Learning Long-Context Diffusion Policies via Past-Token Prediction

Marcel Torne* Andy Tang* Yuejiang Liu* Chelsea Finn
Stanford University

***Abstract*—Reasoning over long sequences of observations and actions is essential for many robotic tasks. Yet, learning effective long-context policies from demonstrations remains challenging. As context length increases, training becomes increasingly expensive due to rising memory demands, and policy performance often degrades as a result of spurious correlations. Recent methods typically sidestep these issues by truncating context length, discarding historical information that may be critical for subsequent decisions. In this paper, we propose an alternative approach that explicitly regularizes the retention of past information. We first revisit the copycat problem in imitation learning and identify an opposite challenge in recent diffusion policies: rather than over-relying on prior actions, they often fail to capture essential dependencies between past and future actions. To address this, we introduce Past-Token Prediction (PTP), an auxiliary task in which the policy learns to predict past action tokens alongside future ones. This regularization significantly improves temporal modeling in the policy head, with minimal reliance on visual representations. Building on this observation, we further introduce a multistage training strategy: pre-train the visual encoder with short contexts, and fine-tune the policy head using cached long-context embeddings. This strategy preserves the benefits of PTP while greatly reducing memory and computational overhead. Finally, we extend PTP into a self-verification mechanism at test time, enabling the policy to score and select candidates consistent with past actions during inference. Experiments across four real-world and six simulated tasks demonstrate that our proposed method improves the performance of long-context diffusion policies by 3× and accelerates policy training by more than 10×. Videos are available at https://long-context-dp.github.io.**

I. Introduction

Many robotic tasks are inherently non-Markovian: an appropriate choice of action may depend not only on the current observation but also on past observations and actions [21, 47, 15, 48]. For example, consider manipulation tasks where the robot arm occludes critical parts of the scene, or multi-stage tasks where early steps inform later strategies [24]. Likewise, past actions can prescribe a style of execution, such as speed, curvature, or strategy, that shapes how future actions should unfold [8, 19].

Despite the importance of historical observations, learning long-context robotic policies through imitation learning remains difficult. First, longer observation histories often introduce features that spuriously correlate with actions in the training data. Policies that latch onto such information may diverge from expert behavior during deployment, leading to performance degradation [10, 34]. Second, conditioning on high-dimensional image sequences imposes a rapidly growing memory and computation burden, making end-to-end training excessively expensive at scale [48, 16].

To cope with these challenges, recent methods typically limit the amount of historical information the policy sees – either by truncating the context length [8, 5] or by engineering past observations into compact representations, such as selecting key frames [42] and summarizing observations [48]. While these strategies reduce memory requirement, they risk discarding information critical to subsequent decisions.

In this paper, we introduce a simple and effective approach for learning long-context robot policies, illustrated in Fig. 1. At the core of our method is to explicitly regularize the information preserved from past observations. Specifically, we start with an analysis on the discrepancy between recent diffusion policies and their corresponding demonstrations [8, 15]. We observe that action sequences generated by learned policies often exhibit weaker temporal dependencies than those in expert data. To address this, we present past-token prediction (PTP), an auxiliary task where the policy learns to predict past actions alongside future ones. This regularizer encourages the model to attend more effectively to past context, significantly boosting performance. Crucially, we find that the benefits of PTP primarily emerge in the policy head for sequence modeling, rather than the visual encoder.

Building upon this analysis, we introduce a multi-stage training recipe: first, pre-train the visual encoder in a short-context setting, where the policy learns to predict a chunk of future actions from only a few past frames [47, 8], and subsequently fine-tune a long-context decoder that jointly predicts past and future actions from precomputed image embeddings. This design enables the policy to capture long-range temporal dependencies while substantially reducing memory and computational overhead. Beyond training, we further leverage PTP as a self-verification mechanism during inference. At each time step, the policy generates multiple candidate actions and selects the one most consistent with its previously executed actions.

In summary, our main contributions are twofold: (i) identify a critical discrepancy in temporal action dependencies between learned policies and expert demonstrations (§III), (ii) propose a training and inference method for long-context imitation learning via past-token prediction (§IV). Empirically, we validate our method on diffusion-based policies [8] across six simulation and four real-world tasks (§V). On average, our

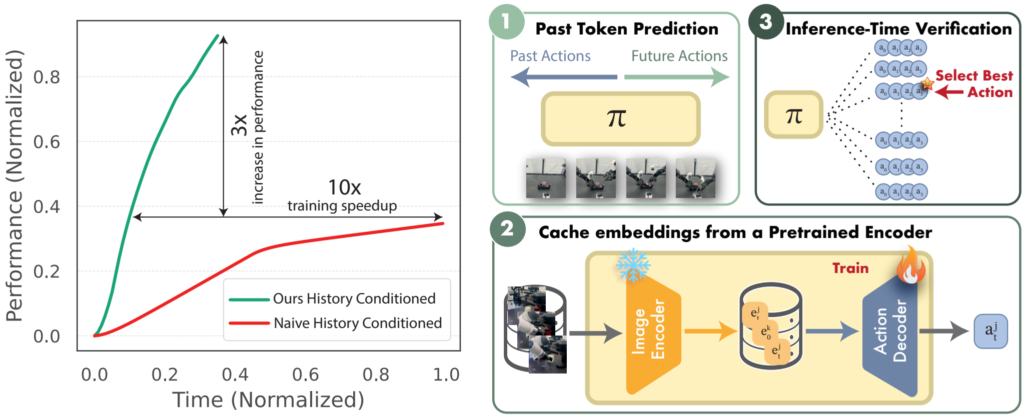

Fig. 1: We propose a simple framework for learning long-context diffusion policies from human demonstrations. Our method leads to 3× gains in performance while reducing the training expense by more than 10×.

method increases the success rate of long-context policies by 3× while reducing training overhead by over 10 times. Notably, it enables policies to achieve 80% success on history-critical tasks where existing methods fail entirely.

II. Related Work

a) Imitation Learning.: Imitation learning has long served as a simple yet powerful paradigm for robot learning [1, 26, 45]. Early approaches typically framed it as a supervised learning problem, where the policy learns to map a given observation to the target action [28]. More recent works have shifted toward modeling the distribution of demonstrations [47, 8, 15, 46, 3, 40, 12, 19]. This approach has recently achieved remarkable success towards generalist robot policies [7, 4]. However, imitation learning remains highly susceptible to covariate shift [10, 41, 32], e.g. Ross et al. [29] and Spencer et al. [34] characterize compounding errors in a feedback loop once the learned policy diverges from the demonstration manifold. This problem is exacerbated by high-dimensional visual inputs, where less robust features might be learned due to underspecification [23]. Notably, recent works [8, 48] have empirically found that image-conditioned specialist and generalist policies degrade with history, leading many works to exclude history altogether [37, 4, 6, 47, 14, 38, 17]. Our work introduces and analyzes a training recipe that counteracts this degradation.

