PAST2NEXT: PAST ACTION CONDITIONING POLICY WITH DATA-AUGMENTED TOKENIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Discrete-token autoregressive policies have emerged as a compelling paradigm for robot learning. Their downstream performance depends on two axes: the action tokenizer's reconstruction fidelity and the policy's reliability in predicting action tokens. Prior work mostly targets the former, yet reconstruction gains often fail to reach downstream success, a compression gap we trace to the latter, under-addressed axis. We introduce **Past2Next**, a discrete-token autoregressive policy that addresses this bottleneck by using past action conditioning for token prediction, then decoding the action tokens into a continuous action chunk via a geometry-aware tokenizer. During tokenizer training, we apply data augmentation to improve codebook coverage on the $SO(3)$ action manifold. During policy learning, Past2Next conditions token prediction on observations, a short history of executed actions, and their first- and second-order finite differences to acquire dynamic features. Our method exploits locally smooth action trajectories governed by the equations of motion. Specifically, finite-difference signals capture the local tangent and higher-order motion trends of the action stream, narrowing the feasible set of next-step actions to a compact neighborhood around its extrapolated trajectory and reducing next-token prediction uncertainty. Across simulation and real-robot experiments, Past2Next reduces predictive Shannon entropy and improves task success rates over the evaluated baselines. Moreover, by such a mechanism, increased codebook capacity improves downstream performance rather than degrading autoregressive learning, allowing finer action quantization to translate into better control performance. Modeling local dynamics and respecting action geometry thus makes an autoregressive policy both accurate and predictable.

1 INTRODUCTION

Recent advances in imitation learning have expanded the capabilities of robot learning policies in manipulation tasks Ai et al. (2025). Among the dominant paradigms, diffusion-based policies Chi et al. (2025); Jiang et al. (2025); Wang et al. (2024) and autoregressive-based discrete-token sequence models Zhang et al. (2025); Cui et al. (2022); Kim et al. (2024); Zhao et al. (2023) have shown particular promise. Diffusion policies formulate action generation as a conditional denoising process, allowing the policy to capture multimodal action distributions arising from diverse demonstrations Song et al. (2025). AR-based policies decompose action prediction into a sequence of next-token predictions, enabling scalable discrete action modeling Kim et al. (2024); Bu et al. (2025), a reusable action tokenizer Tan et al. (2025); Pertsch et al. (2025); Liu et al. (2026a), and inference efficiency Su et al. (2025); Gong et al. (2025).

Despite their different ways of generating actions, existing visuomotor policies model action chunks primarily using data-driven fitting, while physical dynamics and motion continuity are only implicitly imposed through expert demonstrations or architectural choices. Recent works Hu et al. (2026b); Jia et al. (2026) have begun to incorporate history information into policy learning and have demonstrated its empirical util-

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.

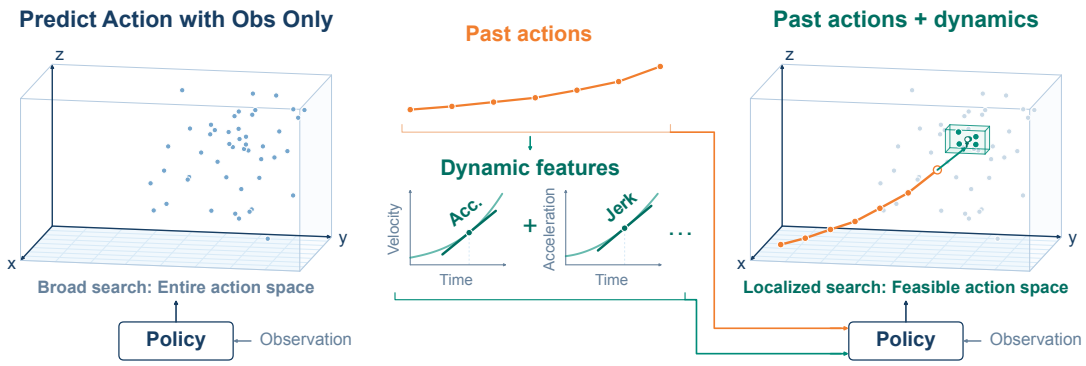

Figure 1: Past2Next conditions on past actions and their dynamic features to localize next-action search. The schematic feasible neighborhood is motivated by the dynamics bound in Sec. 4.

ity, but they treat history only as a generic conditioning signal or initialization heuristic; its role remains poorly understood from either a dynamics or an information-theoretic standpoint.

In this work, we make this connection explicit. Under bounded actuator inputs, finite control bandwidth, and standard regularity conditions on the closed-loop dynamics, robot motion is locally continuous: it cannot change arbitrarily between consecutive control steps Lynch & Park (2017). The executed end-effector delta action is therefore a locally smooth signal, and the next action should follow this signal rather than be drawn independently from the full action space. We expose this structure to the policy by computing first- and second-order finite differences of recent executed actions together with the raw action history, as conditioning features. These features encode local acceleration- and jerk-level motion trends, allowing the policy to contract the feasible next action region from the full action space to a compact neighborhood around the extrapolated trajectory (Fig. 1).

We instantiate this principle on autoregressive (AR) policies built on discrete action tokens, where its effect is both most pronounced and directly measurable. Such policies face two distinct bottlenecks. The first is tokenizer distortion, e.g., the mean-squared error (MSE) between decoded and expert action chunks. The second governs downstream learning and is easily overlooked: how predictable each token is under the available context. Let T map a continuous action chunk $a_{t:t+H-1}$ to a token sequence $z_{1:L}$, with decoder D; the difficulty of AR prediction is then captured by the conditional entropy $\mathcal{H}(z_i \mid c_t, z_{<i})$, where c_t is the observation and execution context. These axes are largely independent: a tokenizer can reconstruct faithfully yet still yield a token stream that is hard to predict. Prior tokenizer-centric work targets the first axis while leaving the second underexamined.

