XTRAJECTORY: LEARNING WORLD DYNAMICS IN VISION-LANGUAGE MODELS THROUGH UNIFIED FUTURE TRAJECTORY PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Future prediction is essential for agents that reason and act in the physical world, yet existing visual world models commonly represent future dynamics through pixels, latent states, or specialized motion modules. We investigate whether predictive physical dynamics can instead be acquired directly by a general-purpose vision-language model (VLM) through structured future trajectory prediction. We introduce **XTrajectory**, a unified framework that represents diverse future motion as autoregressively generated coordinate sequences through the VLM's native language interface, without introducing an additional trajectory decoder or dynamics head. XTrajectory unifies the future motion of visual points, physical objects, human hands, and robot end-effectors under a common history-to-future formulation, and learns this capability through geometrically verifiable reinforcement post-training. Experiments on existing trajectory benchmarks and our XTrajectory-Bench show improvements in future trajectory forecasting and strong transfer to unseen manipulation and physical-interaction scenarios. Evaluation on general multimodal benchmarks further reveals meaningful, though non-uniform, transfer beyond trajectory prediction. These results suggest that future trajectories provide a compact, interpretable, and directly verifiable interface for introducing predictive world dynamics into general-purpose VLMs.

1 INTRODUCTION

An agent acting in the physical world must not only recognize the current scene, but also anticipate how it will evolve over time and under interaction. World models seek to acquire such predictive dynamics by modeling future environment states, typically through pixels, video tokens, or learned latent representations Ha & Schmidhuber (2018); Hafner et al. (2018; 2024); Bruce et al. (2024); Zhou et al. (2024); Assran et al. (2025). These predictive representations have enabled increasingly capable simulation, planning, and control, but they often rely on dedicated dynamics modules, tokenizers, or generative decoders. In particular, predicting complete future observations requires modeling visual details beyond task-relevant motion, while latent dynamics are not directly expressed in an explicit geometric form. Meanwhile, multimodal foundation models are increasingly used as generalist backbones for embodied reasoning and action Brohan et al. (2023); Kim et al. (2024); Team et al. (2025); Team (2026). This raises a complementary question: can predictive world dynamics be learned directly through the native autoregressive interface of a general-purpose VLM?

We explore this question through *point-centric world modeling*, where future dynamics are represented by the trajectories of grounded points rather than complete future observations. A point trajectory explicitly describes how an entity evolves in space over time, providing a compact representation with direct geometric meaning and deterministic evaluation. Importantly, the same 2D coordinate representation can describe the future motion of human hands, manipulated objects, robot end-effectors, physically interacting objects, and arbitrary visual points, and can be serialized directly through the language output space of a VLM. Such trajectories do not replace pixels or latent states; rather, they provide an interpretable and measurable projection of future dynamics that is closely related to the motion and interaction patterns relevant to physical reasoning and embodied decision making.

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.

Recent work has begun to model future dynamics in point or trace space, demonstrating that structured motion representations can provide a compact alternative to pixel-level prediction Boduljak et al. (2026); Lee et al. (2026); Huang et al. (2026a); Zhang et al. (2026a). However, existing approaches are typically designed around specific geometric spaces, target entities, conditioning schemes, or downstream domains. It therefore remains unclear whether heterogeneous future dynamics can be internalized directly through a general-purpose VLM's native output interface and acquired without reducing the model to a narrow trajectory specialist.

To address this, we introduce **XTrajectory**, a unified trajectory post-training framework for general-purpose VLMs. XTrajectory formulates heterogeneous future motion, including human hands, manipulated objects, robot end-effectors, arbitrary visual points, and physically interacting objects, as future 2D coordinate generation from visual history and task context. The model directly generates structured trajectories through its native autoregressive language interface, without introducing a trajectory-specific decoder or dynamics head. We convert diverse physical, robotic, and egocentric trajectory data into a common history-to-future interface, enabling direct geometric verification across tasks. Building on this representation, we develop a coordinate-based reinforcement-learning objective that integrates complementary aspects of trajectory quality, together with **Trajectory Segment GSPO (TS-GSPO)**, which provides fine-grained credit assignment to individual future prediction steps.

To evaluate whether the learned capability transfers beyond the post-training domains, we introduce **XTrajectory-Bench**, a cross-domain benchmark for out-of-distribution trajectory forecasting. It unifies Physics-IQ Motamed et al. (2026), Apple-π Yao et al. (2026), and HOT3D Banerjee et al. (2025) under a unified history-to-future 2D trajectory interface, covering physical dynamics and egocentric manipulation. XTrajectory model shows strong transfer on this benchmark and improves over the base VLM on established trajectory benchmarks. Evaluation on general multimodal and spatial-reasoning benchmarks further reveals positive transfer to several point-centric capabilities together with a non-uniform specialization–retention trade-off.

Our main contributions are:

- We introduce **XTrajectory**, a unified framework for post-training general-purpose VLMs on heterogeneous future trajectory prediction through a shared 2D coordinate interface.
- We develop a **geometrically verifiable reinforcement-learning framework** together with **TS-GSPO** for fine-grained trajectory credit assignment.
- We introduce **XTrajectory-Bench** for cross-domain trajectory forecasting and systematically evaluate both forecasting generalization and broader multimodal capability retention.

2 Related Work

Future Trajectory Prediction. Recent work increasingly models future dynamics with trajectories rather than high-dimensional pixels. ATM predicts future 2D motion of arbitrary query points for robot policy learning Wen et al. (2023), while Track2Act uses predicted point tracks as intermediate representations for manipulation Bharadhwaj et al. (2024). More recent methods extend trajectory forecasting to generative and open-world settings: What Happens Next models dense future point trajectories Boduljak et al. (2026), TrajLoom jointly predicts dense trajectories and visibility Zhang et al. (2026b), and Forecasting Motion in the Wild treats point trajectories as category-agnostic motion tokens Thakkar et al. (2026). In 3D, PointWorld predicts action-conditioned point flows Huang et al. (2026a), while MolmoMotion forecasts language-conditioned 3D trajectories of object-attached points Zhang et al. (2026a). Trajectory prediction has also been explored with vision-language models. Merlin incorporates object-level trajectories into multimodal future reasoning Yu et al. (2023), and TrajVLM-Gen predicts coarse trajectories for controllable video generation Yang et al. (2025). Nevertheless, existing approaches typically rely on task-specific architectures, trajectory representations, or conditioning schemes, resulting in fragmented solutions across 2D/3D, sparse/dense, and object-/point-level forecasting. Therefore, we hope to formulate heterogeneous trajectory prediction through the native autoregressive interface of a single general-purpose VLM and explicitly examine whether trajectory specialization can be acquired without substantially sacrificing its broader multimodal capabilities.

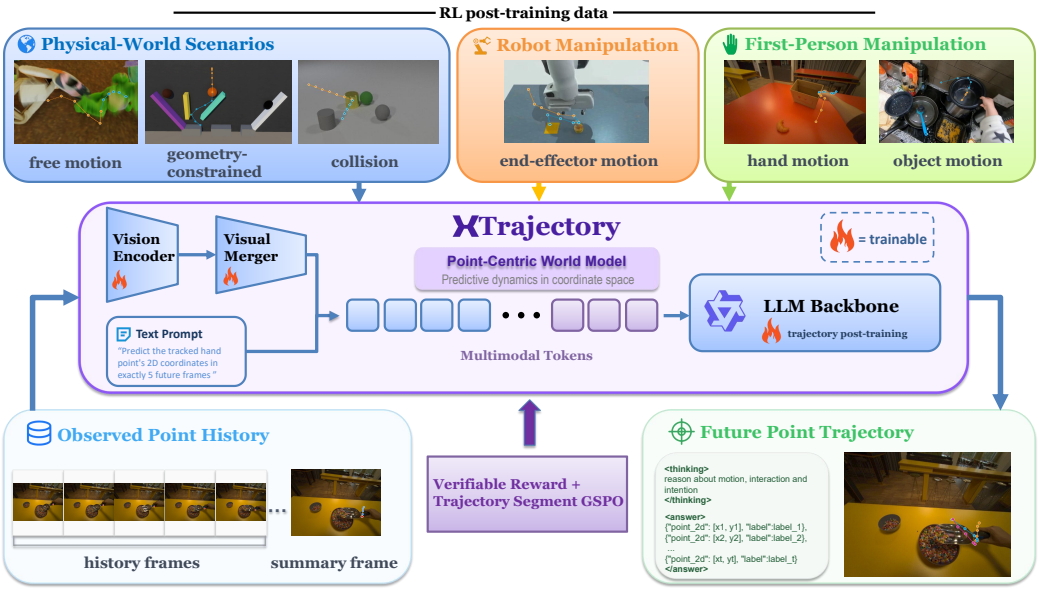

Figure 1: **Overview of XTrajectory.** XTrajectory post-trains a generalist VLM on heterogeneous trajectory data spanning physical-world dynamics, robot manipulation, and first-person human–object interactions. At inference, the model takes the observed point history, reasons about the underlying motion dynamics, and autoregressively predicts the future point trajectory directly in coordinate space as well as labels explaining current step through its native language interface.

Reinforcement Learning for VLMs. Reinforcement learning with verifiable rewards has recently become an effective post-training paradigm for vision-language models. Visual-RFT introduces task-specific visual rewards for reinforcement fine-tuning Liu et al. (2025), while VLM-R1 demonstrates the effectiveness of rule-based RL across general visual tasks Shen et al. (2025). Vision-R1 applies GRPO to multimodal reasoning Huang et al. (2026b), and R1-VL further introduces step-wise rewards to improve intermediate credit assignment Zhang et al. (2025). Subsequent studies refine reward attribution through consistency-, reflection-, or role-aware objectives Chen et al. (2026); Wan et al. (2026); Jiang & Zou (2026). Despite these advances, specialized RL post-training can still degrade pretrained capabilities. RECAP and Continual Policy Optimization both report substantial forgetting under multimodal RL and introduce mechanisms to preserve prior knowledge Phan et al. (2026); Luo et al. (2026). Our work studies this trade-off for future trajectory prediction, where geometric trajectories provide structured and directly verifiable supervision. Rather than optimizing an isolated perception or reasoning benchmark, we use RL to acquire a unified spatiotemporal forecasting capability and jointly evaluate its effect on general-purpose multimodal competence.

3 XTrajectory: A Unified Framework for Future Trajectory Prediction

3.1 Task Formulation

We study *future trajectory prediction* as a unified multimodal task for a general-purpose vision-language model (VLM). Given an observed visual context $\mathcal{V}$, the corresponding history trajectory $\mathcal{H}$, and textual context $\mathcal{C}$, the model forecasts the target trajectory over T future steps:

$$\hat{\mathcal{T}} = f_\theta(\mathcal{V}, \mathcal{H}, \mathcal{C}), \qquad \hat{\mathcal{T}} = \{\hat{\mathbf{p}}_t\}_{t=1}^{T}, \tag{1}$$

where $\hat{\mathbf{p}}_t = [\hat{x}_t, \hat{y}_t]$ denotes the predicted 2D coordinate and each predicted point is additionally accompanied by a semantic label $\hat{l}_t$ describing the corresponding motion state. These labels are auxiliary free-form descriptions introduced for interpretability and are not treated as separately supervised or evaluated prediction targets.