b) Long-Context Policies.: Handling long sequences of high-dimensional observations has been a persistent challenge in robot learning. A common strategy is to reduce the input history—by discarding parts of the past via adversarial regularization [41], information bottlenecks [30], or selecting salient subsets through techniques like keyframes [42] and motion tracks [27]. Other methods construct higher-level summaries, such as sketch synthesis [36] or visual trace prompting [48], especially for generalist policies. These approaches rely on the assumption that much of the historical context is irrelevant—a simplification that may break down in temporally complex tasks. An alternative line of work attempts to model the full context in an autoregressive manner using action tokens [25, 11]. Yet, designing action tokenizers that can effectively capture long-range temporal structure remains an open problem [39]. Our method takes an orthogonal approach: we explicitly regularize diffusion policies to retain information about past actions that would otherwise be lost from historical context.

c) Test-Time Scaling.: Recent research in language modeling, image generation, and robotics has shown that inference-time compute may allow models to improve their performance [2, 20, 22]. Some seek to build an additional verifier to re-rank the output samples [9, 43, 18, 44], while others propose to leverage the internal knowledge to improve reasoning through self-verification [35]. Our method echoes the latter paradigm in the robotic context: our policy is trained to predict accurate past actions before predicting the present action and can self-verify at test-time through past action accuracy. Similarly to how it may be more compute-efficient to use test-time compute on a small LLM [33], we show that checkpoints trained for fewer epochs or at shorter histories can approach the performance of optimal checkpoints by using more test-time compute.

III. Preliminaries

a) Problem Setting.: We consider the problem of imitation learning, where a robot learns to perform complex tasks from expert demonstrations. At each time step t, the robot receives a visual observation o_t and executes an action a_t. Crucially, we assume that each observation o_t contains only partial information about the underlying state s_t, but the complete information about s_t can be inferred from the history of observations. This setting encapsulates practical challenges commonly encountered in robotic tasks, such as latent strategies in the demonstrations (e.g., expert preference), temporal context (e.g., stage within a task), and perceptual limitations (e.g., visual occlusions).

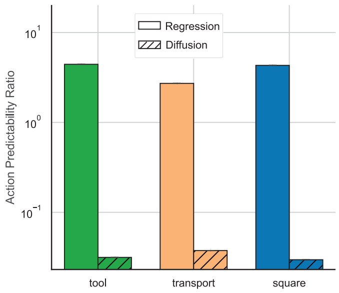

Fig. 2: Comparison of regression-based and diffusion-based policies in temporal action dependency, normalized by that in demonstrations.

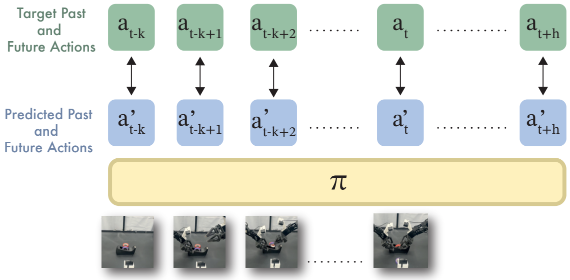

Fig. 3: Illustration of past-token prediction. The policy head is trained to jointly predict both past and future action tokens, encouraging the model to capture the temporal dependencies that are otherwise lost between past and future actions.

Given a dataset of N expert demonstrations $\mathcal{D} = \{\tau_i\}_{i=1}^N$, where each demonstration trajectory τ_i consists of a sequence of observation-action pairs, our goal is to learn a long-context policy $\pi_\theta(\mathbf{a}_{t:t+l}|\mathbf{o}_{t-k:t})$ that takes as input the current observation along with the history $\mathbf{o}_{t-k:t} = (o_{t-k}, \ldots, o_t)$ over the past k time steps, and predicts the current and future actions $\mathbf{a}_{t:t+l} = (a_t, \ldots, a_{t+l})$ spanning the next l time steps. While increasing the context length k provides richer historical information, the resulting long-context policies often suffer from substantial performance declines [8, 48].

b) Practical Challenges.: One central challenge in long-context imitation learning arises from the prevalence of spurious features in observation history. As context length increases, the model is exposed to a growing set of input features, some of which correlate with but do not causally influence the expert actions. Policies relying on these spurious features in observation history may reach high prediction accuracy within the training distribution but generalize poorly during deployment [10]. One notable manifestation is the copycat behavior [41], where the learned policy simply mimics previous actions as predictions for future ones, ignoring current state observations. Does this phenomenon persist in modern imitation learning methods?

To understand this, we evaluate temporal action dependencies by measuring how predictable the current action is from prior actions alone. Specifically, given a set of demonstrations, we first train long-context policies with varying observation history lengths. We then collect policy rollouts and train a simple two-layer MLP $\phi(a_t|a_{t-1})$ to predict the current action based solely on the previous action. We measure the mean-squared error ϵ_π of the MLP predictor on holdout rollouts and similarly obtain ϵ_{π^*} for expert demonstrations. Following [41, 31], we define the action predictability ratio as $\epsilon_{\pi^*}/\epsilon_\pi$. Intuitively, a ratio greater than 1 indicates an over-reliance on previous actions (i.e., copycat behavior), while a ratio less than 1 indicates weaker-than-expert action dependency.

Fig. 2 shows the action predictability ratios for classical regression-based policies and modern diffusion-based policies [8] across three simulation tasks in RoboMimic [21]. Interestingly, the two approaches exhibit opposite failure modes: The regression-based policies indeed exhibit high action predictability, even exceeding that of the expert demonstrations. In contrast, *modern diffusion-based policies yield predictability ratios significantly below 1, indicating a surprising underuse of past action information despite conditioning on long observation histories.* Ideally, an effective imitator should not only learn to accurately predict expert actions in the training set, but also reach a similar level of temporal action dependencies in its rollouts. We will next introduce a method designed to explicitly bridge this gap.

IV. Method

In this section, we introduce a long-context imitation learning method, aiming to improve both policy performance and training efficiency. We will first describe a simple but crucial auxiliary task to enhance temporal dependencies in sequential decision-making (§IV-A). We will then present a multi-stage training recipe that preserves the benefit of this auxiliary task while reducing memory consumption (§IV-B). Finally, we will introduce an inference technique that leverages the auxiliary task to effectively self-verify sampled predictions at test time (§IV-C).

A. Past-Token Prediction

One common design choice in imitation learning is next-token prediction, where the policy predicts only the immediate next action token at each time step. To better capture temporal dependencies, recent methods have extended this to predict a chunk of future action tokens [47, 8]. However, as shown in §III, this design alone remains insufficient for modeling the critical dependencies between past and future decisions.

We address this issue through Past-Token Prediction (PTP), an auxiliary objective that tasks the policy to predict past action tokens alongside future ones. Formally, given a sequence of observations $\mathbf{o}_{t-k:t}$, the policy is trained to jointly predict the action tokens from the past time step $t-k$ to the upcoming time step $t+h$:

$$\hat{\mathbf{a}}_{t-k:t+h} = \pi_\theta(\mathbf{o}_{t-k:t}). \tag{1}$$

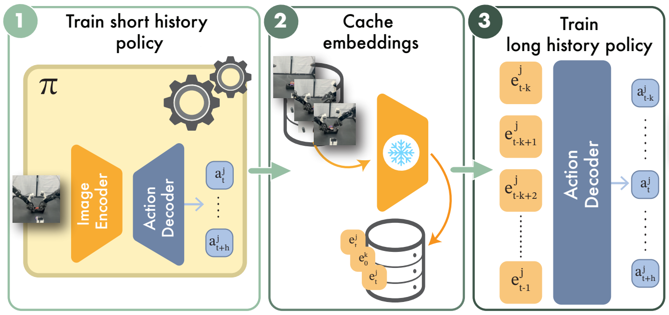

Fig. 4: Overview of multistage training with embedding caching.