Our two interventions act on these axes separately. Action history attacks the predictability axis. Because physically feasible motions occupy only a small, dynamics-constrained subset of the action space, the feasible-set contraction established above enriches c_t with the robot's recent dynamics, contact state, and temporal continuity, shrinking the set of plausible next actions and lowering $\mathcal{H}(z_i \mid c_t, z_{<i})$. $SO(3)$-aware augmentation attacks the distortion axis: exposing the tokenizer to geometry-preserving transformations of action trajectories broadens its coverage of the action manifold, improves codebook utilization, and yields more faithful discretization. **The two are orthogonal: past action conditioning improves *modelability* by making token prediction easier, while augmentation improves *distortion* by aligning the codebook with the geometry of the continuous action manifold.** Although we develop the framework for AR discrete-token policies, the contraction principle is not specific to them; we provide proof-of-concept experiments showing that the same conditioning benefits continuous flow and diffusion policies. Our contributions are:

1. $SO(3)$**-manifold-aware tokenization.** Because the rotation slice of the action lives on $SO(3)$ and the smoothness argument is intrinsically a tangent-space statement, we apply a data augmentation when learning the tokenizer, broadening codebook coverage of the action manifold and reducing the reconstruction-distortion term.
2. **Dynamics-conditioned action history.** We show, from rigid-body dynamics, that local motion structure localizes the next action to a compact set on the action manifold, and we inject this structure as an enriched conditioning sequence (raw past actions together with their tangent-space acceleration and jerk features), lowering the conditional entropy of next-token prediction.

2 Related Works

Policy Learning. Contemporary visuomotor policies fall into two paradigms that differ in how they represent and produce actions Urain et al. (2024). Continuous policies model the action chunk directly in its native Euclidean space through iterative refinement: diffusion and flow-matching policies sample an action sequence by integrating a learned denoising process Chi et al. (2025); Yan et al. (2025); Liu et al. (2022). Autoregressive (AR) policies utilize a tokenizer to map each continuous action chunk to a short sequence of discrete codes, and a causal transformer predicts those codes one step at a time Kim et al. (2024); Pertsch et al. (2025); Mete et al. (2024). Casting action generation as next-token prediction inherits the scalability and unified sequence-modeling machinery. The dominant assumption in this second paradigm is that a better tokenizer is one that reconstructs actions more faithfully, and prior work has accordingly pursued higher compression rate, predictability, and lower reconstruction error Liu et al. (2026a); Dong et al. (2026).

Historical Action Information in Robot Policy Learning. Recent works have begun to incorporate history information into robot policy learning. Specifically, AR-VLA uses past robot states as autoregressive context Hu et al. (2026b), while Action-to-Action Flow Matching leverages historical actions as a distribution prior for action generation Jia et al. (2026). Legato Liu et al. (2026b) initializes denoising from a schedule-shaped mixture of unexecuted actions in a chunk and noise. PTP conditions action generation on observation histories Torne et al. (2025). Memory-VLA adds explicit memory mechanisms to the VLA model Shi et al. (2025). All these works establish history information as a strong prior for action generation; however, they treat history actions primarily as an auxiliary conditioning signal or an initialization heuristic. The *dynamic constraints* that past actions impose on the feasible set of future actions, and the structure those constraints lend the prediction target, remain underexplored. This is precisely the gap we address: we show that past actions contract the next potential action chunk to a narrow, low-dimensional region, and that this contraction lowers the per-step conditional entropy of the discrete token stream – the quantity that bottlenecks AR-discrete policies.

Geometry-aware Structure in Robotic Manipulation. Geometry-aware structure has long been central to robotic manipulation, from classical rule-based formulations Mahony et al. (2020) to recent learning-based policies Seo et al. (2025). By building the task's geometry into the policy class, those methods improve spatial generalization and sample efficiency, showing strong performance with fewer demonstrations Wang et al. (2022b); Gao et al. (2024); Wang et al. (2022a). Such structure has also been exploited in open-loop manipulation Huang et al. (2022; 2025; 2024); Zhu et al. (2025a) and in grasping and contact-rich settings Hu et al. (2024; 2025); Seo et al. (2023); Simeonov et al. (2022), where geometric structure serves as a strong inductive bias for reasoning about object and interaction poses. More recently, equivariant policies have shown improved sample efficiency and induced more geometrically structured action representations Wang et al. (2024); Hu et al. (2026a); Zhu et al. (2025b); Klee et al. (2026); Huang et al. (2026). We extend geometry-aware policy learning from continuous action prediction to the action-tokenization stage: rather than treating $SO(3)$ structure as a regularizer on policy outputs, we use it — through manifold-aware data

augmentation — as a principle for constructing a compact, geometrically meaningful action token space, broadening codebook coverage over the action manifold for AR-based policies.

3 Controlled Study of Past-Action Conditioning

To better analyze why past actions help, we compare four conditioning variants (Fig. 2) that share the OAT tokenizer Liu et al. (2026a) ($|V| = 1000$) and the policy architecture, and differ only in their condition tokens. (a) The OAT baseline uses only the observations o_{t-1}, o_t. (b) *Raw past actions* adds the seven most recent executed actions $a_{t-1}, \ldots, a_{t-7}$. (c) *Dynamics features* instead adds $d_t^{(1)}$ and $d_t^{(2)}$, the first and second finite differences of the recent actions. (d) *Dynamics + raw past actions* adds both.

Conditioning variant	Success rate (%) ↑
(a) OAT (baseline)	56.3
(b) Raw past actions only	62.2
(c) Dynamics features only	63.8
(d) Dynamics + raw past actions (final)	**68.8**

Table 1: Mean success rate on the ten LIBERO-Long tasks. All variants share the OAT tokenizer ($|V|$=1000) and policy; only the conditioning tokens differ.