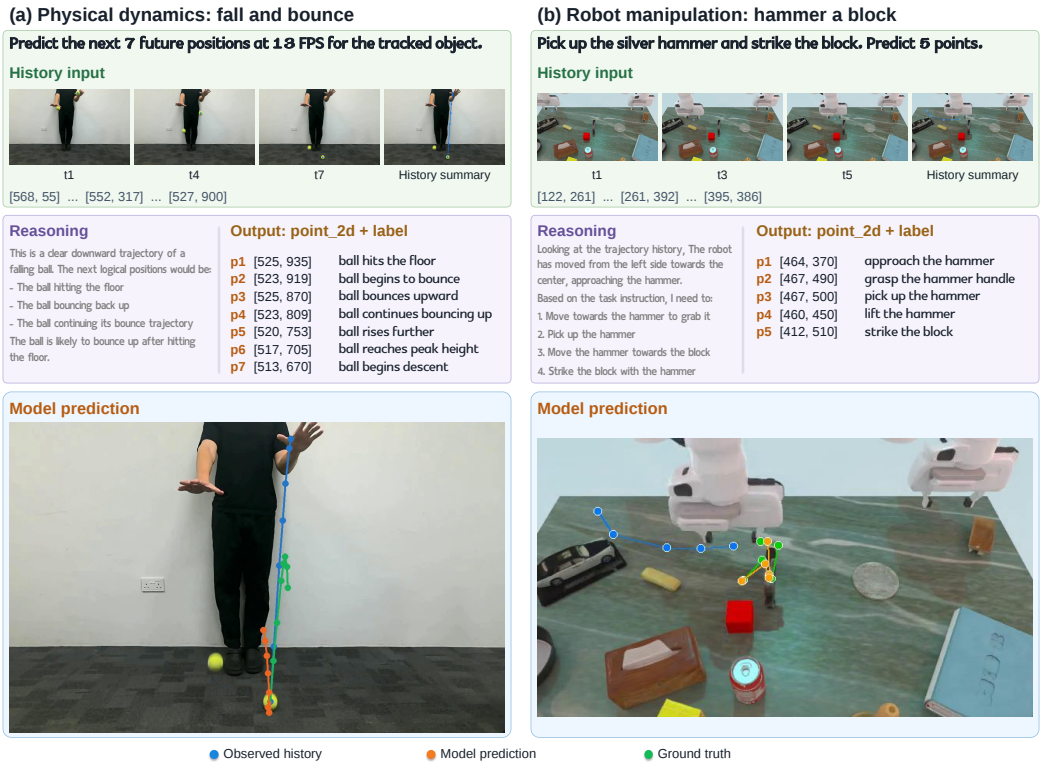

Figure 2: Qualitative examples of XTrajectory on physical dynamics and robot manipulation. The model reasons over the observed history and predicts future 2D trajectories together with semantic motion labels. In (a), it anticipates the fall-and-bounce dynamics of a ball; in (b), it predicts the manipulation sequence for grasping and using a hammer. Blue, orange, and green denote observed history, model prediction, and ground truth, respectively.

The prediction target is grounded by its observed trajectory history. For H observed frames, we denote the visual history and observed trajectory as

$$\mathcal{V} = \{I_1, \ldots, I_H\}, \qquad \mathcal{H} = \{\mathbf{p}_h^{\text{hist}}\}_{h=1}^{H}. \tag{2}$$

The history coordinates can be provided through the textual context, or explicitly marked on each history frame.

3.2 Unified Trajectory Subtasks

Our formulation supports heterogeneous future-motion prediction under the same VLM interface. We organize the training tasks into two broad categories that capture complementary sources of visual dynamics.

Dynamics-driven forecasting predicts how a visual target evolves from its observed motion and physical interactions. The target may correspond to an arbitrary visual point or a physical object, covering motion continuation, collision-induced changes, and geometry-constrained dynamics. These tasks are conditioned on a task prompt $\mathcal{P}$ that specifies the prediction target, forecast horizon, temporal setting, and required output format, while future motion must be inferred from the observed visual dynamics.

Manipulation-driven forecasting predicts future motion arising from human or robotic interaction. The target can be a human hand, manipulated object, or robot end-effector. In addition to the common task prompt $\mathcal{P}$, some samples provide an explicit task instruction $\mathcal{I}$ describing the intended manipulation goal, whereas instruction-free samples require the model to infer intent directly from the observed interaction.

Accordingly, the textual context in Eq. (1) is instantiated as

$$\mathcal{C} = \begin{cases} \mathcal{P} \oplus \mathcal{I}, & \text{instruction-conditioned manipulation,} \\ \mathcal{P}, & \text{dynamics-driven or instruction-free forecasting,} \end{cases} \tag{3}$$

where $\mathcal{P}$ denotes the common task-specified trajectory-prediction prompt, $\mathcal{I}$ denotes an optional task-level instruction, and $\oplus$ denotes their textual composition.

Despite different target semantics, motion regimes, and conditioning signals, all subtasks share the same history-to-future 2D trajectory interface and are handled by a single VLM without task-specific prediction heads.

We instantiate these tasks using diverse data sources. For dynamics-driven forecasting, we use Kubric Greff et al. (2022), CLEVRER Yi et al. (2020), selected PhysInOne Zhou et al. (2026) samples, and the Ball Playground subset of ContPhy Zheng et al. (2024). For manipulation-driven forecasting, we use selected EgoVerse Punamiya et al. (2026) scenes, collected RoboTwin Chen et al. (2025) data, and ObjectForesight/EPIC Soraki et al. (2026); Damen et al. (2020). Together, these datasets yield **110,828** training samples spanning physical dynamics, human interaction, and robotic manipulation. Detailed construction and filtering procedures are provided in Appendix A, and the detailed number of training samples are given in Table 5

3.3 Trajectory Reinforcement Post-Training

Trajectory prediction is naturally suited to reinforcement post-training because future coordinates can be directly verified against ground truth without a learned reward model. Given a prompt x, the current policy π_θ samples a group of G complete responses on-policy. We first evaluate each rollout with a unified response-level reward and optimize the complete response using a sequence-level GSPO objective Zheng et al. (2025). We then exploit the point-wise verifiability of trajectory coordinates through an additional segment objective, which provides finer credit assignment to individual future predictions using the same rollouts.

3.3.1 Verifiable Trajectory Reward

For rollout i, we define the response-level reward as

$$R_i = 0.5R_{\text{fmt}} + 0.5R_{\text{think}} + R_{\text{acc}}, \tag{4}$$

where R_{fmt} rewards a valid structured response with the expected number of future points, R_{think} encourages a sufficiently long reasoning trace, and R_{acc} measures geometric trajectory quality.

Let $\hat{\mathcal{T}} = \{\hat{\mathbf{p}}_t\}_{t=1}^{T}$ and $\mathcal{T} = \{\mathbf{p}_t\}_{t=1}^{T}$ denote the predicted and ground-truth trajectories in normalized $[0, 1000]^2$ coordinates. We evaluate coordinate accuracy, global trajectory shape, and overall motion direction:

$$\begin{aligned} R_{\text{coord}} &= \exp\left(-\frac{d_{\text{RMS}}}{120}\right), \qquad R_{\text{frechet}} = \exp\left(-\frac{d_{\text{F}}}{180}\right), \\ R_{\text{dir}} &= \text{clip}\left(\frac{\cos(\Delta\hat{\mathbf{p}}, \Delta\mathbf{p}) + 1}{2}, 0, 1\right), \end{aligned} \tag{5}$$

where $d_{\text{RMS}} = \sqrt{\frac{1}{T}\sum_t \|\hat{\mathbf{p}}_t - \mathbf{p}_t\|_2^2}$, d_{F} is the discrete Fréchet distance, and $\Delta\mathbf{p} = \mathbf{p}_T - \mathbf{p}_1$. Degenerate cases in which the direction is undefined are handled separately as described in Appendix E.

The final trajectory accuracy reward is

$$R_{\text{acc}} = 0.55R_{\text{coord}} + 0.25R_{\text{frechet}} + 0.20R_{\text{dir}}. \tag{6}$$

For each prompt, the response-level rewards are normalized across the G rollouts to obtain

$$A_i^{\text{global}} = \frac{R_i - \mu_R}{\sigma_R + \epsilon}, \tag{7}$$

which is used by the global GSPO objective to optimize the complete response.

3.3.2 TRAJECTORY SEGMENT GSPO

A single response-level advantage provides only coarse credit assignment: accurate and inaccurate future coordinates within the same rollout receive the same global optimization signal. We therefore introduce **Trajectory Segment GSPO (TS-GSPO)**, which augments the global GSPO objective with fine-grained credit for individual future prediction steps.

For rollout i and future step t, we first measure the accuracy of the predicted coordinate using a local geometric reward

$$r_{i,t} = \exp\left(-\frac{\|\hat{\mathbf{p}}_{i,t} - \mathbf{p}_t\|_2}{100}\right). \tag{8}$$

To account for the effect of the current prediction on later trajectory quality, we aggregate the rewards from step t onward using a discounted return:

$$G_{i,t} = \sum_{k=t}^{T} \gamma^{k-t} r_{i,k}, \qquad \gamma = 0.9. \tag{9}$$

We then normalize $G_{i,t}$ across the G rollouts at the same prediction horizon:

$$A_{i,t}^{\text{seg}} = \frac{G_{i,t} - \mu_t}{\sigma_t + \epsilon}, \tag{10}$$

where μ_t and σ_t are computed across rollouts for the same future step. This same-step comparison avoids mixing prediction steps with different levels of difficulty.

For the token span $S_{i,t}$ corresponding to the serialized t-th coordinate, we combine the segment-level signal with the response-level advantage:

$$\tilde{A}_{i,t} = 0.5 A_i^{\text{global}} + 0.5 A_{i,t}^{\text{seg}}. \tag{11}$$

Importantly, this combined advantage is used in a *separate* segment-level clipped objective rather than replacing the original global GSPO objective. The overall optimization is

$$\mathcal{L} = \mathcal{L}_{\text{GSPO}} + \mathcal{L}_{\text{seg}} + \beta \mathcal{L}_{\text{KL}}. \tag{12}$$

Thus, coordinate spans receive both global response-level supervision and step-specific trajectory supervision, while the remaining response tokens are optimized only by the global objective. TS-GSPO reuses the same on-policy rollouts and requires no additional sampling. Details of span mapping and segment-level optimization are provided in Appendix E.

4 XTRAJECTORY-BENCH

Existing trajectory benchmarks are largely tied to specific embodied tasks, target types, or conditioning formats, making it difficult to determine whether a model has learned transferable future dynamics or merely task-specific motion priors. To address this limitation, we introduce **XTrajectory-Bench**, a cross-domain benchmark for out-of-distribution trajectory forecasting under a history-to-future 2D trajectory interface.

XTrajectory-Bench contains **822** evaluation samples constructed from Physics-IQ Motamed et al. (2026), Apple-π Yao et al. (2026), and HOT3D Banerjee et al. (2025), covering real-world physical interactions, diverse mechanics, and egocentric human–object manipulation. All subsets are converted into the same history-to-future trajectory format, enabling consistent evaluation across domains. We evaluate predictions using **RMSE**, which measures point-wise coordinate error over the forecast horizon, and **Fréchet distance**, which captures the overall geometric similarity between the predicted and ground-truth trajectories. Lower values indicate better performance for both metrics. Detailed construction process for our XTrajectory-Bench is provided in Appendix B.

Physics-IQ Motamed et al. (2026) provides real-world videos depicting diverse physical phenomena, from which we retain **80** processed samples for XTrajectory-Bench to evaluate natural physical interactions. **Apple-π** Yao et al. (2026) contains real and simulated videos spanning a broad range of classical-mechanics scenarios, from which we retain **339** samples to evaluate more diverse physical dynamics. Finally, we construct **xHOT3D** from HOT3D Banerjee et al. (2025), yielding **403** samples for evaluating egocentric human–object interactions and manipulation dynamics.