As illustrated in Fig. 3, this objective expands the prediction window in both temporal directions, explicitly encouraging the policy to preserve information about past actions from the history context.

B. Memory-Efficient Training with PTP

Recent imitation learning approaches typically train visuomotor policies end-to-end, jointly optimizing both the visual encoder and the policy head. However, this strategy incurs memory costs that grow linearly with context length, making it prohibitively expensive to train long-context policies.

To address this, we propose a multi-stage training recipe that decouples visual representation learning from policy optimization. Our training process consists of three specific stages:

1) **Encoder Training:** We first train the visual encoder with a short observation context but a long prediction horizon, encouraging it to extract representations that retain information critical for predicting l subsequent steps.
2) **Feature Caching:** We then freeze the encoder and precompute embeddings for all frames in the training set. This caching step eliminates redundant computation during policy training.
3) **Policy Training:** Finally, we train the policy head conditioned on long-context observations represented by the cached embeddings. This enables the policy to model long-range dependencies without repeatedly processing visual inputs.

As shown in Fig. 4, this multistage training approach retains a computational footprint similar to short-context training while enabling efficient scaling to longer observation contexts. In Appendix C, we show in more detail how the features of a short-history policy are sufficient to support strong long-context performance.

C. Test-Time Verification with PTP

Another common challenge in recent diffusion policies lies in the robustness of sampled predictions. Often, not all samples are equally good at capturing the critical temporal dependencies. Recent work has explored re-ranking sampled predictions based on consistency with past predictions [19]. However, when the previous prediction for future actions is suboptimal, e.g. because of unexpected environmental changes, this approach may propagate errors rather than correct them.

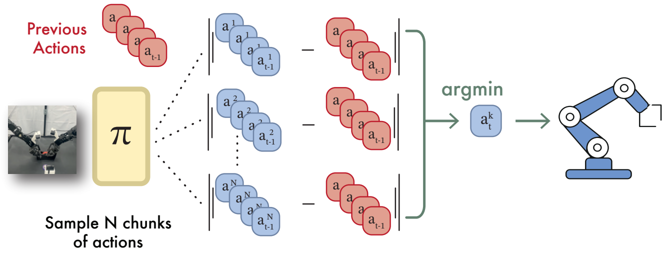

Fig. 5: Test-time verification. Multiple action sequences are sampled from the same observation, and the policy selects the sequence that is most consistent compared to ground-truth previous actions.

To address this shortcoming, we cast Past-Token Prediction as a self-verification mechanism during deployment. At each inference step, we sample a batch of B candidate action sequences:

$$\mathcal{A} = \{\hat{\mathbf{a}}^{(1)}, \ldots, \hat{\mathbf{a}}^{(B)}\}, \quad \hat{\mathbf{a}}^{(i)} \sim \pi_\theta(\mathbf{o}_{t-k:t}), \tag{2}$$

where each sampled candidate $\hat{\mathbf{a}}^{(i)} = (a_{t-k}, \ldots, a_{t+h})^{(i)}$ includes both reconstructed past actions and predicted future actions. Since the first $k-1$ actions have already been executed, we use them as a ground-truth reference and select the candidate whose reconstructed past actions best match the executed ones:

$$\hat{\mathbf{a}}^* = \arg\min_{\hat{\mathbf{a}} \in \mathcal{A}} \sum_{\tau=t-k}^{t-1} \|\hat{a}_\tau - a_\tau\|^2 \tag{3}$$

As illustrated in Fig. 5, this sample selection procedure is fully parallelizable on GPU devices, enabling self-verification of temporal action dependencies with minimal overhead.

V. Experiments

In this section, we evaluate the proposed method for learning long-context diffusion policies. We seek to answer the following questions regarding policy performance and training efficiency:

1) How effectively does PTP mitigate the lack of temporal action dependencies shown in §III?
2) How well do the resulting policies perform on tasks that require history-aware decision-making?
3) To what extent does the proposed multi-stage training recipe accelerate policy learning?
4) Could PTP verification further mitigate deficiencies in temporal dependencies at test time?
5) Finally, how do these findings generalize to history-critical tasks in the real world?

To this end, we evaluate our method on the modern diffusion-based policy [8], in comparison with the classical regression-based policy. By default, both policies receive visual and proprioceptive observations from the past 16 time steps as conditional input. We compare policies trained with *PTP* against two baselines: *no-history* policies that take only

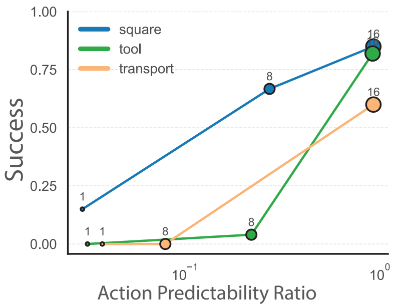

Fig. 6: Effect of PTP on temporal action dependency and performance. Increasing the amount of past-token supervision aligns the learner more closely with expert action dependencies, resulting in higher success rates.

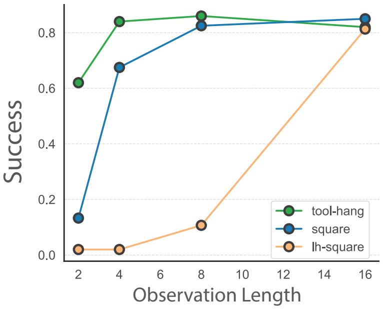

Fig. 7: Effect of history observations on PTP-trained diffusion policies. Increasing the context length progressively enhances policy performance, especially in history-critical tasks such as Long-Horizon Square

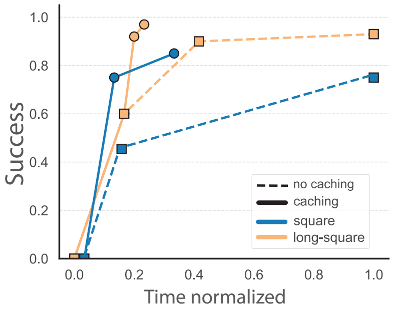

Fig. 8: Effect of feature caching. Caching speeds up training by over 5× without hurting performance. On complex tasks like Tool Hang, long-context policies fail to perform even after two days without caching.

the current and past single frame as input, and *no-PTP* policies that are trained without PTP. Unless otherwise specified, all policies are trained using the multistage recipe with feature caching and evaluated under a single-sample inference setting. The effect of test-time verification is evaluated separately across multiple checkpoints under varying sample budgets. Additional results are presented in Appendix A, with implementation details provided in Appendix E.