Results. Past actions carry local dynamics information, and explicitly exposing this information improves policy performance. In Table 1, raw past-action conditioning raises the success rate from 56.3% to 62.2%. Using only the two finite-difference features achieves 63.8%, demonstrating the value of the local motion trends encoded in action history. Combining both reaches 68.8%, surpassing both raw history and the difference features alone. Raw history and dynamics features play complementary roles: the former preserves absolute action levels and temporal context, while the latter supplies the local change terms needed to extrapolate the next action. Together, they make the predictive structure of recent motion directly available to the policy (Sec. 4). Further analysis of the order selection is in App. A.

Attention. Fig. 3 shows the attention over the conditioning tokens. With raw history only, attention to the history concentrates on the latest action a_{t-1}, mainly in the early layers. When the difference tokens are present, it moves to $d_t^{(1)}$ and $d_t^{(2)}$ in every layer, most strongly to $d_t^{(2)}$. Attention is not a causal attribution, but together with Table 1 it suggests that the policy relies mainly on the local trend of recent actions and uses it more readily when the trend is given explicitly. This raises two questions: why a few recent transitions carry most of the useful history, and why making them explicit helps further. Sec. 4 answers both: the latest action and its first two differences are exactly the terms of a local extrapolator of the next action, whose error is bounded.

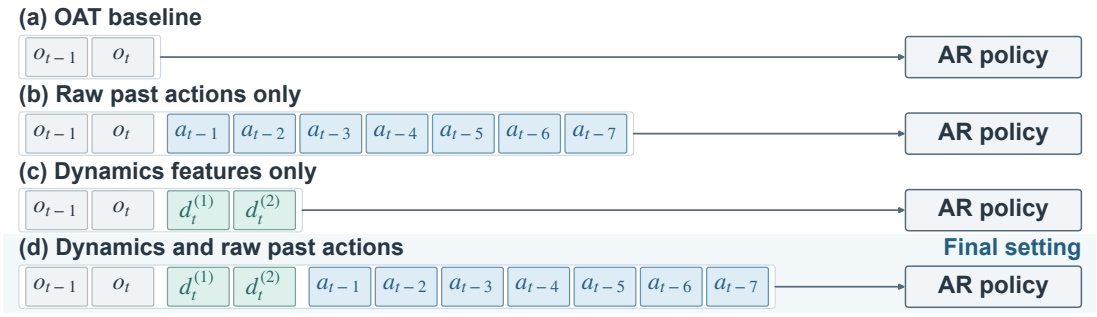

Figure 2: Conditioning variants in the controlled study. Each variant feeds the listed tokens to the same AR policy: observations (gray), dynamics features (teal), and raw past actions (blue).

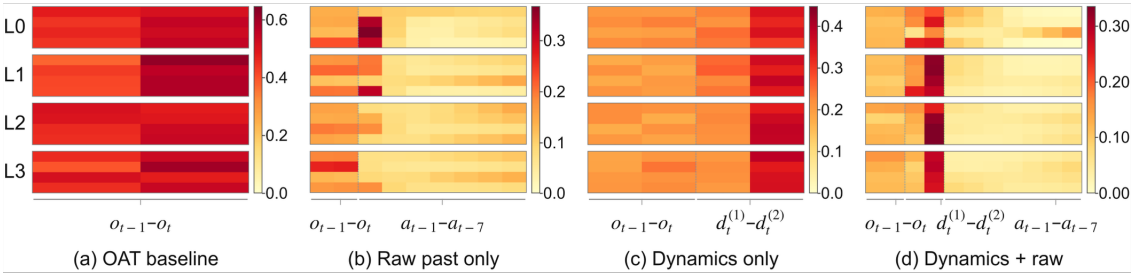

Figure 3: Attention over the conditioning tokens for the four variants in Fig. 2. L0–L3 denote Transformer layers; rows within each layer correspond to attention heads, and columns to conditioning tokens. Warmer colors indicate larger attention weights; color scales differ across panels.

4 Why Past Actions Help: A Dynamics View

We explain the results of Sec. 3 from robot dynamics, for the six continuous pose channels of the end-effector delta action $a_t \in \mathbb{R}^7$. The manipulator dynamics Lynch & Park (2017) give the joint acceleration

$$\ddot{q} = M(q)^{-1}\big[\tau - C(q,\dot{q})\dot{q} - g(q)\big] \tag{1}$$

(inertia $M \succ 0$, Coriolis term $C\dot{q}$, gravity g, torque τ). By Eq. 1, $\ddot{q}$ is bounded on a bounded operating region with bounded torque, and forward kinematics makes the end-effector pose $x(t)$ smooth (App. C). With the controller tracking each pose increment within a control period Δt, $a_t \approx \dot{x}(t)\Delta t$, so the action stream is smooth and can be extrapolated from its recent values. Fitting a polynomial through the K most recent actions and evaluating it one step ahead gives $\hat{a}_t^{(K)}$ (App. D); for $K = 3$,

$$\hat{a}_t^{(3)} = a_{t-1} + d_t^{(1)} + d_t^{(2)}, \tag{2}$$

where $d_t^{(1)}$ and $d_t^{(2)}$ are the dynamics features of Sec. 3, i.e., acceleration- and jerk-level trends. Eq. 2 shows that the latest action and the two dynamics features are exactly a quadratic extrapolation of the next action; it also holds for normalized actions (App. D).

To quantify the extrapolation error, we model the demonstrated actions as band-limited.

Assumption 4.1 (Band-limited demonstrations)**.** Over each control window, the continuous part of the demonstrated action stream $a(t) \in \mathbb{R}^6$ is the restriction of a signal whose components have Fourier support in $[-2\pi f_{\max}, 2\pi f_{\max}]$, with $\|a\|_\infty < \infty$.