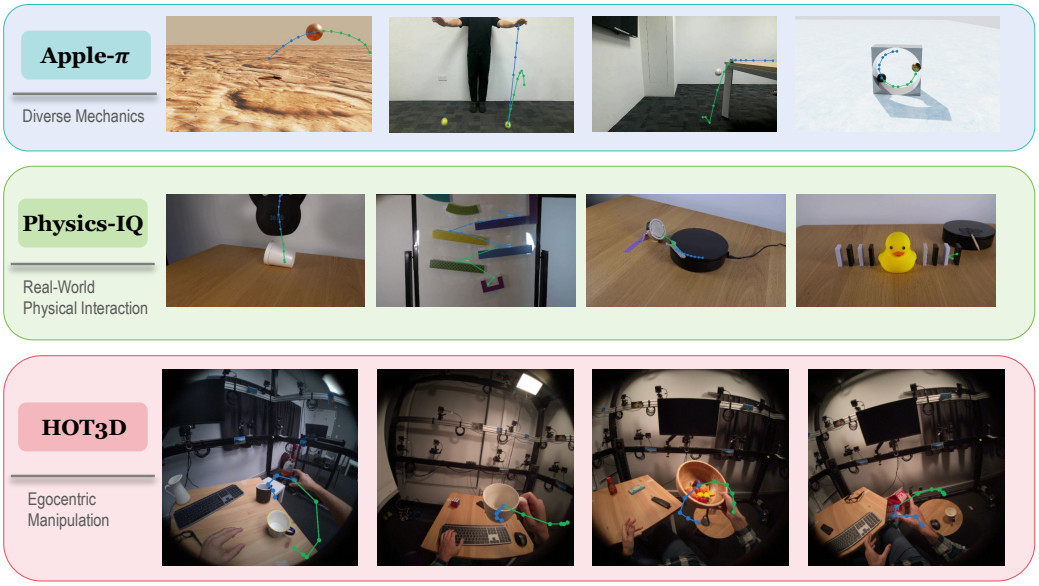

Figure 3: Representative examples from **XTrajectory-Bench**, from top to bottom is Apple-π, Physics-IQ and xHOT3D. Blue and green denote the observed history and held-out future trajectory, respectively.

5 Experiments

5.1 Experiment Settings

We jointly post-train **Qwen3.5-9B** Team (2026) on all trajectory subtasks using our **TS-GSPO** objective together with the unified rewards defined in Sec. 3.3.1. For each prompt, we sample eight responses with temperature 1.0 and train for one epoch with a global batch size of 32 and a learning rate of 1×10^{-6}. Samples from all training datasets are jointly shuffled.

During post-training, each sample provides the observed trajectory coordinates explicitly in the textual prompt. We additionally mark the corresponding target point on each history frame and append a *summary frame*, constructed from the final history frame with the complete observed trajectory overlaid. These visual annotations serve as auxiliary grounding cues. During evaluation on history-conditioned trajectory benchmarks, however, history coordinates are provided only in textual form for comparability. Prediction horizons typically span 5–8 sampled future points. More evaluation details can be found in Appendix D.

We disable Qwen3.5's native thinking mode and adopt a unified structured response format throughout training and evaluation for our model: `<thinking>...</thinking>` followed by `<answer>` containing a JSON list of future point–label pairs, e.g., {`"point_2d": [x, y], "label": "..."`}, in temporal order. As shown in the thinking-mode ablation in Appendix C.2, this custom-tag setting achieves lower trajectory errors on most benchmarks while substantially improving structured-output reliability compared with native-thinking training. For label output, each label provides a brief semantic explanation of the predicted motion, such as "change direction due to collision" in physical scenes or "reaching for the cup" in manipulation scenes. The model is required to output exactly the requested number of future points. We denote the resulting model as **XTrajectory Model** and report checkpoint 2500's evaluation results, selected according to validation performance. All coordinate-based metrics are calculated under [0,1000] scale .

5.2 XTrajectory-Bench

We evaluate XTrajectory on our constructed XTrajectory-Bench. As shown in Table 1, XTrajectory achieves the lowest RMSE and Fréchet distance on all three subsets among the evaluated methods.

Table 1: Results on XTrajectory-Bench constructed from Apple-π, xHOT3D, and Physics-IQ. Lower RMSE and Fréchet distance are better. Best results for each metric are shown in bold. "–" denotes unavailable results.

	Apple-π		xHOT3D		Physics-IQ	
Method	RMSE ↓	Fréchet ↓	RMSE ↓	Fréchet ↓	RMSE ↓	Fréchet ↓
Qwen3.5-9B	102.67	223.02	134.30	271.55	60.42	133.13
XTrajectory	**58.48**	**116.16**	**106.77**	**209.10**	**52.35**	**113.71**
Cosmos3-Nano NVIDIA (2026)	91.24	191.40	141.40	286.88	62.22	135.70
MolmoMotion Zhang et al. (2026a)	–	–	122.74	248.36	–	–
ObjectForesight Soraki et al. (2026)	–	–	155.03	296.86	–	–
TrajLoom Zhang et al. (2026b)	103.66	206.62	144.19	273.28	93.10	187.50
Linear Extrapolation	163.91	338.56	145.43	283.17	98.16	198.86
Quadratic Extrapolation	320.33	776.88	338.25	795.89	141.58	328.10

Table 2: Results on history-free trajectory benchmarks. Qwen3.5-9B and XTrajectory are evaluated in a task-level zero-shot setting. All methods are evaluated under our unified evaluation protocol. Lower values are better for all metrics. Best results are shown in bold, and second-best results are marked with *.

	ShareRobot-Traj. Ji et al. (2025)			VABench Yuan et al. (2026a)	
Method	DFD ↓	HD ↓	RMSE ↓	Trace MAE ↓	Trace RMSE ↓
Task-level zero-shot					
Qwen3.5-9B	283.37	245.18	188.04	73.45	94.62
XTrajectory	205.25*	187.23*	140.59*	48.41*	63.18*
Cosmos3-Nano NVIDIA (2026)	252.85	225.67	164.20	50.91	67.65
RoboBrain 2.5-8B MTT Tan et al. (2026)	**194.85**	**175.71**	**135.75**	72.11	93.45
Embodied-R1.5-8B Yuan et al. (2026b)	242.77	221.10	173.74	**39.93**	**49.17**
FSD-13B Yuan et al. (2026a)	284.93	250.56	220.65	58.04	75.95

Compared with the original Qwen3.5-9B model, XTrajectory consistently improves both point-wise accuracy and global trajectory quality, demonstrating strong transfer to unseen physical and egocentric manipulation scenarios. It also outperforms both general-purpose reasoning world model and specialized trajectory predictors.

Moreover, the clear margin over extrapolation methods indicates that XTrajectory goes beyond simply extending the observed motion trend, and can use visual context and interaction dynamics to anticipate future trajectory changes.

5.3 Existing Benchmarks

Existing trajectory benchmarks. We further evaluate XTrajectory in a **zero-shot** setting on two existing trajectory benchmarks for *embodied manipulation*, as summarized in Table 2. ShareRobot-Bench Trajectory Ji et al. (2025) evaluates robot end-effector trajectory prediction, while VABench Yuan et al. (2026a) evaluates visual motion traces of manipulated objects.

Unlike our post-training setup, where the model observes multiple history frames together with the target's past trajectory, both benchmarks provide only a single image of the current scene and task instruction without explicit motion history. Despite this substantial shift in conditioning, XTrajectory markedly improves over the original Qwen3.5-9B model on both benchmarks and remains competitive with recent embodied VLMs. These zero-shot results suggest that XTrajectory does not merely rely on historical trajectories or extrapolate past motion, but can instead leverage visual context and learned world knowledge to infer plausible future motion from a single observation.

General-purpose multimodal benchmarks. Table 3 evaluates how trajectory-oriented reinforcement post-training affects the general multimodal capabilities of the base VLM. XTrajectory shows the most consistent gain on CrossPoint, improving across all three inference settings which indicates

Table 3: Results on general-purpose multimodal benchmarks. Higher is better for all metrics. We report three evaluation settings: DU uses the custom thinking tag format with native thinking disabled, while EO and DO use the official format with native thinking enabled and disabled, respectively.

	Original 9B			XTrajectory		
Benchmark	DU	EO	DO	DU	EO	DO
RoboPoint Yuan et al. (2024)	43.6	**52.0**	**56.2**	**51.8**	49.9	49.2
PointArena Cheng et al. (2025)	58.8	**73.5**	**72.4**	**68.3**	67.4	66.0
CrossPoint Wang et al. (2025)	39.2	37.9	32.9	**49.1**	**52.9**	**38.1**
ERQA Team et al. (2025)	**52.2**	**55.2**	50.8	48.5	51.8	**51.3**
MMSI Yang et al. (2026)	33.8	**35.3**	33.3	**34.1**	34.7	**34.1**
Macro Avg.	45.52	50.78	**49.12**	**50.36**	**51.34**	47.74

Table 4: RMSE comparison between GSPO and TS-GSPO. Lower is better.

Method	ShareRobot-Traj.	VABench	Physics-IQ	Apple-π	xHOT3D
GSPO	145.57	67.13	**50.63**	70.04	109.93
TS-GSPO	**140.59**	**63.18**	52.35	**58.48**	**106.77**

enhanced cross-view localization and spatial correspondence. XTrajectory also improves RoboPoint, PointArena, and MMSI under DU. At the aggregate level, the macro-average increases under DU and EO, while showing a modest decrease under DO.

Overall, these results indicate that trajectory post-training can transfer positively to related point-centric and spatial reasoning capabilities, although the gains are not uniform across inference settings and reveal a specialization–retention trade-off for broader multimodal competence. The direct-SFT ablation in Appendix C.1 further shows that, unlike direct SFT, reinforcement post-training avoids severe specialization and better preserves the base VLM's broader multimodal capabilities.

5.4 Ablation Study on TS-GSPO

To evaluate the effect of TS-GSPO, we compare it with standard trajectory-level GSPO at training step 2500 under the same training data and inference settings. TS-GSPO improves RMSE on four of the five benchmarks over standard GSPO. The gain is particularly pronounced on Apple-π. The overall trend indicates that segment-level credit assignment provides a more effective optimization signal for trajectory prediction than a single response-level advantage.

More ablations comparing SFT and different thinking-mode settings are provided in Appendix C.

6 Conclusion and Limitations

We introduced **XTrajectory**, a unified framework for learning future world dynamics in general-purpose vision-language models through structured trajectory prediction. XTrajectory represents heterogeneous future motion directly as autoregressive coordinate sequences through the VLM's native output interface. By combining diverse trajectory data with geometrically verifiable reinforcement learning, the model acquires forecasting capabilities across visual points, objects, human hands, and robot end-effectors. Experiments on existing benchmarks and XTrajectory-Bench show strong transfer to unseen manipulation and physical-dynamics scenarios, while general multimodal evaluations reveal both positive transfer and selective minor capability degradation.

Several limitations remain. The effectiveness of trajectory RL post-training is strongly influenced by the underlying base model, particularly its visual grounding, spatial reasoning priors. Finally, although the learned trajectory capability shows strong transfer across forecasting settings, its downstream benefits for planning and control have not yet been directly validated.

Despite these limitations, our results support trajectory prediction as a compact and measurable interface for incorporating predictive world dynamics into general-purpose VLMs.

AI USE STATEMENT

We used generative AI tools primarily for manuscript editing and language refinement, with occasional assistance in experiment implementation, code debugging, and identifying relevant literature. All AI-assisted outputs were reviewed and verified by the authors, and all experimental results, analyses, and scientific claims were independently checked. The authors take full responsibility for the final content of this work.

ETHICS STATEMENT

This work uses existing research datasets and does not involve new human-subject data collection or real-world autonomous deployment. Some datasets contain egocentric human activities and manipulation interactions, and are used following the conditions of their original releases. Our experiments focus on offline trajectory forecasting; deployment in safety-critical embodied systems would require additional validation for robustness, bias, and failure modes.

