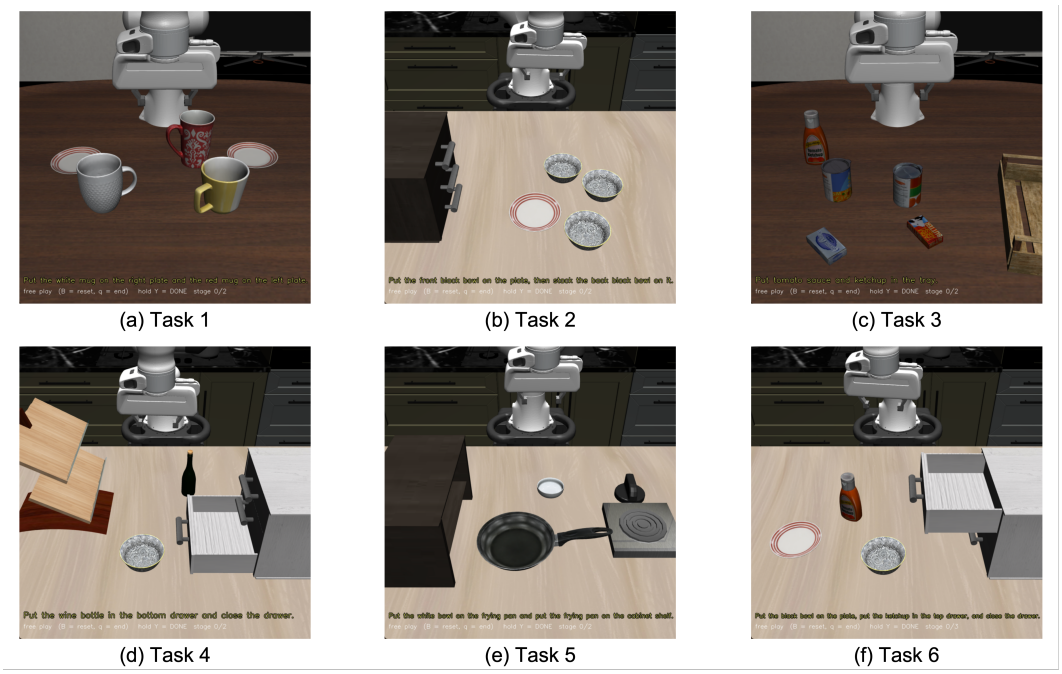
### A.1.2 SIMULATION INITIAL STATES

Simulation fixtures remain at their original scene locations. Movable objects and the initial gripper pose are perturbed around their nominal placements within the bounds in Table 5. Sampled states are rejected if: an object falls, tilts by more than $20^\circ$, contacts a fixture, any task stage is already satisfied, or if gripper initialisation displaces an object by more than 1 cm.

Table 5: Simulation perturbations from nominal object placements and the gripper home pose. Horizontal ranges apply independently to each axis.

| Quantity | Range or constraint |
|---|---|
| Object horizontal position | $\pm 7.5$ cm |
| Tray horizontal position | $\pm 5$ cm |
| Object rotation about the vertical axis | $\pm 45^\circ$ |
| Tray rotation about the vertical axis | $\pm 15^\circ$ |
| Minimum object separation | 10–16 cm, task dependent |
| Gripper horizontal position | $\pm 5$ cm |
| Gripper vertical position | $-2$ to $+5$ cm |
| Gripper yaw | $\pm 10^\circ$ |
| Initial gripper state | Open |

A fresh simulation state is sampled for every demonstration attempt, with seeds indexed by participant, task, and attempt. The 50 held-out evaluation states per task are shared across participants and conditions and excluded from collection and adaptation data.



(a) Task 1 (b) Task 2 (c) Task 3

(d) Task 4 (e) Task 5 (f) Task 6

Figure 8: Simulation objects and targets, with fixed and wrist-camera views inset.

### A.2 Real-robot setup

Real-robot experiments use a FR3 commanded through Polymetis, with a RealSense D435i wrist camera and a D455 front camera. Demonstrations are collected through Xbox-controller teleoperation, and a human assesses success against the full task goals in Table 4. Camera calibration maps forecast EE positions and teaching bubbles into the displayed views.

Object placements and the initial EE position are randomised, with the EE's horizontal position constrained to the $40 \times 55$ cm workspace. Initialisation follows the same procedure across conditions, with held-out configurations for evaluation. The predictive twin is adapted using FR3 data before guided collection. The learner and twin remain fixed during TWINSIGHT collection.