A. Simulation Experiments

We first evaluate our method across six simulated tasks. Four of these are sourced from existing benchmarks: *square, tool hang, and transport* from RoboMimic [21], each provided by multi-human demonstration datasets, and Push-T from Chi et al. [8]. These tasks feature diverse strategies in demonstrations, requiring the policy to infer and commit to consistent behaviors over time based on historical context. In addition, we introduce two new long-horizon simulation tasks: *long-horizon square*, where the robot must place and remove a square onto the peg twice before finally dropping it in the peg; and *long-horizon aloha*, where one arm must pick up a block, move it to the center of the field of view, and return it precisely to its original location. Success in these new tasks critically depends on the ability to recall and act upon information observed earlier in the episode. Each policy-task pair is evaluated over 100 episodes across three random seeds. We next summarize the key findings from these simulation experiments.

Takeaway 1: PTP mitigates deficiencies in modeling temporal action dependencies. To validate the effect of PTP on modeling temporal action dependencies, we use the same set of tasks as in §III and train policies to predict a variable number of past tokens $\{\hat{a}_{t-c-1}, \dots, \hat{a}_t\}$, where c denotes the number of actions included in the prediction target. Specifically, we compare three variants: (i) *no-PTP* with $c = 1$, equivalent to the vanilla next-token prediction baseline; (ii) *half-PTP* with $c = 8$, which predicts action tokens corresponding to half the observation window; and (iii) *full-PTP* with $c = 16$. As shown in Fig. 6, PTP consistently increases the action predictability and gets closer to that observed in the expert demonstrations. Notably, the non-PTP baseline exhibits approximately 10× to 100× weaker action predictability ratios compared to expert behavior, whereas full-PTP yields temporal dependencies comparable to demonstrations.

Takeaway 2: PTP significantly improves the performance of modern policies. To assess the impact of PTP on task performance, we compare our method against the no-history and no-PTP baselines on two classes of policies: diffusion-based versus regression-based. All models are evaluated following the protocol from [8], with action chunking set to 8 time steps. As shown in **??**, while the *no-history* baseline already performs competitively on some existing tasks, PTP matches or surpasses its performance. The advantage of PTP is particularly pronounced in long-horizon tasks: both the *no-history* and *no-PTP* baselines struggle with success rates below 30%, whereas our method achieves near-perfect performance on the long-horizon tasks. Averaged across all six simulation tasks, PTP yields an average 50% improvement for diffusion-based policies when conditioned on long contexts, and outperforms the regression counterpart by nearly 20%.

Takeaway 3: PTP-trained policies benefit from longer contexts. To further understand the role of historical contexts, we evaluate PTP-trained diffusion-based policies conditioned on observation histories of varying lengths, ranging from 2 to 16 time steps. As shown in Fig. 7, longer histories generally lead to improved performance. For relatively simple tasks such as *square*, gains tend to saturate beyond 4 steps; however, for more complex tasks, such as *transport*, *long-horizon square*, and *long-horizon aloha*, longer contexts provide substantial performance boosts.

Takeaway 4: Embedding caching accelerates PTP training without sacrificing performance. To assess the effectiveness of the proposed multistage training strategy, we train history-conditioned diffusion policies with and without embedding caching for two days on the three tasks used above (§III), evaluating checkpoints saved every 50 epochs. As shown in Fig. 8, the vanilla training recipe without caching completes a limited number of epochs within the time budget. In contrast,

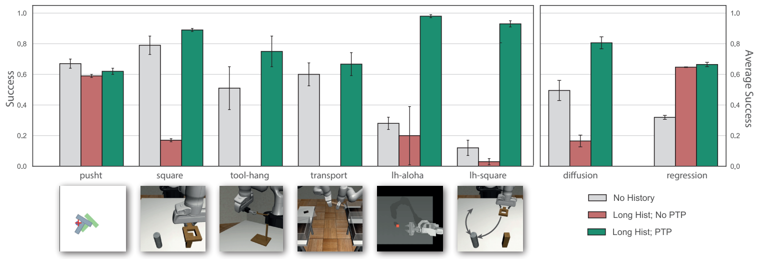

Fig. 9: Comparison of different policies across six simulation tasks. Unlike classical regression-based policies, modern diffusion-based policies exhibit a clear drop in performance when conditioned on historical observations. Our method achieves an average improvement of over 30% compared to no-history diffusion policies, and over 60% compared to no-PTP diffusion policies.

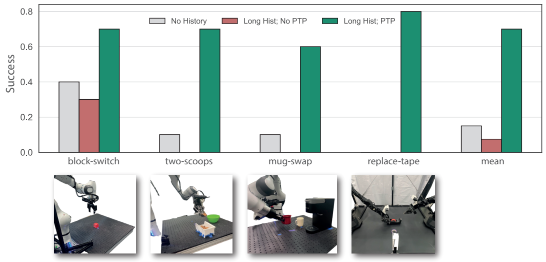

Fig. 10: Comparison of different policies on four real-world tasks that critically depend on historical context. Our method yields over a 55% improvement compared to baselines.

our caching-based approach matches performance in just 20% of the training time and surpasses it within 40% of the compute budget.

Takeaway 5: PTP verification boosts performance in challenging settings at test time. To validate the potential of self-verification through PTP, we evaluate history-conditioned policies on three challenging tasks, including Tool Hang, Transport, and Long Square, trained under constrained compute budgets and tested with varying sampling budgets $\{1, 3, 5, 10\}$. As shown in Fig. 11, PTP-guided sample selection provides notable performance gains. Notably, increasing the number of sampled candidates from 1 to 5 results in approximately 5% improvement in success rate on all these tasks.

B. Real-World Experiments

We next examine our method on four history-critical tasks across two robot platforms in the real world: *Franka block switch*: move a block from one side to another, where history is needed to correctly infer which side to place the block; *Franka two scoops*, transport two scoops to the target, where history is needed to count scoops; *Franka mug replacement* and *ALOHA tape replacement*: replace one mug or tape by another, where history is needed to distinguish old and new objects. Across all tasks, we use diffusion-based policies with a context length of 16 and a chunk size of 8. Due to different ranges of temporal dependency in these tasks, we apply task-specific subsampling rates detailed in Appendix E.

Quantitatively, PTP outperforms baselines by over 4× in the real world. As shown in Fig. 10, the *no-history* baseline is limited to an average success rate of 15% due to the absence of critical history information. The *no-PTP* baseline, which simply conditions on history without PTP, yields near-zero success on three of four tasks. In contrast, our method achieves an average 70% success rate. Notably, on Tape Replacement, one of the most challenging tasks across the board, our method achieves 80% success, while the two baselines fail entirely.

Qualitatively, PTP-trained long-context policies excel at both high-level and low-level memory. As shown in the videos on the website, the two baselines exhibit distinct failure modes: the *no-history* policies often fail at high-level decision-making, such as replacing the wrong object or miscounting scoops, whereas the *no-PTP* baseline struggles with low-level motor control, such as unsuccessful grasps and inaccurate placements. In comparison, policies trained with our method demonstrate improvement in both high-level planning and low-level control, resulting in more coherent and reliable behaviors.