REPRODUCIBILITY STATEMENT

We provide the model formulation, trajectory reward, and TS-GSPO objective in Sec. 3, together with the main training and inference settings in Sec. 5.1. Detailed construction and filtering procedures for all trajectory training data and XTrajectory-Bench are documented in Appendices A and B. Appendix D specifies the evaluation protocols and adaptations used for different baselines, while Appendix E provides additional implementation details for reinforcement post-training, including reward computation, segment-level optimization, and regularization. These details are intended to enable reproduction of both the training procedure and reported evaluations.

REFERENCES

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Prithviraj Banerjee, Sindi Shkodrani, Pierre Moulon, Shreyas Hampali, Shangchen Han, Fan Zhang, Linguang Zhang, Jade Fountain, Edward Miller, Selen Basol, Richard Newcombe, Robert Wang, Jakob Julian Engel, and Tomas Hodan. HOT3D: Hand and object tracking in 3D from egocentric multi-view videos. *CVPR*, 2025.

Homanga Bharadhwaj, Roozbeh Mottaghi, Abhinav Gupta, and Shubham Tulsiani. Track2act: Predicting point tracks from internet videos enables generalizable robot manipulation. In *European Conference on Computer Vision (ECCV)*, 2024.

Gabrijel Boduljak, Laurynas Karazija, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. What happens next? anticipating future motion by generating point trajectories. In *International Conference on Learning Representations*, volume 2026, pp. 126229–126265, 2026.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *Forty-first international conference on machine learning*, 2024.

Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu Gu, et al. Robotwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. *arXiv preprint arXiv:2506.18088*, 2025.

Yi Chen, Yuying Ge, Rui Wang, Yixiao Ge, Junhao Cheng, Ying Shan, and Xihui Liu. Grpo-care: Consistency-aware reinforcement learning for multimodal reasoning. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 4305–4320, 2026.

Long Cheng, Jiafei Duan, Yi Ru Wang, Haoquan Fang, Boyang Li, Yushan Huang, Elvis Wang, Ainaz Eftekhar, Jason Lee, Wentao Yuan, et al. Pointarena: Probing multimodal grounding through language-guided pointing. *arXiv preprint arXiv:2505.09990*, 2025.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. The epic-kitchens dataset: Collection, challenges and baselines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):4125–4141, 2020.

Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, et al. Kubric: A scalable dataset generator. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3749–3761, 2022.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2(3):440, 2018.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. *arXiv preprint arXiv:1811.04551*, 2018.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models, 2024. URL https://arxiv.org/abs/2301.04104.

Adam W. Harley, Yang You, Xinglong Sun, Yang Zheng, Nikhil Raghuraman, Yunqi Gu, Sheldon Liang, Wen-Hsuan Chu, Achal Dave, Pavel Tokmakov, Suya You, Rares Ambrus, Katerina Fragkiadaki, and Leonidas J. Guibas. AllTracker: Efficient dense point tracking at high resolution. In *ICCV*, 2025.

Wenlong Huang, Yu-Wei Chao, Arsalan Mousavian, Ming-Yu Liu, Dieter Fox, Kaichun Mo, and Li Fei-Fei. Pointworld: Scaling 3d world models for in-the-wild robotic manipulation. *arXiv preprint arXiv:2601.03782*, 2026a.

Wenxuan Huang, Bohan Jia, Shaosheng Cao, Zheyu Ye, Zhe Xu, Yao Hu, Shaohui Lin, et al. Vision-r1: Incentivizing reasoning capability in multimodal large language models. In *International Conference on Learning Representations*, volume 2026, pp. 63794–63812, 2026b.

Yuheng Ji, Huajie Tan, Jiayu Shi, Xiaoshuai Hao, Yuan Zhang, Hengyuan Zhang, Pengwei Wang, Mengdi Zhao, Yao Mu, Pengju An, et al. Robobrain: A unified brain model for robotic manipulation from abstract to concrete. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1724–1734. IEEE, 2025.

Bingqing Jiang and Difan Zou. Structured role-aware policy optimization for multimodal reasoning. *arXiv preprint arXiv:2605.07274*, 2026.

Nikita Karaev, Iurii Makarov, Jianyuan Wang, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker3: Simpler and better point tracking by pseudo-labelling real videos. In *Proc. arXiv:2410.11831*, 2024a.

Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. In *Proc. ECCV*, 2024b.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Seungjae Lee, Yoonkyo Jung, Inkook Chun, Yao-Chih Lee, Zikui Cai, Hongjia Huang, Aayush Talreja, Tan Dao, Yongyuan Liang, Jia-Bin Huang, et al. Tracegen: World modeling in 3d trace space enables learning from cross-embodiment videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20721–20731, 2026.

Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi Wang. Visual-rft: Visual reinforcement fine-tuning. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2034–2044. IEEE, 2025.

Mao-Lin Luo, Zhe-Xu Wang, Zi-Hao Zhou, Bo Ye, Jian Zhao, Min-Ling Zhang, and Tong Wei. Rl forgets! towards continual policy optimization. *arXiv preprint arXiv:2607.04364*, 2026.

Saman Motamed, Laura Culp, Kevin Swersky, Priyank Jaini, and Robert Geirhos. Do generative video models understand physical principles? In *2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 948–958. IEEE, 2026.

NVIDIA. Cosmos 3: Omnimodal world models for physical ai, 2026. URL https://arxiv.org/abs/2606.02800.

Hoang Phan, Xianjun Yang, Yuanshun Yao, Jingyu Zhang, Shengjie Bi, Xiaocheng Tang, Madian Khabsa, Lijuan Liu, and Deren Lei. Beyond reasoning gains: Mitigating general-capability forgetting in large reasoning models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 34368–34398, 2026.

Ryan Punamiya, Simar Kareer, Zeyi Liu, Josh Citron, Ri-Zhao Qiu, Xiongyi Cai, Alexey Gavryushin, Jiaqi Chen, Davide Liconti, Lawrence Y. Zhu, Patcharapong Aphiwetsa, Baoyu Li, Aniketh Cheluva, Pranav Kuppili, Yangcen Liu, Dhruv Patel, Aidan Gao, Hye-Young Chung, Ryan Co, Renee Zbizika, Jeff Liu, Xiaomeng Xu, Haoyu Xiong, Geng Chen, Sebastiano Oliani, Wenkai Xuan, Chenyu Yang, Xi Wang, James Fort, Richard Newcombe, Josh Gao, Jason Chong, Garrett Matsuda, Aseem Doriwala, Marc Pollefeys, Robert Katzschmann, Xiaolong Wang, Shuran Song, Judy Hoffman, and Danfei Xu. Egoverse: An egocentric human dataset for robot learning from around the world, 2026. URL https://arxiv.org/abs/2604.07607.

Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun Zhang, Kangjia Zhao, Qianqian Zhang, et al. Vlm-r1: A stable and generalizable r1-style large vision-language model. *arXiv preprint arXiv:2504.07615*, 2025.

Rustin Soraki, Homanga Bharadhwaj, Ali Farhadi, and Roozbeh Mottaghi. Objectforesight: Predicting future 3d object trajectories from human videos. *arXiv preprint arXiv:2601.05237*, 2026.

Huajie Tan, Enshen Zhou, Zhiyu Li, Yijie Xu, Yuheng Ji, Xiansheng Chen, Cheng Chi, Pengwei Wang, Huizhu Jia, Yulong Ao, Mingyu Cao, Sixiang Chen, Zhe Li, Mengzhen Liu, Zixiao Wang, Shanyu Rong, Yaoxu Lyu, Zhongxia Zhao, Peterson Co, Yibo Li, Yi Han, Shaoxuan Xie, Guocai Yao, Songjing Wang, Leiduo Zhang, Xi Yang, Yance Jiao, Donghai Shi, Kunchang Xie, Shaokai Nie, Chunlei Men, Yonghua Lin, Zhongyuan Wang, Tiejun Huang, and Shanghang Zhang. Robobrain 2.5: Depth in sight, time in mind. *arXiv preprint arXiv:2601.14352*, 2026.

Gemini Robotics Team, Saminda Abeyruwan, Joshua Ainslie, Jean-Baptiste Alayrac, Montserrat Gonzalez Arenas, Travis Armstrong, Ashwin Balakrishna, Robert Baruch, Maria Bauza, Michiel Blokzijl, et al. Gemini robotics: Bringing ai into the physical world. *arXiv preprint arXiv:2503.20020*, 2025.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Neerja Thakkar, Shiry Ginosar, Jacob C Walker, Jitendra Malik, João Carreira, and Carl Doersch. Forecasting motion in the wild. In *European Conference on Computer Vision (ECCV)*, 2026.

Zhongwei Wan, Zhihao Dou, Che Liu, Yu Zhang, Dongfei Cui, Qinjian Zhao, Hui Shen, Jing Xiong, Yi Xin, Yifan Jiang, et al. Srpo: Enhancing multimodal llm reasoning via reflection-aware reinforcement learning. *Advances in Neural Information Processing Systems*, 38:153676–153713, 2026.

Yipu Wang, Yuheng Ji, Yuyang Liu, Enshen Zhou, Ziqiang Yang, Yuxuan Tian, Ziheng Qin, Yue Liu, Huajie Tan, Cheng Chi, Zhiyuan Ma, Daniel Dajun Zeng, and Xiaolong Zheng. Towards cross-view point correspondence in vision-language models. *arXiv preprint arXiv:2512.04686*, 2025.

Chuan Wen, Xingyu Lin, John So, Kai Chen, Qi Dou, Yang Gao, and Pieter Abbeel. Any-point trajectory modeling for policy learning. *arXiv preprint arXiv:2401.00025*, 2023.

Fan Yang, Zhiyang Chen, Yousong Zhu, Xin Li, and Jinqiao Wang. From seeing to predicting: A vision-language framework for trajectory forecasting and controlled video generation. *arXiv preprint arXiv:2510.00806*, 2025.

Sihan Yang, Runsen Xu, Yiman Xie, Sizhe Yang, Mo Li, Jingli Lin, Chenming Zhu, Xiaochen Chen, Haodong Duan, Xiangyu Yue, et al. Mmsi-bench: A benchmark for multi-image spatial intelligence. In *International Conference on Learning Representations*, volume 2026, pp. 157051–157088, 2026.

Runmao Yao, Kairui Hu, Yukang Cao, Ruisi Wang, Shulin Tian, Ziang Cao, Weichen Fan, Ziqi Huang, Yuhao Dong, Hao Li, et al. Apple-π: Benchmarking thinking with video towards law-grounded physical intelligence. *arXiv preprint arXiv:2607.16401*, 2026.

Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B. Tenenbaum. CLEVRER: collision events for video representation and reasoning. In *ICLR*, 2020.

En Yu, Liang Zhao, Yana Wei, Jinrong Yang, Dongming Wu, Lingyu Kong, Haoran Wei, Tiancai Wang, Zheng Ge, Xiangyu Zhang, and Wenbing Tao. Merlin: Empowering multimodal llms with foresight minds. *arXiv preprint arXiv:2312.00589*, 2023.

Wentao Yuan, Jiafei Duan, Valts Blukis, Wilbert Pumacay, Ranjay Krishna, Adithyavairavan Murali, Arsalan Mousavian, and Dieter Fox. Robopoint: A vision-language model for spatial affordance prediction for robotics. *arXiv preprint arXiv:2406.10721*, 2024.

Yifu Yuan, Haiqin Cui, Yibin Chen, Zibin Dong, Fei Ni, Longxin Kou, Jinyi Liu, Pengyi Li, Yan Zheng, and Jianye Hao. From seeing to doing: Bridging reasoning and decision for robotic manipulation. *The Fourteenth International Conference on Learning Representations*, 2026a.