### A.3 Learner adaptation

#### A.3.1 Policy and optimisation

Both platforms use StarVLA QwenOFT, initialised from the LIBERO-4in1 checkpoint saved after 50,000 pretraining updates. LoRA adapters are applied to all linear layers of the frozen Qwen3-VL-4B-Instruct backbone, including the vision tower. The OFT action head is trained in full with an L1 regression loss and predicts $H = 8$ actions, each containing six pose increments and one gripper command. Table 6 lists the principal settings.

#### A.3.2 Eligible examples and computation

Each adaptation starts independently from the pretrained checkpoint. Training examples use stride-one starts and contain $H = 8$ actions entirely within one retained demonstration or selected segment. Incomplete chunks are excluded, so an interval containing $L_j$ actions contributes $\max(L_j - H + 1, 0)$ eligible examples. Unique supervised action steps, eligible examples, and optimisation updates are counted separately. Learner adaptation and inference each use one NVIDIA GeForce RTX 5090 with 32 GB of memory.

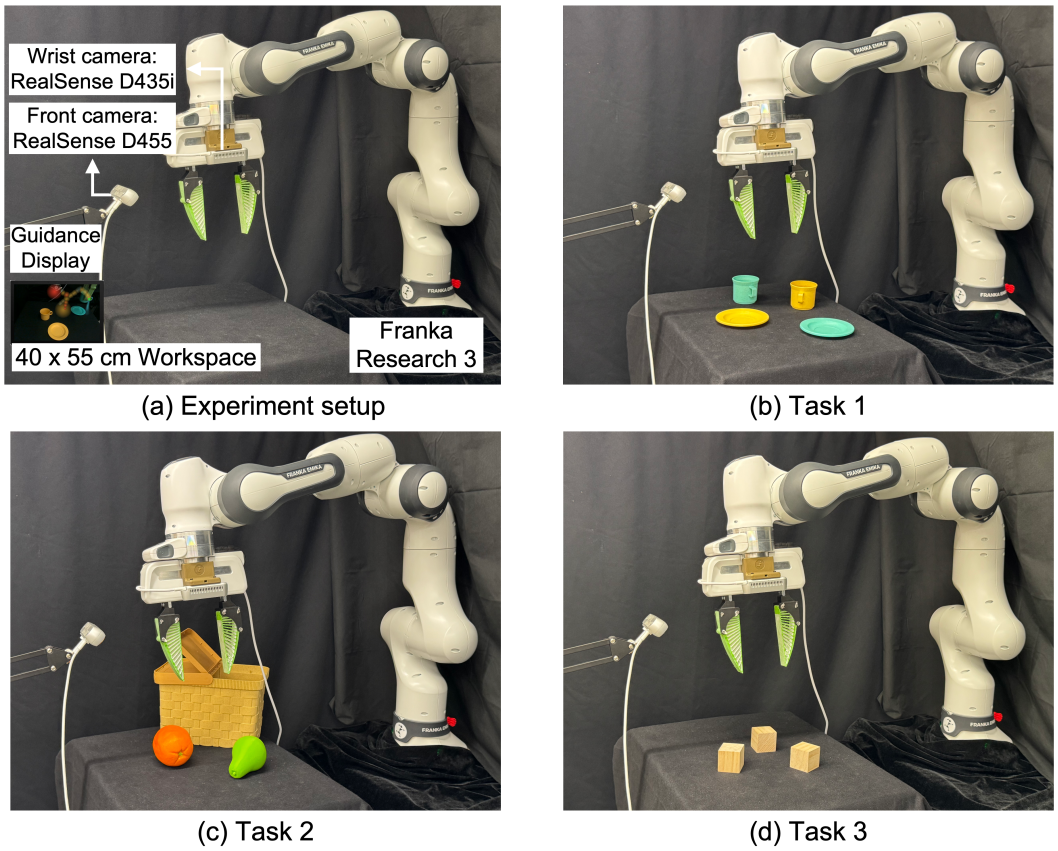



Figure 9: FR3 workspace, cameras and three manipulation tasks.