Proposition 4.2 (Past-action contraction)**.** *Let $\rho := 2\pi f_{\max}\Delta t$. Under Assumption 4.1,*

$$\left\| a_t - \hat{a}_t^{(K)} \right\|_\infty \le \rho^K \, \|a\|_\infty . \tag{3}$$

For $\rho < 1$, Eq. 3 confines the next action to a neighborhood of the extrapolation whose radius shrinks geometrically with K (proof in App. E). On LIBERO-Long, the 99% spectral-energy cutoff is $f_{\max} = 1.58$ Hz, so $\rho \approx 0.50$ at $\Delta t = 0.05$ s, and the three-point extrapolation of Eq. 2 is within about 12.5% of the action scale. Higher orders tighten the bound but amplify recording noise (App. D), so the latest three actions carry most of the predictive information. Since real demonstrations are not strictly band-limited, the bound is a scale estimate rather than a guarantee (App. A). The dynamics features add no information beyond the raw history; they give the policy the extrapolation terms directly, while the raw window keeps absolute action levels and longer context, consistent with the gain of the combined setting and the attention shift in Sec. 3. The policy is not constrained to follow the extrapolation. A smaller set of feasible next actions should leave fewer plausible tokens per step, i.e., a lower conditional entropy $\mathcal{H}(z_i \mid c_t, z_{<i})$; Sec. 6.3 measures a drop from 2.87 to 1.60 nats.

5 METHOD

Past2Next trains in two stages (Fig. 4) that target the two bottlenecks identified in the introduction. Stage 1 reduces tokenizer distortion with $SO(3)$ augmentation; Stage 2 freezes the tokenizer and makes each token easier to predict by conditioning on the local motion information analyzed in Sec. 4.

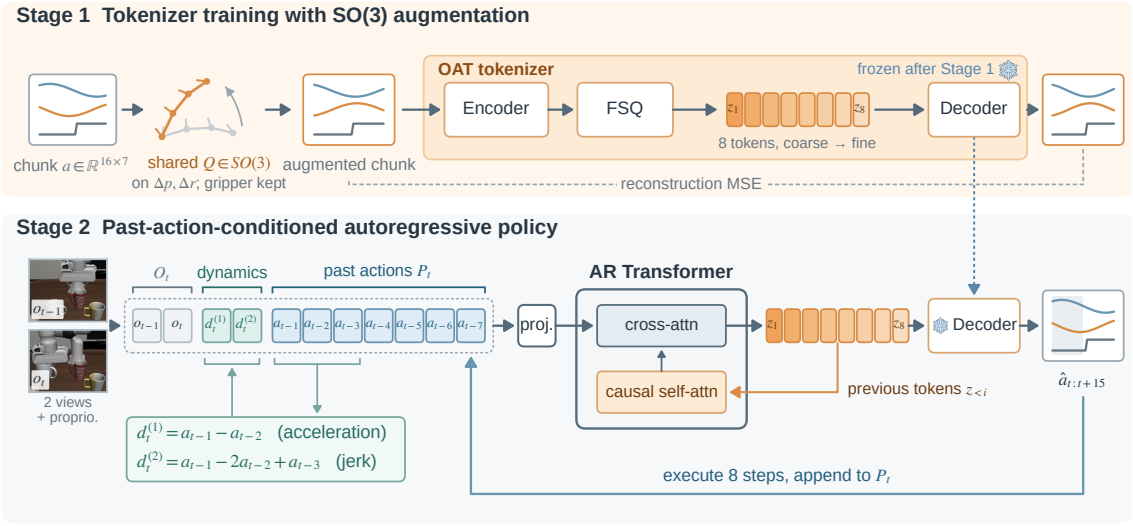

Figure 4: Past2Next overview. Stage 1 trains an action tokenizer with $SO(3)$ augmentation, then freezes it. Stage 2 predicts action tokens from observations, dynamics features, and past actions; the frozen decoder reconstructs the predicted chunk.

Stage 1: Tokenizer training with $SO(3)$ augmentation. Each chunk $A_t = a_{t:t+15} \in \mathbb{R}^{16\times7}$ holds 16 actions, each with a translation Δp_k, a rotation vector Δr_k, and a gripper command. During training, we rotate the whole chunk by one random rotation $Q \in SO(3)$: $\Delta p'_k = Q\Delta p_k$ and $\Delta r'_k = Q\Delta r_k$ for all 16 steps (equivalently, $R'_k = QR_kQ^\top$ for the rotation matrices), while the gripper commands stay unchanged. The shared Q keeps the shape of the motion and changes only its direction, broadening the tokenizer's coverage of the action manifold. The tokenizer follows OAT Liu et al. (2026a): a causal register-attention Transformer Darcet et al. (2024) encodes the normalized chunk, finite scalar quantization (FSQ) Mentzer et al. (2024) maps it to eight discrete tokens $z_{1:8} = T(A_t)$, ordered from coarse to fine by nested dropout Rippel et al. (2014), and a Transformer decoder D reconstructs the chunk under a mean-squared loss.

Stage 2: Past-action-conditioned token prediction. Let $P_t = a_{t-7:t-1}$ be the seven most recent actions and $\bar{a}_j = N(a_j)$ their normalized values. The dynamics features $d_t^{(1)} = \bar{a}_{t-1} - \bar{a}_{t-2}$ and $d_t^{(2)} = \bar{a}_{t-1} - 2\bar{a}_{t-2} + \bar{a}_{t-3}$ are the extrapolation terms of Eq. 2. For each observation in $O_t = (o_{t-1}, o_t)$, we concatenate features from both camera views, the end-effector pose, the gripper state, and the task identifier into one vector. The condition sequence stacks these two observation vectors, the two dynamics features embedded by separate MLPs, and the seven normalized past actions embedded by a shared MLP. A shared linear projection, learned positional embeddings that mark feature type and time, and a position-wise MLP map it to the cross-attention memory M_t, which serves as the context c_t of Sec. 4.