Yifu Yuan, Yaoting Huang, Xianze Yao, Yutong Li, Shuoheng Zhang, Linqi Han, Pengyi Li, Jiangeng Sun, Wenting Jia, Zhao Zhang, Yuhao Liu, Ruihao Liao, Yucheng Hu, Qiyu Wu, Yuxiao Li, Zibin Dong, Fei Ni, Yan Zheng, Shuyang Gu, Yi Ma, Hongyao Tang, Han Hu, and Jianye Hao. Embodied-r1.5: Evolving physical intelligence via embodied foundation models. *arXiv preprint*, 2026b.

Jianing Zhang, Chenhao Zheng, Yajun Yang, Max Argus, Rustin Soraki, Winson Han, Taira Anderson, Chun-Liang Li, Shuo Liu, Jiafei Duan, et al. Molmomotion: Forecasting point trajectories in 3d with language instruction. *arXiv preprint arXiv:2606.18558*, 2026a.

Jingyi Zhang, Jiaxing Huang, Huanjin Yao, Shunyu Liu, Xikun Zhang, Shijian Lu, and Dacheng Tao. R1-vl: Learning to reason with multimodal large language models via step-wise group relative policy optimization. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1859–1869. IEEE, 2025.

Zewei Zhang, Jia Jun Cheng Xian, Kaiwen Liu, Ming Liang, Hang Chu, Jun Chen, and Renjie Liao. Trajloom: Dense future trajectory generation from video. *arXiv preprint arXiv:2603.22606*, 2026b.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization, 2025. URL https://arxiv.org/abs/2507.18071.

Zhicheng Zheng, Xin Yan, Zhenfang Chen, Jingzhou Wang, Qin Zhi Eddie Lim, Joshua B Tenenbaum, and Chuang Gan. Contphy: Continuum physical concept learning and reasoning from videos. In *International Conference on Machine Learning*. PMLR, 2024.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning. *arXiv preprint arXiv:2411.04983*, 2024.

Siyuan Zhou, Hejun Wang, Hu Cheng, Jinxi Li, Dongsheng Wang, Junwei Jiang, Yixiao Jin, Jiayue Huang, Shiwei Mao, Shangjia Liu, Yafei Yang, Hongkang Song, Shenxing Wei, Zihui Zhang, Peng Huang, Shijie Liu, Zhengli Hao, Hao Li, Yitian Li, Wenqi Zhou, Zhihan Zhao, Zongqi He, Hongtao Wen, Shouwang Huang, Peng Yun, Bowen Cheng, Pok Kazaf Fu, Wai Kit Lai, Jiahao Chen, Kaiyuan Wang, Zhixuan Sun, Ziqi Li, Haochen Hu, Di Zhang, Chun Ho Yuen, Bing Wang, Zhihua Wang, Chuhang Zou, and Bo Yang. Physinone: Visual physics learning and reasoning in one suite. *CVPR*, 2026.

Appendix

A Trajectory Training Data Construction

We convert all source datasets into a unified history-to-future trajectory representation. The selected datasets expose the model to complementary sources of future motion: generic and interaction-driven physical dynamics, egocentric human–object interaction, and instruction-conditioned robotic manipulation.

A.1 EgoVerse

EgoVerse Punamiya et al. (2026) provides instruction-conditioned egocentric hand motion, exposing the model to human action trajectories that connect task intent with future manipulation behavior. We only utilize a small part of the original dataset for data generation: our source contains **2,252** local Zarr episodes. Each episode provides front-camera RGB images, camera/head poses, timestamps, and 21 world-space MANO keypoints for each available hand. We represent each hand by MANO keypoint 0, corresponding to the wrist. The world-space wrist positions are transformed using the camera poses and projected onto the camera plane of the last history frame, providing a common coordinate system for both observed and future motion.

We sample every fifth raw frame and divide each sequence into non-overlapping windows containing five history points and five future points. When both hands are available, the hand with the larger history path length on the anchor plane is selected using history information only. Only the five observed RGB frames are retained as visual input.

We reject samples with invalid 3D points or projections, decoding failures, inconsistent image geometry, or temporal discontinuities. Full-rate trajectories are checked for tracking spikes, and the first future point must remain consistent with a velocity estimate from the recent history. We further require sufficient motion over both the observed history and the full forecasting horizon, reject cases in which motion begins only after the observation period, and require all future points to remain within the anchor image.

A.2 ObjectForesight / EPIC

ObjectForesight/EPIC Soraki et al. (2026); Damen et al. (2020) provides natural egocentric hand and manipulated-object motion without relying on explicit task instructions, encouraging future prediction from observed interaction context and latent human intent. ObjectForesight/EPIC source contains **29,006** processed clips with both the required RGB video and SpaTracker data. Each clip provides RGB video, SpaTracker-derived depth and camera information, object masks, and optional two-hand masks. Objects are represented by segmentation-mask centroids. For hands, small connected components are removed and observations are temporally associated into persistent left/right tracks to reduce occasional label swaps.

Each 2D centroid is lifted into 3D using local depth, transformed through the camera poses, and reprojected onto the camera plane of the last history frame. This expresses the complete trajectory in a common coordinate system and reduces the direct effect of egocentric camera motion.

We construct windows with five history and five future points. Sampling first uses a frame stride of five and progressively reduces the stride to one only when no complete valid window exists at the current stride. Candidate windows are scored using recent history motion, future displacement, and trajectory path length, and the highest-scoring trajectory across objects and hands is retained for each clip. Thus, trajectory selection uses both observed and future motion during dataset construction.

We require valid mask centers, positive depth, and finite anchor-camera projections throughout the window. Hand-specific processing removes unreliable components, implausible temporal associations, and long missing intervals, while excessively large sampled jumps are rejected in both native and anchor image planes. Motion thresholds are used to prefer informative windows rather than as strict rejection criteria when no candidate satisfies them.

Table 5: Number of training samples per dataset.

Dataset	Samples
Kubric	10,870
CLEVRER	21,670
RoboTwin	11,494
EgoVerse	31,693
ObjectForesight/EPIC	25,373
PhysInOne	8,673
Ball Playground (ContPhy)	1,055
Total	**110,828**

A.3 ROBOTWIN

RoboTwin Chen et al. (2025) complements human manipulation data with instruction-conditioned robot end-effector motion, providing trajectories that connect task-level goals with robotic action execution.

We generate **22,481** HDF5 episodes across 60 task directories with RoboTwin 2.0 data generator. The current conversion pipeline operates on **19,119** generated trajectory records spanning 39 tasks, containing frame-aligned 2D gripper/end-effector trajectories, image references, original frame indices, and optional gripper states.

We retain samples with exactly one physically active arm. Intermediate gripper-transition frames are removed, while frames corresponding to completed open/close transitions are preserved. Stationary trajectory segments are then compressed using a small displacement threshold, and the remaining real frames are sampled approximately uniformly according to cumulative 2D path distance rather than uniformly in time. Trajectory endpoints and mandatory gripper-transition points are preserved during this process.

The final five retained points are used as future targets and all preceding points form the observed history. A nominal budget of ten retained points therefore typically produces five history and five future points, although mandatory transition points may increase the total sequence length. Coordinates are converted from the source image convention to (x, y) and normalized to the $[0, 1000]$ range. Only media referenced by the final retained trajectory are used.

A.4 KUBRIC

Kubric Greff et al. (2022) provides generic arbitrary-point trajectories, allowing the model to learn broad visual motion patterns without restricting the prediction target to a predefined semantic object or agent. We generate **11,000** Kubric source clips following CoTracker Karaev et al. (2024b), each with RGB frames and corresponding 2D point tracks and visibility annotations. These trajectories correspond to arbitrary visual points rather than predefined semantic objects or agents.

We enumerate sliding windows over all available tracks directly in the native image plane. The default configuration samples every second source frame and uses six history points followed by five future points. Every possible window start is considered, and the highest-scoring valid window is retained for each track before selecting the best trajectory for the video.

All points in the complete history–future window must remain inside the image. We require sufficient displacement in both history and future, reject trajectories with very small observed motion or excessively large sampled steps, and use constant-velocity extrapolation error to remove highly irregular tracks. Ranking additionally favors informative motion, visibility, and distance from image boundaries while penalizing abrupt jumps. When multiple trajectories are requested from the same video, trajectories with overly similar spatial locations or normalized motion shapes are removed.

A.5 CLEVRER

CLEVRER Yi et al. (2020) provides collision-centered object motion in which future trajectories can change abruptly due to inter-object interactions, thereby discouraging simple continuation of

the observed motion. Its training set contains **10,000** videos together with collision annotations and per-frame object-mask annotations. We construct collision-centered forecasting samples around annotated physical interactions.

For each collision, we select one participating object whose color, material, and shape uniquely identify it in the scene. The corresponding segmentation mask is decoded at every selected frame, and its foreground centroid is used as the 2D trajectory point. Coordinates remain in the native image plane, without camera-plane reprojection.

The original 25 FPS videos are sampled at 5 FPS. In the configuration used for training, the observed history ends three source frames before the annotated collision, while the future sequence starts at the collision frame and contains eight sampled points. Up to ten pre-collision history points are retained, with at least three required. This construction exposes the model to trajectory changes induced by physical interactions rather than only smooth motion continuation.

We reject samples with insufficient temporal coverage, missing collision or mask information, ambiguous object identities, or empty masks at any required timestep. At most one trajectory is retained per collision. No additional displacement, smoothness, or motion-ranking filter is applied.

A.6 PHYSINONE

PhysInOne Zhou et al. (2026) contains 5,340 Blender-rendered physical scenes, each observed from approximately 13 fixed camera viewpoints. Each video contains 150 frames at 30 FPS and provides frame-wise instance segmentation masks. For each scene–view–object combination, we use the centroid of the object mask as the 2D trajectory point.

We uniformly subsample each video and construct samples with six history points and six future points. Trajectories are filtered based on motion magnitude, temporal consistency, object visibility, and viewpoint redundancy. We remove nearly stationary trajectories, abrupt trajectory jumps, and cases with insufficient or unstable object visibility. When the same object yields valid trajectories from multiple viewpoints, only a small number of views are retained to reduce redundancy.

This procedure produces 8,673 samples, with final coordinates normalized to the $[0, 1000]$ range.

A.7 BALL PLAYGROUND (CONTPHY)

We use the **Ball Playground** subset of ContPhy Zheng et al. (2024), which contains 500 Unity-rendered soft-body interaction videos. Each scene contains three balls with different material properties falling from above and undergoing rolling and collision dynamics among ramps, barriers, and floor gaps. Each video contains 118 frames at 30 FPS and provides a 2D bounding box for every ball at each annotated frame; we use the bounding-box center as the trajectory point. We observe a fixed two-frame offset between the video and annotation streams and verify this alignment by cross-checking segmentation-derived centroids against the bounding boxes.

We sample trajectories every four source frames, corresponding to 7.5 FPS, and construct windows with six history and six future points, spanning approximately 1.5 seconds. Because most meaningful motion occurs near the beginning of each video, candidate windows are enumerated with a two-frame start stride. In addition to the motion criteria used for PhysInOne, we apply ball-size-normalized geometric constraints to remove trajectories dominated by local bouncing or near-stationary motion. Specifically, the future trajectory is required to cover a sufficiently large spatial extent relative to the ball diameter, while the last four history points must also exhibit sufficient motion to exclude cases in which the ball becomes trapped shortly after entering the scene. Near-stationary motion is permitted only at the final future step, and future points are required to remain sufficiently separated to remove overlapping trajectories caused by repeated local bouncing. The prediction window must also end no later than the estimated stopping time of the ball.