Table 6: Learner adaptation settings shared across platforms.

| Setting | Value |
|---|---|
| LoRA rank / $\alpha$ / dropout | 32 / 16 / 0 |
| Trainable components | Backbone adapters and full action head |
| Action head | MLP-ResNet with two residual blocks |
| Action chunk | Eight seven-dimensional actions |
| Batch size | 8 |
| Training budget | Up to four passes; at most 6,250 updates |
| Adapter / action-head learning rate | $4 \times 10^{-4}$ / $2 \times 10^{-4}$ |
| Optimiser | AdamW, $\beta_1 = 0.9$, $\beta_2 = 0.95$ |
| Weight decay / gradient clipping | $10^{-8}$ / 1.0 |
| Learning-rate decay | Cosine decay to $10^{-6}$ |
| Precision | bfloat16 mixed precision |
| Evaluation checkpoint | Final checkpoint |

## A.4 PREDICTIVE-TWIN TRAINING

### A.4.1 ARCHITECTURE AND OPTIMISATION

The simulation twin uses a frozen DINOv2 ViT-L/14 encoder with registers (Oquab et al., 2024; Darcet et al., 2024) to extract main-camera features. Its visual context comprises the current frame and the two preceding chunk-boundary frames, spaced eight control steps apart. Current-frame patch features are retained, while each history frame is pooled into one feature vector. A frozen CLIP ViT-B/32 text encoder (Radford et al., 2021) embeds the task instruction. A Transformer combines this context with the learner-proposed action chunk and predicts all horizons in one forward pass. The twin receives only main-camera images, while the learner receives both camera views. Table 7 lists the principal settings.

Table 7: Simulation predictive-twin architecture and training settings. Both feature encoders remain frozen.

| Setting | Value |
|---|---|
| Visual encoder | DINOv2 ViT-L/14 with registers, $224 \times 224$ input |
| Instruction encoder | CLIP ViT-B/32, 512-dimensional embedding |
| Predictor | 8 Transformer layers, width 512, 8 attention heads |
| Feed-forward width / dropout | 2,048 / 0 |
| Forecast horizons | $H$, $2H$, $4H$, $8H$, with $H = 8$ |
| Batch size / optimisation updates | 96 / 6,000 |
| Optimiser / learning rate | AdamW / $3 \times 10^{-4}$ |
| Weight decay / gradient clipping | 0.01 / 1.0 |
| Learning-rate schedule | Cosine decay to zero, without warm-up |
| Loss weights | $\lambda_p = 10$, $\lambda_r = 1$, $\lambda_a = 1$ |

### A.4.2 TRAINING DATA AND SUPERVISION

The simulation twin is fitted using learner executions, short trials under alternative actions, and existing human demonstrations. Table 8 summarises the data sources and their supervision. The learner preparation episodes have task-specific step caps that define their failure labels. These caps are separate from the final task-evaluation limits. Targets extending beyond a recorded episode are excluded from the forecast loss.

Table 8: Predictive-twin simulation data. Entry counts precede the training/held-out split.

| Source | Entries | Supervision |
|---|---|---|
| Learner executions | 29,142 | Visual and EE change at all horizons; failure and action reconstruction |
| Alternative-action trials | 8,364 | One-chunk visual change |
| Human demonstrations | 10,760 | One-chunk visual change and EE displacement |
| Total | 48,266 | 40,967 training; 7,299 held out |

Learner data comprise 750 episodes from 15 LIBERO-90 tasks. Alternative-action trials use 697 starting states. Human data come from the IPEC-COMMUNITY LIBERO-90 release and include 8,981 distinct chunks from 579 distinct demonstrations. The larger entry count includes repeated chunks. The held-out partition contains the final ten learner episodes per task and a separate subset of alternative-action states. Human demonstrations are used only for training.

## A.5 EVALUATION AND EFFORT ACCOUNTING

### A.5.1 TASK SUCCESS

Each participant's adapted learner is evaluated over 50 runs per task. Success requires completing the full task goal within its execution limit. Simulation reuses the held-out states described in Appendix A.1.2, and the FR3 uses held-out configurations generated under a common procedure

across conditions. Let $y_{ikj} \in \{0, 1\}$ denote success for participant $i$, task $k$, and evaluation run $j$. For a condition with $n$ participants and $K$ tasks, the reported means are

$$\bar{S}_k = \frac{100}{50n} \sum_{i=1}^{n} \sum_{j=1}^{50} y_{ikj}, \qquad \bar{S} = \frac{1}{K} \sum_{k=1}^{K} \bar{S}_k. \tag{6}$$

Participants receive equal weight within each task, and tasks receive equal weight overall. Means are computed before rounding the displayed rates.

#### A.5.2 Human effort and training cost

A warm-up session of about 10 minutes was arranged for each participant to help them become familiar with the procedures. Then human effort includes human-controlled steps from both retained and discarded attempts, including takeover steps in intervention conditions. Collection time includes resets, planning, review, learner execution, and waiting. Training-data size counts unique supervised human action steps, including selected context, while eligible training examples and optimisation updates are recorded separately. Selection and adaptation time includes candidate preparation, score computation, and final learner adaptation. The reported selection and adaptation times exclude predictive-twin data collection and fitting before teaching.