VI. Conclusion

We have presented Past Token Prediction (PTP), a simple yet effective auxiliary objective for learning history-conditioned diffusion policies from demonstrations. We have shown that PTP can effectively strengthen temporal action dependencies that are often lost in recent diffusion policies. In addition, we have introduced a multistage training strategy and a self-verification mechanism that allow for effective use of PTP during both training and inference. Experiments across ten manipulation tasks in both simulations and the real world demonstrate its advantages in efficiency and effectiveness.

VII. Limitations and Discussion.

Our work has focused on extending context length specifically for diffusion policies, motivated by their growing prevalence in the robot learning community. Nevertheless, the effectiveness of our method may generalize to other classes of modern policies as well. In fact, concurrently with our work, Vuong et al. [39] observes similar challenges in tokenization-based policies. Extending our approach to such settings, and more broadly, designing action tokenizers that better preserve temporal structure, can be an exciting avenue for future research. Another practical challenge our method faces is inference overhead. While we have shown that caching and reusing visual embeddings can increase training efficiency, inference remains a practical bottleneck for closed-loop operations. To make inference time manageable, we followed common practices from recent literature by downsampling observation history and extending action chunk. However, these adjustments are known to compromise policy reactivity. Designing strategies to further accelerate inference could be another fruitful direction for future research.

Acknowledgments

We thank Sergey Levine and the members of the Stanford IRIS lab for their insightful discussions. We thank Joey Hejna, Moo Jin Kim, Alec Lessing, and Amber Xie for their support with the real world experiments. This work was supported in part by The Robotics and AI Institute, DARPA, and the ONR grant N00014-22-1-2621. C. Finn is a CIFAR fellow. Y. Liu is an SNSF postdoc fellow.

References

[1] Brenna D. Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57 (5):469–483, May 2009.

[2] Hritik Bansal, Arian Hosseini, Rishabh Agarwal, Vinh Q. Tran, and Mehran Kazemi. Smaller, Weaker, Yet Better: Training LLM Reasoners via Compute-Optimal Sampling. *arXiv preprint arXiv:2408.16737*, August 2024.

[3] Homanga Bharadhwaj, Jay Vakil, Mohit Sharma, Abhinav Gupta, Shubham Tulsiani, and Vikash Kumar. RoboAgent: Generalization and Efficiency in Robot Manipulation via Semantic Augmentations and Action Chunking. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4788–4795, May 2024.

[4] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. \$π_0\$: A Vision-Language-Action Flow Model for General Robot Control. *arXiv preprint arXiv:2410.24164*, November 2024.

[5] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control, 2024. URL https://arxiv.org/abs/2410.24164.

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

[7] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alexander Herzog, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Kuang-Huei Lee, Sergey Levine, Yao Lu, Utsav Malla, Deeksha Manjunath, Igor Mordatch, Ofir Nachum, Carolina Parada, Jodilyn Peralta, Emily Perez, Karl Pertsch, Jornell Quiambao, Kanishka Rao, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Kevin Sayed, Jaspiar Singh, Sumedh Sontakke, Austin Stone, Clayton Tan, Huong Tran, Vincent Vanhoucke, Steve Vega, Quan Vuong, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. RT-1: Robotics Transformer for Real-World Control at Scale. In *Robotics: Science and Systems XIX*. Robotics: Science and Systems Foundation, July 2023. ISBN 978-0-9923747-9-2.

[8] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion Policy: Visuomotor Policy Learning via Action Diffusion. In *Robotics: Science and Systems XIX*. Robotics: Science and Systems Foundation, July 2023. ISBN 978-0-9923747-9-2.

[9] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training Verifiers to Solve Math Word Problems. *arXiv preprint arXiv:2110.14168*, November 2021.

[10] Pim de Haan, Dinesh Jayaraman, and Sergey Levine. Causal Confusion in Imitation Learning. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[11] Letian Fu, Huang Huang, Gaurav Datta, Lawrence Yunliang Chen, William Chung-Ho Panitch, Fangchen Liu, Hui Li, and Ken Goldberg. In-Context Imitation Learning via Next-Token Prediction. *arXiv preprint arXiv:2408.15980*, September 2024.

[12] Siddhant Haldar, Zhuoran Peng, and Lerrel Pinto. BAKU: An Efficient Transformer for Multi-Task Policy Learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, November 2024.

[13] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, Peter David Fagan, Joey Hejna, Masha Itkina, Marion Lepert,

Yecheng Jason Ma, Patrick Tree Miller, Jimmy Wu, Suneel Belkhale, Shivin Dass, Huy Ha, Arhan Jain, Abraham Lee, Youngwoon Lee, Marius Memmel, Sungjae Park, Ilija Radosavovic, Kaiyuan Wang, Albert Zhan, Kevin Black, Cheng Chi, Kyle Beltran Hatch, Shan Lin, Jingpei Lu, Jean Mercat, Abdul Rehman, Pannag R. Sanketi, Archit Sharma, Cody Simpson, Quan Vuong, Homer Rich Walke, Blake Wulfe, Ted Xiao, Jonathan Heewon Yang, Arefeh Yavary, Tony Z. Zhao, Christopher Agia, Rohan Baijal, Mateo Guaman Castro, Daphne Chen, Qiuyu Chen, Trinity Chung, Jaimyn Drake, Ethan Paul Foster, Jensen Gao, David Antonio Herrera, Minho Heo, Kyle Hsu, Jiaheng Hu, Donovon Jackson, Charlotte Le, Yunshuang Li, Kevin Lin, Roy Lin, Zehan Ma, Abhiram Maddukuri, Suvir Mirchandani, Daniel Morton, Tony Nguyen, Abigail O'Neill, Rosario Scalise, Derick Seale, Victor Son, Stephen Tian, Emi Tran, Andrew E. Wang, Yilin Wu, Annie Xie, Jingyun Yang, Patrick Yin, Yunchu Zhang, Osbert Bastani, Glen Berseth, Jeannette Bohg, Ken Goldberg, Abhinav Gupta, Abhishek Gupta, Dinesh Jayaraman, Joseph J. Lim, Jitendra Malik, Roberto Martín-Martín, Subramanian Ramamoorthy, Dorsa Sadigh, Shuran Song, Jiajun Wu, Michael C. Yip, Yuke Zhu, Thomas Kollar, Sergey Levine, and Chelsea Finn. DROID: A Large-Scale In-The-Wild Robot Manipulation Dataset. *arXiv preprint arXiv:2403.12945*, March 2024.

[14] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P. Foster, Pannag R. 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. In *8th Annual Conference on Robot Learning*, September 2024.

[15] Seungjae Lee, Yibin Wang, Haritheja Etukuru, H. Jin Kim, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. Behavior Generation with Latent Actions. *arXiv preprint arXiv:2403.03181*, March 2024.

[16] Xinghang Li, Peiyan Li, Minghuan Liu, Dong Wang, Jirong Liu, Bingyi Kang, Xiao Ma, Tao Kong, Hanbo Zhang, and Huaping Liu. Towards Generalist Robot Policies: What Matters in Building Vision-Language-Action Models. *arXiv preprint arXiv:2412.14058*, December 2024.