All 12 trajectory points are additionally required to remain sufficiently far from the image boundary, since partially visible balls lead to cropped bounding boxes and systematically biased center estimates. When this condition is violated, an earlier valid window is preferred rather than discarding the trajectory. We retain at most two windows per trajectory: the earliest valid window and, when available, the latest valid window separated sufficiently in time to reduce redundancy. This results

in 1,055 samples. Coordinate normalization, input formatting, and history-trajectory visualization follow the same procedure as PhysInOne.

B XTRAJECTORY-BENCH CONSTRUCTION

This section provides the detailed data construction and preprocessing procedures for the three subsets of XTrajectory-Bench. All subsets are converted into the unified history-to-future trajectory prediction format used throughout the paper.

B.1 PHYSICS-IQ

Physics-IQ Motamed et al. (2026) contains 396 real-world videos spanning diverse physical phenomena. We select 80 videos with clear and trackable rigid-object motion and convert them into future trajectory prediction samples.

For each video, Qwen3.5-27B is first used to identify a useful motion interval, select a salient moving object, and localize an initial tracking point. We then use CoTracker3 Karaev et al. (2024a) or AllTracker Harley et al. (2025) to track the selected point throughout the interval. The resulting trajectories are manually verified, and samples with invalid tracking, insufficient motion, or stationary trajectory tails are removed.

For each retained trajectory, the final five sampled points are used as the future ground truth, while the preceding frames and coordinates form the observation history. When the history is long, frames are temporally downsampled to fewer than ten observations. This produces 80 trajectory forecasting examples covering diverse real-world physical interactions.

B.2 APPLE-π

Apple-π Yao et al. (2026) is built on the *Orchard* dataset, which contains 400 videos spanning 10 classical-mechanics tasks and provides instance-segmentation annotations. We use these annotations to construct object-centric trajectory forecasting examples. We remove the videos with stationary objects and Youtube-sourced videos.

For each video, we compute the image-plane centroid of each visible object mask over time to obtain candidate object trajectories. We select objects with sufficient visibility and motion, and construct candidate history–future windows along their trajectories. Candidate windows are ranked according to observed motion, future displacement, and total path length, and the highest-ranked valid window is retained.

The resulting sample provides the model with the history RGB frames together with the observed target coordinates, while the future coordinates are withheld as prediction targets. This construction allows the original physical-reasoning videos to be evaluated under the same trajectory forecasting interface as the other subsets.

B.3 XHOT3D

We construct xHOT3D from HOT3D Banerjee et al. (2025), which provides egocentric RGB observations together with camera calibration, object poses, and object meshes. We form temporal windows containing five history frames and five future frames.

For each object, its 3D center is transformed according to the annotated object pose and projected onto the camera plane of the last history frame. Using the last observed camera as a common reference frame reduces the effect of egocentric camera motion and produces a consistent 2D trajectory representation across the history and future frames.

We retain one valid object trajectory per clip based on motion magnitude and projection quality. Each final sample contains five history RGB frames and the corresponding observed target coordinates, while the five future coordinates are withheld for evaluation. Coordinates are normalized to the $[0, 1000]$ range following the unified trajectory representation used by XTrajectory.

Table 6: Comparison with direct SFT. Lower is better for trajectory errors and higher is better for general-purpose benchmarks.

Benchmark	Metric	SFT	XTrajectory
Trajectory prediction			
ShareRobot-Traj.	RMSE ↓	179.40	**140.59**
VABench	Trace RMSE ↓	121.03	**63.18**
Physics-IQ	RMSE ↓	60.28	**52.35**
Apple-π	RMSE ↓	71.85	**58.48**
xHOT3D	RMSE ↓	106.93	**106.77**
General-purpose multimodal capability			
RoboPoint	Acc. ↑	11.41	**51.8**
PointArena	Acc. ↑	23.42	**68.3**
CrossPoint	Acc. ↑	1.00	**49.1**
ERQA	Acc. ↑	0.0	**48.5**
MMSI	Acc. ↑	0.0	**34.1**

C Additional Ablations

C.1 Comparison with Direct SFT

We compare XTrajectory with a direct SFT baseline trained for two epochs on the same trajectory data with a learning rate of 1×10^{-5}. Native thinking is disabled, and SFT directly supervises ground-truth future coordinates without reasoning or semantic labels.

As shown in Table 6, direct SFT learns competitive coordinate prediction when evaluation closely matches the history-conditioned training format, but strongly specializes to coordinate-only outputs, leading to trajectory-length errors and severe degradation on general-purpose multimodal tasks. This dependence is also evident on ShareRobot-Bench and VABench, which do not provide explicit trajectory history. In contrast, XTrajectory transfers more effectively to these history-free settings by combining verifiable trajectory rewards with explicit reasoning, enabling greater use of visual context, task semantics, and interaction cues. Overall, XTrajectory achieves a better balance between trajectory generalization and retention of general-purpose multimodal capability than direct SFT.

C.2 Ablation on Official Thinking Mode

We compare two Qwen3.5-9B checkpoints at training step 2000: one trained with the model's official thinking mode and one trained with explicit `<thinking>` and `<answer>` tags while official thinking is disabled. The former is evaluated with official thinking enabled (EO) or disabled (DO), while the latter uses the custom-tag format with official thinking disabled (DU).

As shown in Table 7, the custom-tag setting (DU) provides the strongest overall trade-off between trajectory quality and output reliability. Since all trajectory metrics are computed on the same subset of samples that are valid under all three settings, the observed differences reflect prediction quality rather than differences in parser coverage. Under this paired comparison, DU achieves the best trajectory errors on most benchmarks, with particularly consistent gains on the out-of-distribution subsets of XTrajectory-Bench.

The distinction is more pronounced in output validity. Enabling official thinking frequently leads to incomplete or overlong generations, substantially reducing the number of parseable predictions, especially on out-of-distribution XTrajectory-Bench. Disabling official thinking improves validity, while the custom-tag recipe further combines high parser coverage with stronger trajectory accuracy in most settings. Overall, these results motivate using explicit task-controlled reasoning tags instead of the backbone's native thinking mode for trajectory prediction.

Table 7: Thinking-mode ablation at step 2000. Errors are computed on samples valid in all three settings: 183 for ShareRobot, 267 for VABench, 170 for Apple-π, 204 for xHOT3D, and 26 for Physics-IQ. Parsed counts refer to each full evaluation set. ShareRobot errors are multiplied by 1000. Lower error and higher parsed counts are better.

		Official-thinking training		Custom-tag training
Benchmark	Metric	EO	DO	DU
ShareRobot-Traj.	DFD ↓	211.77	226.39	**197.35**
	RMSE ↓	141.60	160.60	**135.28**
	Parsed / total ↑	183/200	**200/200**	**200/200**
VABench	Trace MAE ↓	**51.06**	54.30	54.27
	Trace RMSE ↓	**66.92**	70.48	70.83
	Parsed / total ↑	269/300	298/300	**300/300**
Apple-π	RMSE ↓	85.29	90.27	**79.20**
	Fréchet ↓	170.61	180.88	**160.37**
	Parsed / total ↑	192/339	310/339	**339/339**
xHOT3D	RMSE ↓	112.29	115.15	**98.96**
	Fréchet ↓	210.29	219.11	**189.91**
	Parsed / total ↑	212/403	394/403	**402/403**
Physics-IQ	RMSE ↓	79.75	89.16	**66.25**
	Fréchet ↓	166.92	187.58	**138.88**
	Parsed / total ↑	27/80	77/80	**79/80**

D Evaluation Details

General protocol. All models are evaluated with deterministic decoding using temperature 0. For XTrajectory-Bench, the input consists of the observed history frames together with the corresponding history trajectory coordinates provided in the textual prompt. No future coordinates are exposed to the evaluated model. For ShareRobot-Bench and VABench, the model receives the input image, task instruction, and the coordinate of the initial query point.

3D trajectory baselines. MolmoMotion and ObjectForesight natively operate on 3D trajectory coordinates. We therefore evaluate them only on xHOT3D, where 3D annotations and camera calibration are available. Their inputs are constructed from the observed ground-truth 3D history coordinates following their native interfaces. The predicted 3D trajectories are projected back to the benchmark's 2D image plane using the corresponding camera parameters, after which the same 2D trajectory metrics are applied as for the other methods.

Temporal alignment. Several models use fixed history or prediction horizons that differ from the evaluation benchmarks. ObjectForesight expects four history frames and seven future frames, whereas xHOT3D uses a $5{+}5$ split. We therefore provide the last four observed frames and evaluate only the first five predicted future points. For MolmoMotion and TrajLoom, whose native temporal discretizations differ more substantially, we uniformly sample the benchmark target timestamps within the temporal range covered by the model output and obtain the required trajectory points through linear interpolation. For ShareRobot-Bench and VABench, when the model predicts different numbers of points, we follow the ShareRobot-Bench evaluation convention and compute metrics over the first $n = \min(N_{\text{pred}}, N_{\text{gt}})$ temporally matched points.

Trajectory-set adaptation. TrajLoom requires 390 observed trajectories as input rather than a single query trajectory. We construct the additional trajectories using DIS optical flow and include the benchmark query trajectory among them. Only the designated benchmark trajectory is used for evaluation.

All adaptation rules are fixed for each method and applied uniformly across samples, without selecting predictions or interpolation points based on future ground truth.

E RL IMPLEMENTATION DETAILS

This section provides implementation details omitted from Sec. 3.3 for clarity.

Reward implementation. $R_{\text{fmt}} \in \{0, 1\}$ verifies the task-specific response structure, including the expected trajectory length and exactly one

`<thinking>...</thinking><answer>...</answer>`

sequence, with only whitespace permitted outside the tagged response. $R_{\text{think}} \in \{0, 1\}$ checks whether the extracted reasoning trace contains at least 300 characters after stripping leading and trailing whitespace; internal whitespace counts toward this threshold. It encourages reasoning presence and length rather than explicitly evaluating reasoning correctness.

For the direction reward, let $\Delta\hat{\mathbf{p}} = \hat{\mathbf{p}}_T - \hat{\mathbf{p}}_1$, $\Delta\mathbf{p} = \mathbf{p}_T - \mathbf{p}_1$, and $\epsilon_d = 10^{-6}$. We use

$$R_{\text{dir}} = \begin{cases} 0, & T < 2, \\ 0, & \|\Delta\mathbf{p}\|_2 > \epsilon_d \text{ and } \|\Delta\hat{\mathbf{p}}\|_2 \le \epsilon_d, \\ \text{clip}\left(\dfrac{\cos(\Delta\hat{\mathbf{p}}, \Delta\mathbf{p}) + 1}{2}, 0, 1\right), & \|\Delta\mathbf{p}\|_2 > \epsilon_d \text{ and } \|\Delta\hat{\mathbf{p}}\|_2 > \epsilon_d, \\ \exp\left(-\dfrac{\|\Delta\hat{\mathbf{p}}\|_2}{120}\right), & \text{otherwise.} \end{cases} \tag{13}$$

The last branch is triggered by negligible ground-truth net displacement; this criterion concerns first-to-last displacement rather than requiring every intermediate point to be stationary.

Coordinate span mapping and segment objective. The segment advantage $\tilde{A}_{i,t}$ is defined for a future coordinate rather than for the entire generated response. To apply this signal during policy optimization, we first identify the tokens that correspond to each predicted coordinate in the serialized output. Specifically, $S_{i,t}$ denotes the token span overlapping the substring $[\hat{x}_{i,t}, \hat{y}_{i,t}]$ for the t-th future point. Because token boundaries do not necessarily align with character boundaries, the mapped span may also include nearby punctuation such as brackets or commas.