With the frozen tokenizer providing target tokens $z_{1:L} = T(A_t)$, $L = 8$, the policy factorizes

$$p_\theta(z_{1:L} \mid O_t, P_t) = \prod_{i=1}^{L} p_\theta(z_i \mid z_{<i}, M_t). \tag{4}$$

Each Transformer block applies causal self-attention over the token prefix $z_{<i}$ and then cross-attention to M_t, so every factor in Eq. 4 sees the observations, the dynamics features, and the raw history. We train the observation encoder, feature embeddings, and Transformer with teacher forcing and token cross-entropy, taking P_t from the demonstration commands preceding A_t. At inference, the policy generates eight tokens, D decodes them into a 16-step chunk, and after denormalization the first eight actions are executed and appended to the history, whose latest seven entries form the next P_t (Fig. 4).

6 Experiments

6.1 Simulation Benchmarks

Setting and metrics. Following OAT Liu et al. (2026a), we evaluate on the ten long-horizon tasks of LIBERO-Long Liu et al. (2023) (50 demonstrations per task) and ten atomic RoboCasa tasks Nasiriany et al. (2024) (200 demonstrations per task: 50 human and 150 from MimicGen Mandlekar et al. (2023)); see Fig. 5. LIBERO-Long tasks chain several subtasks in household scenes, such as turning on a stove and placing a moka pot on it, while the RoboCasa tasks operate drawers, doors, faucets, a microwave, and a coffee machine across varied kitchen layouts and objects. Policies observe an in-hand and a side-view camera; for every method, we train chunk sizes 16 and 32. We report the $H = 32$ result for OAT on LIBERO-Long and the better chunk size for the remaining results. We run 50 test episodes per task at each checkpoint and report the best mean success rate.

Baselines. We compare against three families of policies: discrete-token autoregressive policies (FAST Pertsch et al. (2025), Dense Policy Su et al. (2025), AR-VLA Hu et al. (2026b), and OAT Liu et al. (2026a)), continuous generative policies (DP Chi et al. (2025), a vanilla flow-matching policy (Flow) Liu et al. (2022) and its variant with our past-action conditioning, $\mathrm{Flow}_{\mathrm{past}}$, and A2A Jia et al. (2026)), and the continuous action-chunking policy ACT Zhao et al. (2023). Among these methods, AR-VLA and A2A provide the closest history-aware comparisons. AR-VLA encodes historical robot states as autoregressive context, whereas A2A uses past actions to construct the source prior for flow matching. These methods evaluate temporal context in discrete and continuous formulations, respectively, but do not derive action history as a dynamics-based localization prior or analyze its relationship to next-token predictive entropy. Thus, AR-VLA tests whether generic historical-state conditioning is sufficient within an autoregressive policy, while A2A distinguishes Past2Next from a history-informed continuous flow model without discrete tokenization. All methods are trained and evaluated under the same protocol. Table 2 reports the results.

(a) LIBERO-Long (10) (b) RoboCasa



Figure 5: Simulation setups. We evaluate Past2Next on two robotic manipulation benchmarks: LIBERO-Long (10) and RoboCasa.

Results. Table 2 supports three conclusions. **(1) Past2Next is the best policy on both benchmarks.** It reaches 85.5% on LIBERO-Long and 66.3% on RoboCasa, 10.7 and 3.7 points above the strongest baseline, A2A, and 29.2 and 9.5 points above OAT. **(2) The past-action prior is not specific to discrete tokens.** Adding the same conditioning to Flow raises its success rate by 26.2 points on LIBERO-Long (39.0% $\rightarrow$ 65.2%) and by 2.3 on RoboCasa, and adding it to DP gives +6.8 on LIBERO-Long (Table 6). **(3) Reconstruction fidelity and generic history do not make tokens predictable.** FAST, whose tokenizer targets compression fidelity, is the weakest method on both benchmarks (23.0% and 21.3%), and AR-VLA, which conditions on past robot states, trails OAT on both, consistent with Secs. 3–4. Per-task results are in App. B.

Policy	**LIBERO-Long (10)** Avg.	**RoboCasa** Avg.
DP	36.6 ± 0.2	46.6 ± 1.0
Flow	39.0 ± 1.2	50.3 ± 1.0
$\text{Flow}_{\text{past}}$	65.2 ± 1.4	52.6 ± 0.9
A2A	74.8 ± 2.8	62.6 ± 2.0
FAST	23.0 ± 0.5	21.3 ± 1.1
ACT	64.1 ± 2.2	44.6 ± 0.9
AR-VLA	35.9 ± 0.5	43.1 ± 2.6
Dense	67.7 ± 2.9	58.8 ± 1.5
OAT	56.3 ± 1.0	56.8 ± 1.2
Past2Next	**85.5** ± 0.4	**66.3** ± 0.8

Table 2: Average success rate (%) $\pm$ standard error on LIBERO-Long (10) and RoboCasa.

6.2 Real World Evaluation

Setup. We evaluate three tabletop tasks on a Piper-X arm, each a four-stage sequence (Fig. 6): Prepare Fruit, Pen to Drawer, and Nut Washer Removal, trained with 40, 70, and 80 demonstrations. We compare Past2Next with the observation-only DP and OAT over 20 trials per task and report success rate and inference time per action chunk (Table 3).

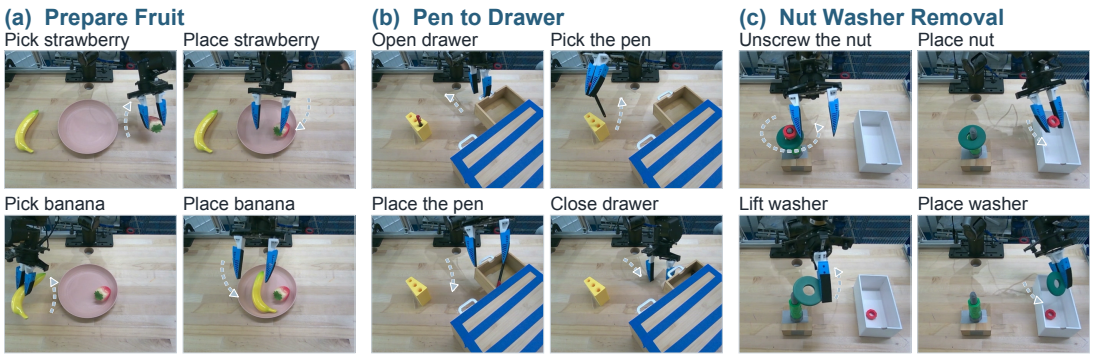

Figure 6: Real-world setups. We validate Past2Next on three tabletop manipulation tasks using a Piper-X arm: (a) Prepare Fruit, (b) Pen to Drawer, and (c) Nut Washer Removal.