## B Comparison definitions and validation protocol

### B.1 Segment-selection comparisons

#### B.1.1 Shared candidates and training budget

All selectors use each participant–task dataset collected with TWINSIGHT guidance and the same candidate pool from Section 3.3, restricted to $q(C) > 0$. Every adaptation set includes the five initial complete demonstrations. Budget caps are 20%, 40%, or 60% of unique action steps in later retained records, including context. Initial demonstrations contribute to total training data and human effort but lie outside these budgets.

Table 9: Training sets and ranking rules for the selection comparisons.

| Condition | Training set or ranking rule |
|---|---|
| All retained | All eligible chunks from retained human-action records, without a later-action budget cap. |
| All candidate | Union of eligible chunks from every candidate in the common pool, with duplicate chunks counted once and no later-action budget cap. |
| Random segments | A random ordering of the common candidate pool. |
| Policy-only | Descending mean L1 imitation loss of the frozen learner, averaged over complete chunks within each candidate. |
| Performance-aware | Descending mean predicted episode-failure score over the original visit's comparison intervals. |
| TWINSIGHT selection | Descending mean recorded motion discrepancy, $q(C)$ in Equation 5. |

The four budgeted rankings accept whole candidates, skipping overlaps with accepted segments from the same source record and candidates exceeding the remaining budget. Ties are resolved by source-record order and original-visit start time. Random selection uses a fixed seed. For All candidate, complete $H = 8$ chunks are constructed within each candidate, then deduplicated by source record and chunk start. All comparisons use the same initial checkpoint, paired adaptation seeds, and the training recipe in Appendix A.3.

#### B.1.2 Performance-aware selection

Performance-aware selection ranks candidates by the twin's mean episode-failure score. Using the interval-start set $I_C$ from Section 3.3,

$$q_{\mathrm{PA}}(C) = \frac{1}{|I_C|} \sum_{t \in I_C} r_t. \tag{7}$$

Each interval contributes once, and context is excluded from the score. The learner and twin remain frozen, and final evaluation outcomes are excluded from scoring. This baseline contrasts predicted episode failure with the local motion discrepancy used by TWINSIGHT.

### B.2 FORECAST VALIDATION

#### B.2.1 QUERIES AND LEARNER CONTINUATIONS

The learner and twin remain frozen during validation. Simulation records collected with TWINSIGHT guidance and excluded from twin fitting and tuning provide up to ten evenly spaced chunk-boundary queries per record with the required visual history. Both predictors use the same queries. The Action preview endpoint is the final commanded EE position obtained from the same action chunk using the controller's action scaling and pose-update convention. Each saved simulator state is restored, the exact learner-proposed $H$-step chunk supplied to the twin is executed, and the learner then observes and replans every $H$ steps. Its EE endpoints are recorded at $H$, $2H$, $4H$, and $8H$.

#### B.2.2 ENDPOINT ERROR AND AGGREGATION

Using the main paper's forecast notation, endpoint error is

$$e_t(h) = \left\| \hat{p}_{t+h|t} - p^{\pi}_{t+h} \right\|_2 , \tag{8}$$

where $p^{\pi}_{t+h}$ is the observed endpoint under learner control. The human endpoint is used for teaching-bubble discrepancy. Action preview is compared only at the shared $H$-step horizon of 0.4 s. Endpoints beyond execution termination are excluded.

Valid errors are averaged within records, records within participant–task pairs, tasks within participants, and then participants equally.

### B.3 TEACHING ASSESSMENTS

#### B.3.1 EVIDENCE AND ASSESSMENT UNITS

The guidance assessment compares the four conditions using five complete demonstrations plus 15 retained segment records, with selection disabled and all retained data used for adaptation. A short task-stage rubric is fixed before rating, using separate frozen-learner development rollouts excluded from final evaluation. The rubric associates each observed learner difficulty with the relevant corrective behaviour and an observable local completion criterion.

Each later human attempt is compared with the most recent earlier attempt covering the same task stage, within participant, task, and guidance condition. A revision is an observable change in behaviour beyond differences caused by initial object placements. Each revised attempt receives one judgment, positive if at least one change addresses an evidenced difficulty. Each retained later segment receives one usefulness judgment (Table 10).

Table 10: Teaching measures, criteria and assessment units.

| Measure | Positive judgment | Assessed unit |
|---|---|---|
| Appropriate change | The observable revision addresses a difficulty supported by the development rollouts. Complete task success is not required. | One revised attempt, including failed or discarded attempts. |
| Useful example | The retained segment demonstrates a response to an evidenced difficulty, meets its local completion criterion, and preserves coherent surrounding actions. | One of the 15 retained later segments; initial demonstrations are excluded. |