[17] Yi Li, Yuquan Deng, Jesse Zhang, Joel Jang, Marius Memmel, Caelan Reed Garrett, Fabio Ramos, Dieter Fox, Anqi Li, Abhishek Gupta, and Ankit Goyal. HAMSTER: Hierarchical Action Models for Open-World Robot Manipulation. In *The Thirteenth International Conference on Learning Representations*, October 2024.

[18] Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's Verify Step by Step. In *The Twelfth International Conference on Learning Representations*, October 2023.

[19] Yuejiang Liu, Jubayer Ibn Hamid, Annie Xie, Yoonho Lee, Maximilian Du, and Chelsea Finn. Bidirectional Decoding: Improving Action Chunking via Closed-Loop Resampling. *arXiv preprint arXiv:2408.17355*, December 2024.

[20] Nanye Ma, Shangyuan Tong, Haolin Jia, Hexiang Hu, Yu-Chuan Su, Mingda Zhang, Xuan Yang, Yandong Li, Tommi Jaakkola, Xuhui Jia, and Saining Xie. Inference-Time Scaling for Diffusion Models beyond Scaling Denoising Steps. *arXiv preprint arXiv:2501.09732*, January 2025.

[21] Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What Matters in Learning from Offline Human Demonstrations for Robot Manipulation. In *Proceedings of the 5th Conference on Robot Learning*, pages 1678–1690. PMLR, January 2022.

[22] Mitsuhiko Nakamoto, Oier Mees, Aviral Kumar, and Sergey Levine. Steering Your Generalists: Improving Robotic Foundation Models via Value Guidance. In *8th Annual Conference on Robot Learning*, September 2024.

[23] Soroush Nasiriany, Sean Kirmani, Tianli Ding, Laura Smith, Yuke Zhu, Danny Driess, Dorsa Sadigh, and Ted Xiao. Rt-affordance: Affordances are versatile intermediate representations for robot manipulation, 2024. URL https://arxiv.org/abs/2411.02704.

[24] Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. RoboCasa: Large-Scale Simulation of Everyday Tasks for Generalist Robots. *arXiv preprint arXiv:2406.02523*, June 2024.

[25] Ilija Radosavovic, Baifeng Shi, Letian Fu, Ken Goldberg, Trevor Darrell, and Jitendra Malik. Robot Learning with Sensorimotor Pre-training. In *Proceedings of The 7th Conference on Robot Learning*, pages 683–693. PMLR, December 2023.

[26] Harish Ravichandar, Athanasios S. Polydoros, Sonia Chernova, and Aude Billard. Recent Advances in Robot Learning from Demonstration. *Annual Review of Control, Robotics, and Autonomous Systems*, 3(Volume 3, 2020): 297–330, May 2020.

[27] Juntao Ren, Priya Sundaresan, Dorsa Sadigh, Sanjiban Choudhury, and Jeannette Bohg. Motion tracks: A unified representation for human-robot transfer in few-shot imitation learning, 2025. URL https://arxiv.org/abs/2501.06994.

[28] Stephane Ross and Drew Bagnell. Efficient Reductions for Imitation Learning. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 661–668. JMLR Workshop and Conference Proceedings, March 2010.

[29] Stephane Ross, Geoffrey J. Gordon, and J. Andrew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning, 2011. URL https://arxiv.org/abs/1011.0686.

[30] Seokin Seo, HyeongJoo Hwang, Hongseok Yang, and Kee-Eung Kim. Regularized behavior cloning for blocking the leakage of past action information. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 2128–2153. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/06b71ad997f7e3e4b2e2f2ea12e5a759-Paper-Conference.pdf.

[31] Seokin Seo, HyeongJoo Hwang, Hongseok Yang, and Kee-Eung Kim. Regularized Behavior Cloning for Blocking the Leakage of Past Action Information. *Advances in Neural Information Processing Systems*, 36: 2128–2153, December 2023.

[32] Daqian Shao, Thomas Kleine Buening, and Marta Kwiatkowska. A Unifying Framework for Causal Imitation Learning with Hidden Confounders. *arXiv preprint arXiv:2502.07656*, February 2025.

[33] Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters, 2024. URL https://arxiv.org/abs/2408.03314.

[34] Jonathan Spencer, Sanjiban Choudhury, Arun Venkatraman, Brian Ziebart, and J. Andrew Bagnell. Feedback in imitation learning: The three regimes of covariate shift, 2021. URL https://arxiv.org/abs/2102.02872.

[35] Kaya Stechly, Karthik Valmeekam, and Subbarao Kambhampati. On the Self-Verification Limitations of Large Language Models on Reasoning and Planning Tasks. *arXiv preprint arXiv:2402.08115*, August 2024.

[36] Priya Sundaresan, Quan Vuong, Jiayuan Gu, Peng Xu, Ted Xiao, Sean Kirmani, Tianhe Yu, Michael Stark, Ajinkya Jain, Karol Hausman, Dorsa Sadigh, Jeannette Bohg, and Stefan Schaal. Rt-sketch: Goal-conditioned imitation learning from hand-drawn sketches, 2024. URL https://arxiv.org/abs/2403.02709.

[37] Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, You Liang Tan, Lawrence Yunliang Chen, Pannag Sanketi, Quan Vuong, Ted Xiao, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An Open-Source Generalist Robot Policy. *arXiv preprint arXiv:2405.12213*, May 2024.

[38] Marcel Torne, Arhan Jain, Jiayi Yuan, Vidaaranya Macha, Lars Ankile, Anthony Simeonov, Pulkit Agrawal, and Abhishek Gupta. Robot Learning with Super-Linear Scaling. *arXiv preprint arXiv:2412.01770*, December 2024.

[39] An Dinh Vuong, Minh Nhat Vu, Dong An, and Ian Reid. Action Tokenizer Matters in In-Context Imitation Learning. *arXiv preprint arXiv:2503.01206*, March 2025.

[40] Dian Wang, Stephen Hart, David Surovik, Tarik Kelestemur, Haojie Huang, Haibo Zhao, Mark Yeatman, Jiuguang Wang, Robin Walters, and Robert Platt. Equivariant Diffusion Policy. In *8th Annual Conference on Robot Learning*, September 2024.

[41] Chuan Wen, Jierui Lin, Trevor Darrell, Dinesh Jayaraman, and Yang Gao. Fighting Copycat Agents in Behavioral Cloning from Observation Histories. In *Advances in Neural Information Processing Systems*, volume 33, pages 2564–2575. Curran Associates, Inc., 2020.

[42] Chuan Wen, Jierui Lin, Jianing Qian, Yang Gao, and Dinesh Jayaraman. Keyframe-focused visual imitation learning, 2021. URL https://arxiv.org/abs/2106.06452.

[43] Yixuan Weng, Minjun Zhu, Fei Xia, Bin Li, Shizhu He, Shengping Liu, Bin Sun, Kang Liu, and Jun Zhao. Large Language Models are Better Reasoners with Self-Verification. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, December 2023.