Policy	Success rate			Inference time (ms)
	Prepare Fruit (40 demos)	Pen to Drawer (70 demos)	Nut Washer Removal (80 demos)	
DP	12/20	10/20	8/20	40.7
OAT	14/20	12/20	6/20	27.2
Past2Next	18/20	16/20	15/20	27.4

Table 3: Evaluation results on three real-world tasks, for the DP we apply DDIM with 10 denoising steps.

Results. On the two multi-stage pick-and-place tasks, Prepare Fruit (Fig. 6(a)) and Pen to Drawer (Fig. 6(b)), Past2Next succeeds in 18 and 16 of 20 trials, four more than OAT and six more than DP on each task, at nearly the same inference time as OAT (27.4 vs. 27.2 ms). **The largest gap is on Nut Washer Removal** (Fig. 6(c)), where Past2Next succeeds in 15 of 20 trials against 8 for DP and 6 for OAT. The nut must be unscrewed before the washer can be lifted, and nuts are mounted tight or loose, so the number of turns varies across trials. When the nut comes free is hard to see: frames just before and after look nearly the same. The observation-only policies tend to replay the number of turns seen in the demonstrations and fail when a nut needs more or fewer. Past2Next conditions on its recent commands together with the observed state. We attribute its success to this dynamics signal, which indicates whether the latest turns are still loosening the nut, so the policy adjusts its attempts and moves on once the nut is free.

6.3 Token Predictability Analysis

Conditional entropy. With the same OAT tokenizer ($|V| = 1000$), past-action conditioning lowers the per-step conditional entropy of next-token prediction from 2.87 to 1.60 nats on average, shrinking the effective branching factor $e^{\mathcal{H}}$ from about 18 to about 5 candidate tokens (Fig. 7). The entropy is lower at every token position and the per-sample distributions barely overlap, the token-level counterpart of the contraction in Sec. 4.

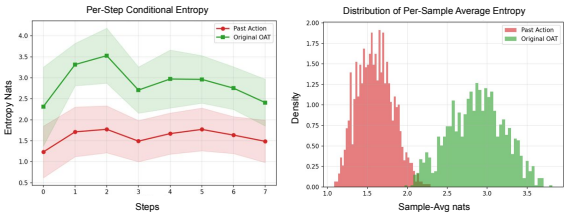

Figure 7: Next-token entropy per step (left) and per sample (right), $|V|$=1000.

Codebook scaling. Codebook size sets the action resolution: a larger codebook quantizes actions more finely, so tokens can express smaller, more precise motions, but each prediction must choose among more candidates. Conditioning decides which effect wins (Table 4). At $H = 32$, OAT peaks at $|V| = 1000$ (56.3%) and falls to 46.9% at $|V| = 5000$, whereas the conditioned policy is best at $|V| = 5000$ (73.4%). We attribute the reversal to conditioning keeping the next-token distribution concentrated, so the finer actions become usable (full cross-ablation in Table 5).

$\|V\|$	OAT	Past2Next
240	29.2	51.2
512	53.5	58.0
1000	56.3	68.8
1920	54.6	63.2
5000	46.9	**73.4**

Table 4: **Codebook size ($H = 32$).** LIBERO-Long success rate (%); Past2Next w/o $SO(3)$ augmentation.

7 Limitations and Conclusion

In this work, we presented **Past2Next**, an autoregressive discrete-token policy built on a simple observation: success depends not only on how faithfully the tokenizer reconstructs actions, but on how predictable the resulting tokens are. Past2Next addresses both axes: data augmentation broadens codebook coverage to lower reconstruction distortion, and past-action conditioning uses the local smoothness of robot motion to shrink the set of plausible next actions, lowering their predictive Shannon entropy. Furthermore, a larger codebook now helps rather than hurts. By modeling local dynamics and respecting action geometry, Past2Next makes an autoregressive policy both more accurate and more predictable.

Limitations: Because we train the tokenizer and policy in two stages rather than end-to-end, an information gap can open between them and may limit performance; relatedly, we have established scaling only along the codebook size and leave joint data–codebook scaling to future work.

AI USE STATEMENT

In this work, we used generative AI tools for proof reading, language polishing, coding, and image adjustment. We have not used generative AI tools for idea search, experiment design, and real world robot experiment. We have reviewed all AI-assisted work. We checked LLM-generated language polishing for potential plagiarism through a manual literature survey. LLM-generated code was verified and tested for correctness by two authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI. See the ICLR 2027 AI Policy for Authors for more details.

REFERENCES

Bo Ai, Stephen Tian, Haochen Shi, Yixuan Wang, Tobias Pfaff, Cheston Tan, Henrik I Christensen, Hao Su, Jiajun Wu, and Yunzhu Li. A review of learning-based dynamics models for robotic manipulation. *Science Robotics*, 10(106):eadt1497, 2025.

Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. Univla: Learning to act anywhere with task-centric latent actions. *arXiv preprint arXiv:2505.06111*, 2025.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

Zichen Jeff Cui, Yibin Wang, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. From play to policy: Conditional behavior generation from uncurated robot data. *arXiv preprint arXiv:2210.10047*, 2022. URL https://arxiv.org/abs/2210.10047.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *International conference on learning representations*, volume 2024, pp. 2632–2652, 2024.