We then compute a segment-level importance ratio by averaging the token-wise log probability ratios over this coordinate span:

$$\rho_{i,t} = \exp\left[\frac{1}{|S_{i,t}|} \sum_{j \in S_{i,t}} \log \frac{\pi_\theta(y_{i,j} \mid x, y_{i,<j})}{\pi_{\text{old}}(y_{i,j} \mid x, y_{i,<j})}\right]. \tag{14}$$

Using the segment advantage $\tilde{A}_{i,t}$ defined in Sec. 3.3.2, we construct a clipped surrogate analogous to the global GSPO objective:

$$\ell_{i,t}^{\text{seg}} = -\min\left[\rho_{i,t}\tilde{A}_{i,t}, \text{clip}(\rho_{i,t}, 1 - \epsilon_{\text{low}}, 1 + \epsilon_{\text{high}})\,\tilde{A}_{i,t}\right], \tag{15}$$

where $\epsilon_{\text{low}} = 3 \times 10^{-4}$ and $\epsilon_{\text{high}} = 4 \times 10^{-4}$.

Not every generated coordinate can always be mapped cleanly to a token span. We therefore define $m_{i,t} \in \{0, 1\}$ to indicate whether the mapping for future step t is valid. The segment loss averages over the expected future steps of each sample and then over the batch:

$$\mathcal{L}_{\text{seg}} = \frac{1}{B} \sum_{i=1}^{B} \frac{1}{T_i} \sum_{t=1}^{T_i} m_{i,t}\, \ell_{i,t}^{\text{seg}}. \tag{16}$$

This normalization ensures that samples with more successfully mapped coordinate tokens do not receive disproportionately larger optimization weight.

Global and segment optimization. The global branch uses the sequence-level GSPO objective with A_i^{global}, while the segment branch independently uses $\tilde{A}_{i,t} = 0.5A_i^{\text{global}} + 0.5A_{i,t}^{\text{seg}}$. Both objectives use the same asymmetric clipping range. They are added with equal loss weights:

$$\mathcal{L} = \mathcal{L}_{\text{GSPO}} + \mathcal{L}_{\text{seg}} + \beta\mathcal{L}_{\text{KL}}, \qquad \beta = 0.001. \tag{17}$$

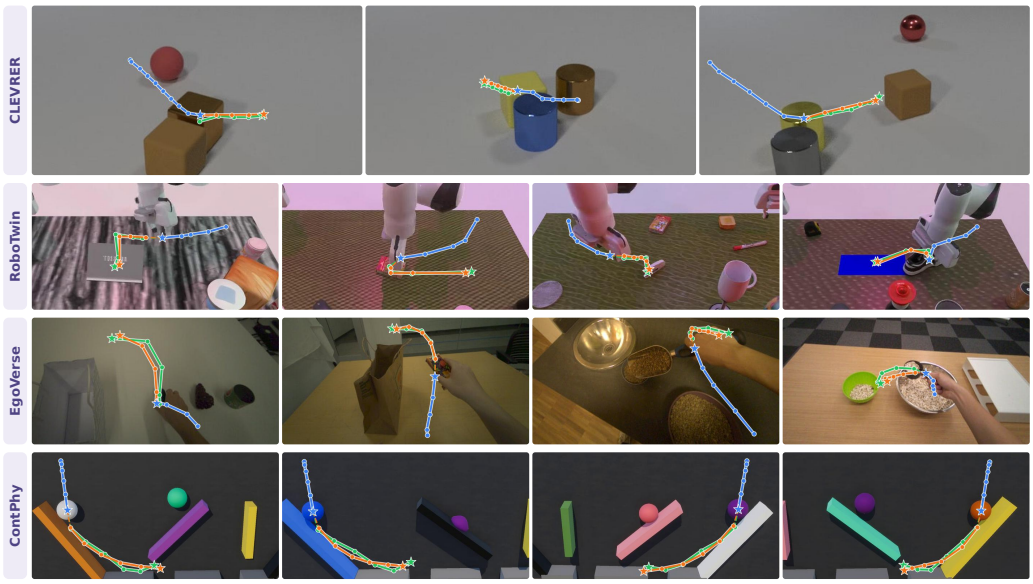

Figure 4: Training rollout examples from CLEVRER, RoboTwin, EgoVerse, and ContPhy (top to bottom). Blue curves show observed trajectories; orange and green curves show predicted and ground-truth future trajectories, respectively.

The KL term uses the actor-side low-variance KL estimator and is applied separately from the reward.

Tokens overlapping mapped coordinate spans participate in both the global and segment objectives. Tokens outside these spans—including reasoning, labels, and other response content when not overlapped by the span mapping—receive no segment objective, but remain part of the global GSPO objective. All response tokens are additionally subject to the configured KL regularization.

F Qualitative Examples

We present training rollouts, held-out predictions, and side-by-side comparisons to illustrate the trajectory behaviors captured by XTrajectory. Throughout this section, blue denotes observed history, orange denotes model predictions, and green denotes ground-truth future trajectories.

F.1 Rollout Examples

Figure 4 shows sampled training rollouts spanning collision-driven object motion, robot manipulation, egocentric hand motion, and soft-body dynamics.

F.2 Evaluation Set Examples

Figures 5 and 6 show held-out history-conditioned predictions on physical and egocentric scenes. Figures 7 and 8 show instruction-conditioned manipulation trajectories on VABench and ShareRobot.

F.3 Detailed CoT and Label Output Examples

Figures 9 and 10 provide detailed examples of the generated reasoning traces and semantic trajectory labels. Unlike the trajectory-only visualizations above, these examples expose the complete model response, illustrating how XTrajectory relates visual context, motion history or task instruction, and future coordinate prediction.

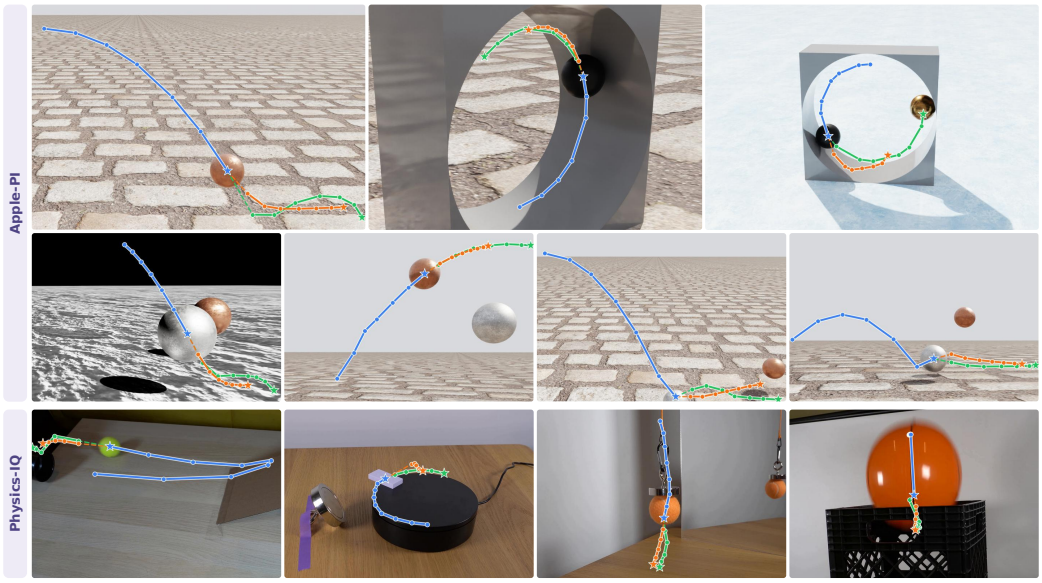

Figure 5: Evaluation examples from Apple-π (upper two rows) and Physics-IQ (bottom row). Blue denotes observed history, orange the predicted future, and green the ground-truth future.

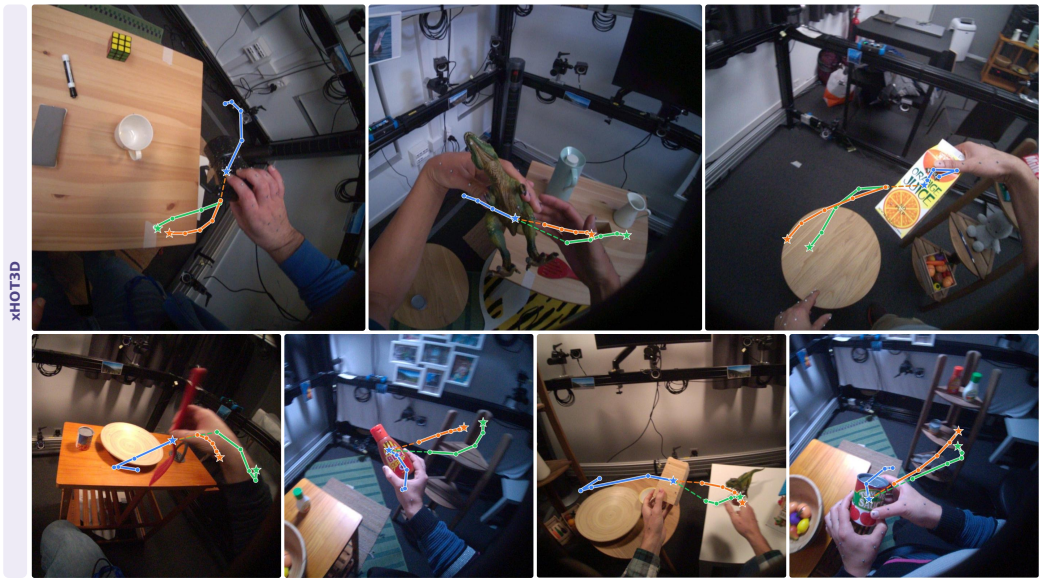

Figure 6: Evaluation examples from xHOT3D showing tracked-point trajectories during egocentric object manipulation. Observed history (blue), predictions (orange), and ground-truth futures (green) are displayed in the reference image plane.

F.4 Comparison with Other Methods

Figures 11 and 12 compare predictions for the same scenes with several other methods.

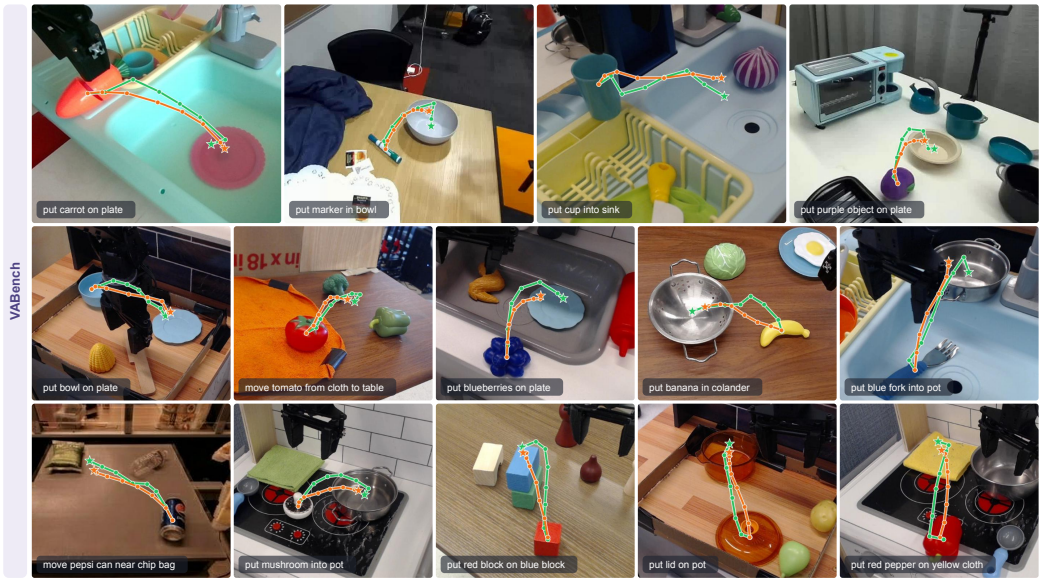

Figure 7: Instruction-conditioned trajectory predictions on VABench. Each panel displays the manipulation instruction and overlays the predicted manipulated-object path (orange) with the ground-truth path (green).