[44] Fei Yu, Anningzhe Gao, and Benyou Wang. OVM, Outcome-supervised Value Models for Planning in Mathematical Reasoning. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 858–875, Mexico City, Mexico, June 2024. Association for Computational Linguistics.

[45] Maryam Zare, Parham M. Kebria, Abbas Khosravi, and Saeid Nahavandi. A Survey of Imitation Learning: Algorithms, Recent Developments, and Challenges. *IEEE Transactions on Cybernetics*, 54(12):7173–7186, December 2024.

[46] Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3D Diffusion Policy. *arXiv preprint arXiv:2403.03954*, March 2024.

[47] Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware. *arXiv preprint arXiv:2304.13705*, April 2023.

[48] Ruijie Zheng, Yongyuan Liang, Shuaiyi Huang, Jianfeng Gao, Hal Daumé III, Andrey Kolobov, Furong Huang, and Jianwei Yang. Tracevla: Visual trace prompting enhances spatial-temporal awareness for generalist robotic policies, 2024. URL https://arxiv.org/abs/2412.10345.

Appendix

A. Additional Experiments

In addition to the main results presented in §V-A, we conduct three experiments to further validate the design decisions behind our proposed method.

B. Test-time Scaling Further Results

We provide the figure for the test-time scaling results.

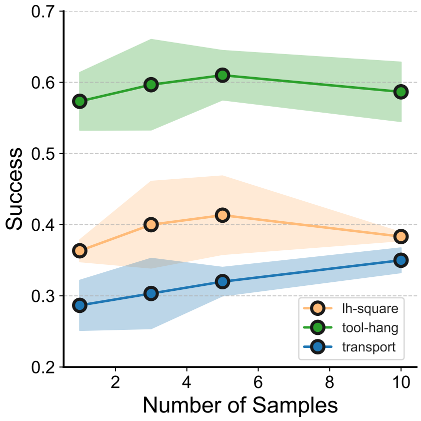

Fig. 11: Effect of PTP self-verification. Increasing sampling budgets yields a 5% gain in challenging closed-loop settings.

C. Which component of the policy benefits most from PTP?

To identify which part of the policy is most influenced by PTP, we compare a fully PTP-trained long-context policy against two ablated variants: *Encoder PTP*, where we first train the visual encoder with PTP, then freeze it and train the action decoder without PTP; *Decoder PTP*, where we conversely train the encoder without PTP, freeze it, and then apply PTP only during decoder training. As shown in Fig. 12, *Decoder PTP* achieves performance on par with the fully trained PTP policy, whereas *Encoder PTP* performs significantly worse. This result suggests that the benefits of PTP primarily stem from improved temporal modeling in the action decoder, rather than from changes to the visual encoder, directly motivating our multi-stage training recipe that decouples encoder pretraining from long-context policy learning.

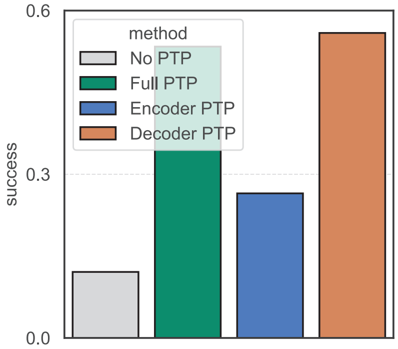

Fig. 12: Performance of ablated PTP variants on Push-T. Applying PTP only to the decoder recovers the full PTP policy performance, whereas encoder-only PTP does not.

D. Does our method reduce reliance on action chunking?

Existing short-context policies typically rely on action chunking compensate for limited access to past observations. However, this common design choice comes at the cost of reduced reactivity. To assess whether our method alleviates this limitation, we compare the performance of three policy variants: (i) *short-context short-chunk*, which receives the past 2 frames as input and outputs single-step actions (chunk size 1); (ii) *long-context short-chunk*, which receives the past 16 frames and also outputs single-step actions; and (iii) *long-context long-chunk*, which receives the past 16 frames and outputs action chunks of size 8. As shown in Fig. 13, the *long-context short-chunk* policies trained by our method substantially outperform the *short-context* counterparts and recover most of the performance of the *long-context long-chunk* policies. This result demonstrates the effectiveness of our method in reducing reliance on open-loop action chunking.

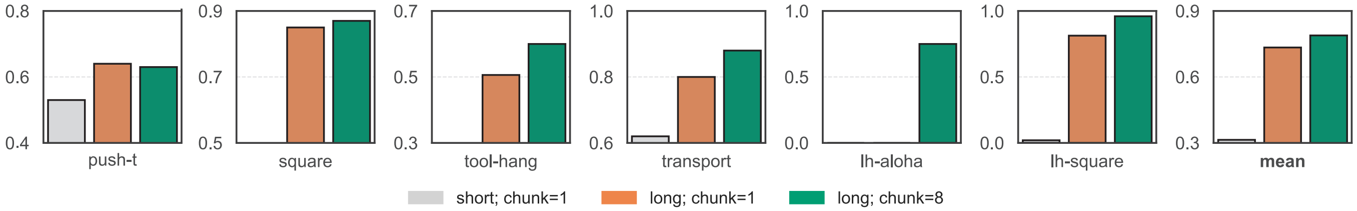

Fig. 13: Comparison of policies with different context lengths and chunk sizes. Long-context policies trained with PTP perform significantly better than short-context policies when run in a fully closed-loop setting (chunk size 1). Moreover, they achieve performance comparable to long-context policies that use open-loop chunking (chunk size 8), indicating reduced reliance on chunking during execution.

E. Is PTP still critical when conditioning on past actions?

Our earlier analysis in Fig. 6 has shown the importance of PTP in capturing temporal action dependencies when the policy is conditioned on past observations. A natural question is whether PTP remains necessary when the model also has direct access to past actions. To understand this, we augment the input to diffusion policies with the previous 16 actions and compare performance with and without PTP. As shown in Fig. 14, even with access to past actions, the vanilla baseline performs poorly without PTP, while our method consistently yields substantially better results. Consistent with our previous findings, this result highlights the critical role of PTP in enabling diffusion policies to effectively model temporal structure, even when past actions are explicitly provided.