Zibin Dong, Yicheng Liu, Shiduo Zhang, Baijun Ye, Yifu Yuan, Fei Ni, Jingjing Gong, Xipeng Qiu, Hang Zhao, Yinchuan Li, et al. Actioncodec: What makes for good action tokenizers. *arXiv preprint arXiv:2602.15397*, 2026.

Chongkai Gao, Zhengrong Xue, Shuying Deng, Tianhai Liang, Siqi Yang, Lin Shao, and Huazhe Xu. Riemann: Near real-time se (3)-equivariant robot manipulation without point cloud segmentation. *arXiv preprint arXiv:2403.19460*, 2024.

Zhefei Gong, Pengxiang Ding, Shangke Lyu, Siteng Huang, Mingyang Sun, Wei Zhao, Zhaoxin Fan, and Donglin Wang. Carp: Visuomotor policy learning via coarse-to-fine autoregressive prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13460–13470, 2025.

Boce Hu, Xupeng Zhu, Dian Wang, Zihao Dong, Haojie Huang, Chenghao Wang, Robin Walters, and Robert Platt. Orbitgrasp: $se(3)$-equivariant grasp learning. *arXiv preprint arXiv:2407.03531*, 2024.

Boce Hu, Heng Tian, Dian Wang, Haojie Huang, Xupeng Zhu, Robin Walters, and Robert Platt. Push-grasp policy learning using equivariant models and grasp score optimization. *IEEE Robotics and Automation Letters*, 2025.

Boce Hu, Dian Wang, David Klee, Heng Tian, Xupeng Zhu, Haojie Huang, Robert Platt, and Robin Walters. 3d equivariant visuomotor policy learning via spherical projection. *Advances in Neural Information Processing Systems*, 38:5685–5715, 2026a.

Yutong Hu, Jan-Nico Zaech, Nikolay Nikolov, Yuanqi Yao, Sombit Dey, Giuliano Albanese, Renaud Detry, Luc Van Gool, and Danda Paudel. Ar-vla: True autoregressive action expert for vision-language-action models. *arXiv preprint arXiv:2603.10126*, 2026b.

Haojie Huang, Dian Wang, Robin Walters, and Robert Platt. Equivariant transporter network. *arXiv preprint arXiv:2202.09400*, 2022.

Haojie Huang, Karl Schmeckpeper, Dian Wang, Ondrej Biza, Yaoyao Qian, Haotian Liu, Mingxi Jia, Robert Platt, and Robin Walters. Imagination policy: Using generative point cloud models for learning manipulation policies. *arXiv preprint arXiv:2406.11740*, 2024.

Haojie Huang, Haotian Liu, Dian Wang, Robin Walters, and Robert Platt. Match policy: A simple pipeline from point cloud registration to manipulation policies. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 16907–16914. IEEE, 2025.

Haojie Huang, Linfeng Zhao, Haotian Liu, Zhang Ye, Si-Yuan Huang, Mingxi Jia, Boce Hu, Fangzhou Lin, Yu Qi, Dian Wang, et al. Pix2act: Image-space manipulation policies with equivariant augmentation. *arXiv preprint arXiv:2607.11167*, 2026.

Jindou Jia, Gen Li, Xiangyu Chen, Tuo An, Yuxuan Hu, Jingliang Li, Xinying Guo, and Jianfei Yang. Action-to-action flow matching. *arXiv preprint arXiv:2602.07322*, 2026.

Sunshine Jiang, Xiaolin Fang, Nicholas Roy, Tomás Lozano-Pérez, Leslie Pack Kaelbling, and Siddharth Ancha. Streaming flow policy: Simplifying diffusion/flow-matching policies by treating action trajectories as flow trajectories. *arXiv preprint arXiv:2505.21851*, 2025.

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. URL `https://arxiv.org/abs/2406.09246`.

David Klee, Boce Hu, Andrew Cole, Heng Tian, Dian Wang, Robert Platt, and Robin Walters. Raven: End-to-end equivariant robot learning with rgb cameras. In *The Fourteenth International Conference on Learning Representations*, 2026.

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

Chaoqi Liu, Xiaoshen Han, Jiawei Gao, Yue Zhao, Haonan Chen, and Yilun Du. Oat: Ordered action tokenization. *arXiv preprint arXiv:2602.04215*, 2026a.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Yufeng Liu, Hang Yu, Juntu Zhao, Bocheng Li, Di Zhang, Mingzhu Li, Wenxuan Wu, Yingdong Hu, Junyuan Xie, Junliang Guo, Dequan Wang, and Yang Gao. Learning native continuation for action chunking flow policies. *arXiv preprint arXiv:2602.12978*, 2026b.

Kevin M Lynch and Frank C Park. *Modern robotics*. Cambridge University Press, 2017.

Robert Mahony, Tarek Hamel, and Jochen Trumpf. Equivariant systems theory and observer design. *arXiv preprint arXiv:2006.08276*, 2020.

Ajay Mandlekar, Soroush Nasiriany, Bowen Wen, Iretiolu Akinola, Yashraj Narang, Linxi Fan, Yuke Zhu, and Dieter Fox. Mimicgen: A data generation system for scalable robot learning using human demonstrations. In *Conference on Robot Learning (CoRL)*, 2023.

Fabian Mentzer, David Minnen, Eirikur Agustsson, and Michael Tschannen. Finite scalar quantization: Vq-vae made simple. In *International Conference on Learning Representations*, volume 2024, pp. 51772–51783, 2024.

Atharva Mete, Haotian Xue, Albert Wilcox, Yongxin Chen, and Animesh Garg. Quest: Self-supervised skill abstractions for learning continuous control. *Advances in Neural Information Processing Systems*, 37: 4062–4089, 2024.

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*, 2024.

Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. Fast: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025.

Oren Rippel, Michael Gelbart, and Ryan Adams. Learning ordered representations with nested dropout. In *International Conference on Machine Learning*, pp. 1746–1754. PMLR, 2014.