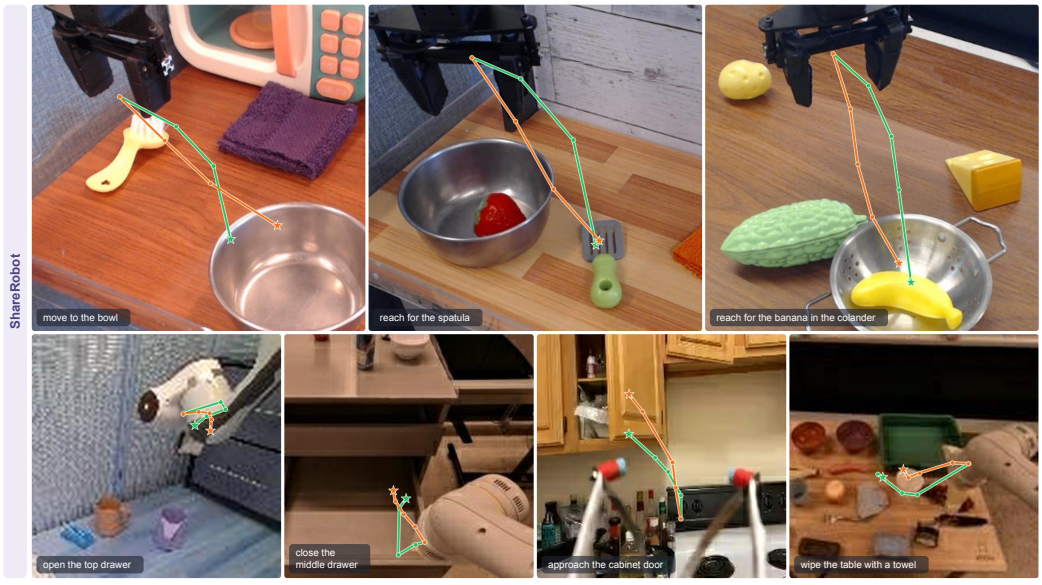

Figure 8: Instruction-conditioned trajectory predictions on ShareRobot. Orange and green denote predicted and ground-truth paths, respectively.

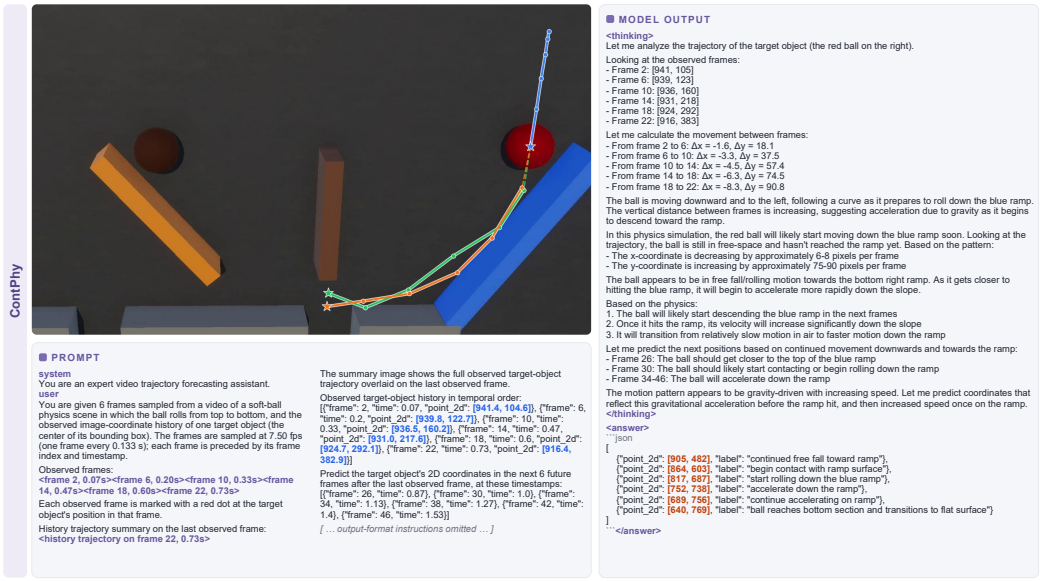

Figure 9: Detailed reasoning and trajectory-label output on a ContPhy example. Given the observed frames, trajectory-history coordinates, and prediction timestamps, XTrajectory identifies the ball's downward motion and anticipates its subsequent interaction with the ramp. The generated future points are accompanied by semantic labels describing successive motion stages, including contact, rolling, and acceleration. Blue, orange, and green denote observed history, model prediction, and ground-truth future trajectory, respectively.

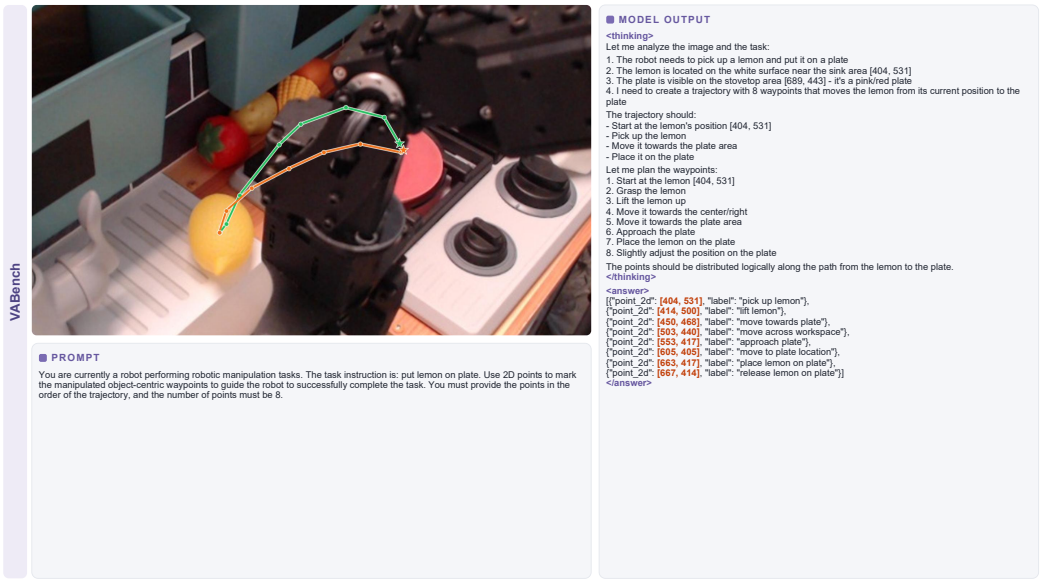

Figure 10: Detailed reasoning and trajectory-label output on a VABench manipulation example. Given only the current scene and the instruction to place the lemon on the plate, XTrajectory localizes the relevant objects, decomposes the manipulation into intermediate stages, and predicts a sequence of object-centric waypoints from pickup to placement. Each predicted point is paired with a semantic label describing the corresponding manipulation step. Orange and green denote the model prediction and ground-truth trajectory, respectively.

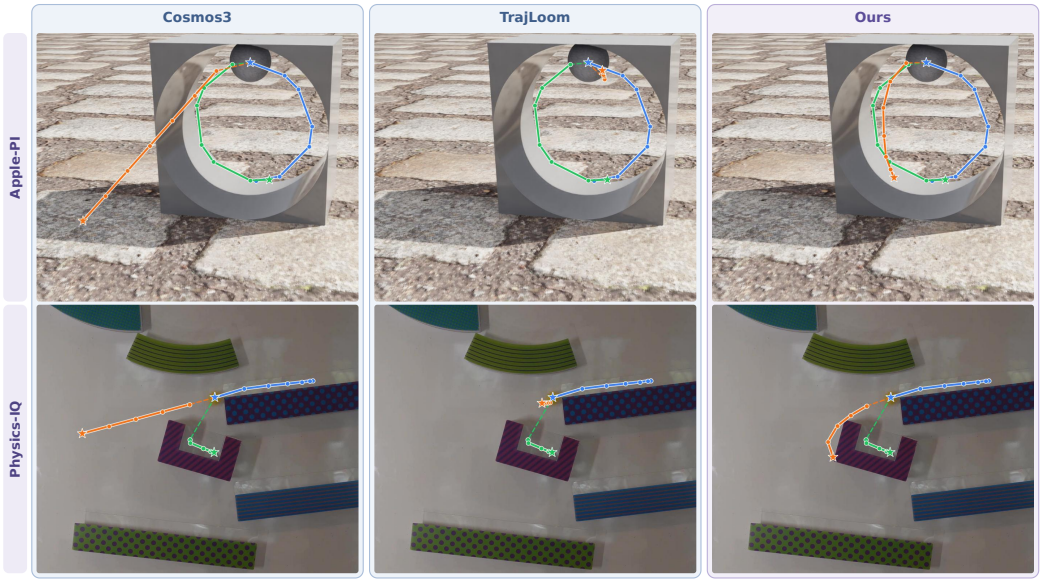

Figure 11: Comparison of Cosmos3, TrajLoom, and XTrajectory on an Apple-π example (top) and a Physics-IQ example (bottom). Blue and green show the shared observed and ground-truth trajectories; orange shows each method's prediction. Cosmos3 produces long, approximately straight continuations, while TrajLoom's predictions remain near the last observation. XTrajectory predicts turning continuations, although their locations and extents still differ from ground truth.

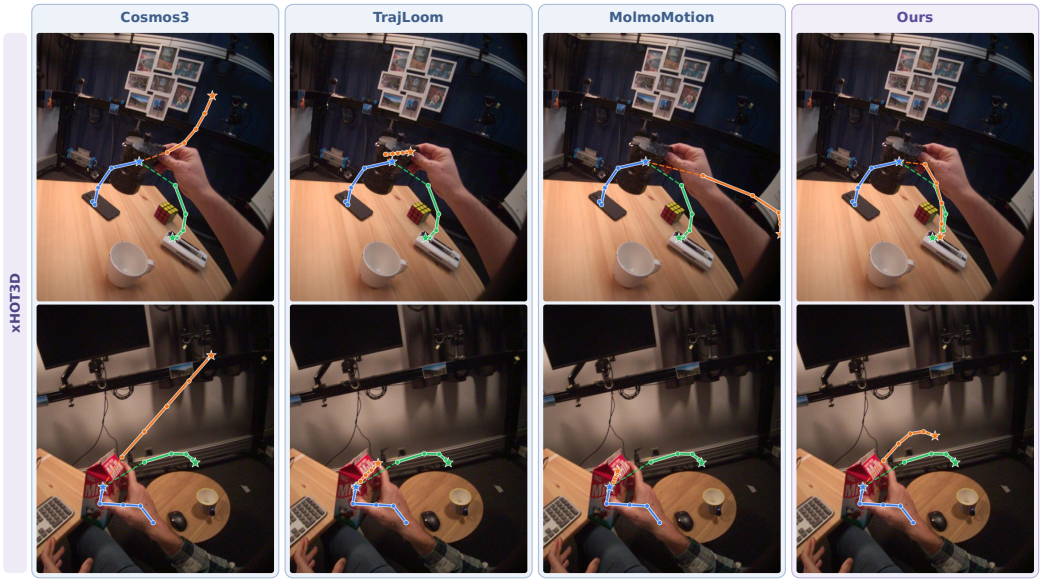

Figure 12: Comparison of Cosmos3, TrajLoom, MolmoMotion, and XTrajectory on two xHOT3D examples. Blue denotes observed history, green the ground-truth future, and orange each method's predicted future.