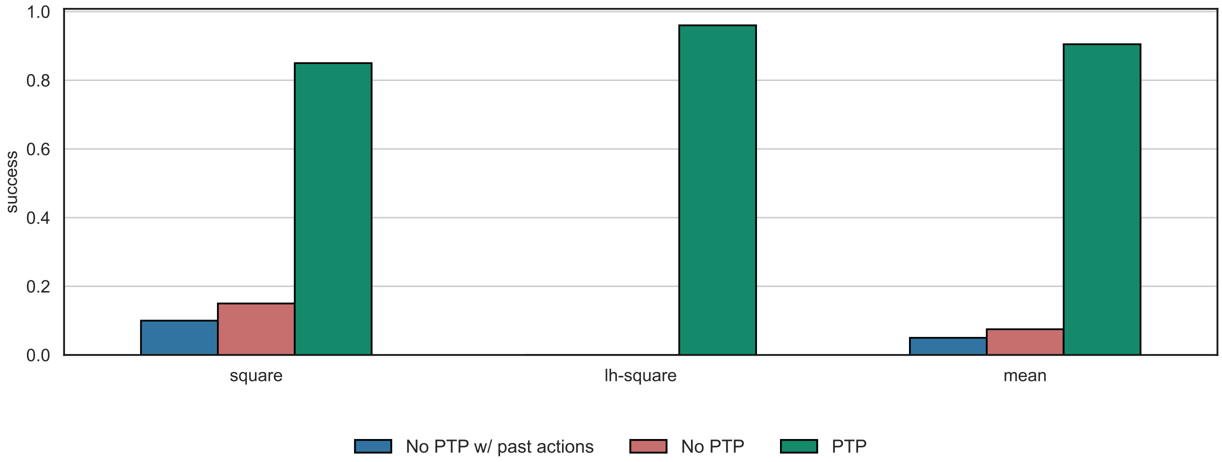

Fig. 14: Comparison of adding past actions into the context history without PTP and our baseline of PTP and no PTP without actions. We observe that adding past actions in the observation doesn't improve performance.

F. Real-World Tasks

We conduct real-world experiments using two robot platforms: a Franka Panda arm set up, and the ALOHA bimanual system. For the Franka setup, we follow the DROID hardware configuration [13], using a single arm with wrist-mounted RGB cameras and proprioceptive sensing. The observation space includes RGB images and end-effector pose, while the action space consists of 6-DoF Cartesian displacements and gripper commands. We collect 50-200 human demonstrations for each task. For the ALOHA platform, we use the bimanual robot as described in [47], with RGB camera inputs and proprioceptive feedback. The observation space consists of dual-arm RGB views and proprioception, and the action space includes joint displacements for both arms and gripper states. We collect 150 demonstrations for the tape replacement task.

TABLE I: Hyper-parameters in simulation and real-world experiments.

Hyperparameter	LH Square	LH Aloha	Block Move	Two Scoops	Mug Repl.	Tape Repl.
Epochs	500	1500	500	500	500	1500
# Demos	100	50	50	200	200	150
# Subsampled frames	20	20	1	20	1	24
# Observations	20	32	10	20	32	20

G. Simulation Tasks

Our simulation tasks include the Push-T task [8], three existing tasks in the Robomimic benchmark [21], along with two new long-horizon tasks that we introduce:

- *Long-Horizon Square*: A variant of the RoboMimic square task, where the robot must place a block onto the farthest peg from its initial location. We collect 100 noisy scripted demonstrations to prevent the policy from inferring the goal using current pose information alone, thus requiring memory of the initial state.
- *Long-Horizon ALOHA*: A simulated bimanual task where one arm picks up a block, moves it to the center of the workspace, and places it back at its original location. Success requires remembering the block's starting position, highlighting the need for long-term memory.

H. Implementation Details

1) Policy Architecture

We build upon the transformer-based Diffusion Policy codebase [8], which supports training and evaluation across multiple Robomimic tasks. All policies are trained for 500 epochs by default, using visual encoders and chunked action prediction. For long-horizon ALOHA tasks, we train for 1500 epochs to accommodate the added difficulty of bimanual coordination and higher-frequency control. To reduce training overhead, all long-context policies are initialized with a frozen short-context encoder, pretrained on 2-frame inputs. This design choice is supported by the analysis in Fig. 12, which shows that freezing the encoder does not impair performance. To further improve training efficiency, we cache visual embeddings during data preprocessing and load them at runtime. This avoids repeatedly passing observations through the encoder and speeds up training.

2) Subsampling Rate

Real-world tasks often require longer history horizons, but full-length observation sequences can be computationally expensive. To reduce inference latency, we apply temporal subsampling to the input sequence. Specifically, instead of feeding all T observations $t_0, t_1, ..., t_{T-1}$, we sample every Kth frame, i.e., $t_{K-1}, t_{2K-1}, ..., t_{T-1}$, where T is a multiple of K. This reduces the effective observation size while retaining broad temporal coverage. Subsampling values are listed in Table I.

3) Context Length

When increasing the observation history, we scale the prediction horizon (past and future tokens) proportionally. We empirically find that the prediction length of future tokens has only a minor effect on task performance. Detailed context length configurations are provided in Table II.

TABLE II: Settings for different context lengths

Observation Length	2	4	8	16
Horizon	16	20	24	32
Future Tokens	14	16	16	16

4) Action Dependency Metric

To quantify how well a policy captures temporal action structure, we use *action predictability* as a proxy metric [41]. Specifically, we measure how accurately the current action a_t can be predicted from a window of $K = 15$ past actions, defined as $p(a_t \mid a_{t-K:t-1})$. We compute this quantity over policy rollouts and compare it to the same metric evaluated on the expert demonstrations. Higher predictability indicates stronger temporal action dependencies captured by the learned policy.

In addition to the results reported in §V-A, we report per-task success rates across varying temporal context lengths, training conditions, and chunking configurations. Table III compares diffusion-based and regression-based baselines on six benchmark tasks, evaluated under different history lengths and with or without PTP. Table IV presents results in the closed-loop setting (chunk size = 1) across different context lengths.

TABLE III: Success rate (%) of diffusion-based and regression-based policies on simulation tasks under different training and history conditions. Results are reported as mean ± standard deviation across 3 seeds.

Method	Push-T	Square	Tool-Hang	Transport	ALOHA	Long Square
Diffusion (PTP)	0.62 ± 0.02	**0.89 ± 0.01**	**0.75 ± 0.10**	**0.67 ± 0.08**	0.98 ± 0.01	**0.93 ± 0.02**
Diffusion (no-PTP)	0.59 ± 0.01	0.17 ± 0.01	0.00 ± 0.00	0.00 ± 0.00	0.20 ± 0.19	0.03 ± 0.02
Diffusion (no-hist)	**0.67 ± 0.03**	0.79 ± 0.06	0.51 ± 0.14	0.60 ± 0.08	0.28 ± 0.04	0.12 ± 0.05
Regression (PTP)	0.40 ± 0.10	0.74 ± 0.02	0.41 ± 0.01	0.63 ± 0.04	0.99 ± 0.02	0.90 ± 0.05
Regression (no-PTP)	0.31 ± 0.31	0.78 ± 0.03	0.40 ± 0.05	0.51 ± 0.01	**1.00 ± 0.00**	0.89 ± 0.01
Regression (no-hist)	0.64 ± 0.08	0.19 ± 0.01	0.14 ± 0.04	0.48 ± 0.03	0.43 ± 0.03	0.00 ± 0.00

TABLE IV: Success rate (%) on simulation tasks under closed-loop execution (chunk size = 1) .

Observations	Push-T	Square	Tool Hang	Transport	Long Square	Mean
2	0.53	0.13	0.62	0.053	0.02	0.37
4	0.53	0.68	0.84	0.13	0.02	0.51
8	0.59	0.83	**0.86**	0.48	0.11	0.63
16	**0.64**	**0.85**	0.82	**0.51**	**0.81**	**0.77**