Joohwan Seo, Nikhil PS Prakash, Xiang Zhang, Changhao Wang, Jongeun Choi, Masayoshi Tomizuka, and Roberto Horowitz. Contact-rich se (3)-equivariant robot manipulation task learning via geometric impedance control. *IEEE Robotics and Automation Letters*, 9(2):1508–1515, 2023.

Joohwan Seo, Soochul Yoo, Junwoo Chang, Hyunseok An, Hyunwoo Ryu, Soomi Lee, Arvind Kruthiventy, Jongeun Choi, and Roberto Horowitz. Se (3)-equivariant robot learning and control: a tutorial survey. *International Journal of Control, Automation and Systems*, 23(5):1271–1306, 2025.

Hao Shi, Bin Xie, Yingfei Liu, Lin Sun, Fengrong Liu, Tiancai Wang, Erjin Zhou, Haoqiang Fan, Xiangyu Zhang, and Gao Huang. Memoryvla: Perceptual-cognitive memory in vision-language-action models for robotic manipulation. *arXiv preprint arXiv:2508.19236*, 2025.

Anthony Simeonov, Yilun Du, Andrea Tagliasacchi, Joshua B Tenenbaum, Alberto Rodriguez, Pulkit Agrawal, and Vincent Sitzmann. Neural descriptor fields: Se (3)-equivariant object representations for manipulation. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 6394–6400. IEEE, 2022.

Mingchen Song, Xiang Deng, Zhiling Zhou, Jie Wei, Weili Guan, and Liqiang Nie. A survey on diffusion policy for robotic manipulation: Taxonomy, analysis, and future directions. *Authorea Preprints*, 2025.

Yue Su, Xinyu Zhan, Hongjie Fang, Han Xue, Hao-Shu Fang, Yong-Lu Li, Cewu Lu, and Lixin Yang. Dense policy: Bidirectional autoregressive learning of actions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14486–14495, 2025.

Xudong Tan, Yaoxin Yang, Peng Ye, Jialin Zheng, Bizhe Bai, Xinyi Wang, Jia Hao, and Tao Chen. Think twice, act once: Token-aware compression and action reuse for efficient inference in vision-language-action models. *arXiv preprint arXiv:2505.21200*, 2025.

Marcel Torne, Andy Tang, Yuejiang Liu, and Chelsea Finn. Learning long-context diffusion policies via past-token prediction. *arXiv preprint arXiv:2505.09561*, 2025.

Julen Urain, Ajay Mandlekar, Yilun Du, Mahi Shafiullah, Danfei Xu, Katerina Fragkiadaki, Georgia Chalvatzaki, and Jan Peters. Deep generative models in robotics: A survey on learning from multimodal demonstrations. *arXiv preprint arXiv:2408.04380*, 2024.

Dian Wang, Mingxi Jia, Xupeng Zhu, Robin Walters, and Robert Platt. On-robot learning with equivariant models. *arXiv preprint arXiv:2203.04923*, 2022a.

Dian Wang, Robin Walters, Xupeng Zhu, and Robert Platt. Equivariant q learning in spatial action spaces. In *Conference on Robot Learning*, pp. 1713–1723. PMLR, 2022b.

Dian Wang, Stephen Hart, David Surovik, Tarik Kelestemur, Haojie Huang, Haibo Zhao, Mark Yeatman, Jiuguang Wang, Robin Walters, and Robert Platt. Equivariant diffusion policy. *arXiv preprint arXiv:2407.01812*, 2024.

Ge Yan, Jiyue Zhu, Yuquan Deng, Shiqi Yang, Ri-Zhao Qiu, Xuxin Cheng, Marius Memmel, Ranjay Krishna, Ankit Goyal, Xiaolong Wang, et al. Maniflow: A general robot manipulation policy via consistency flow training. *arXiv preprint arXiv:2509.01819*, 2025.

Richard N Youngworth, Benjamin B Gallagher, and Brian L Stamper. An overview of power spectral density (psd) calculations. *Optical manufacturing and testing VI*, 5869:206–216, 2005.

Xinyu Zhang, Yuhan Liu, Haonan Chang, Liam Schramm, and Abdeslam Boularias. Autoregressive action sequence learning for robotic manipulation. *IEEE Robotics and Automation Letters*, 10(5):4898–4905, 2025. doi: 10.1109/LRA.2025.3550849.

Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

Xupeng Zhu, Yu Qi, Yizhe Zhu, Robin Walters, and Robert Platt. Equact: An se (3)-equivariant multi-task transformer for open-loop robotic manipulation. *arXiv preprint arXiv:2505.21351*, 2025a.

Xupeng Zhu, Fan Wang, Robin Walters, and Jane Shi. Se (3)-equivariant diffusion policy in spherical fourier space. *arXiv preprint arXiv:2507.01723*, 2025b.

A Ablation Study

Table 5 crosses past-action conditioning with the two capacity axes of an AR discrete-token policy—codebook size $|V| \in \{1000, 5000\}$ and chunk size $H \in \{16, 32\}$—on LIBERO-Long and adds $SO(3)$ augmentation to the best configuration; Table 6 then transplants the conditioning into continuous policies.

LIBERO-Long (10)			
Policy	Chunk Size	Codebook Size	Avg.
OAT	32	1000	56.3
	32	5000	46.9
	16	1000	56.4
	16	5000	46.9
Past2Next w/o $SO(3)$	32	1000	68.8
	32	5000	73.4
	16	1000	68.6
	16	5000	79.4
Past2Next	16	5000	**85.5**

Table 5: Cross-ablation of past-action conditioning, chunk size H, and codebook size $|V|$ on LIBERO-Long (success rate, %). "Past2Next w/o $SO(3)$" denotes the enriched-past policy on the unaugmented OAT tokenizer; "Past2Next" adds $SO(3)$ augmentation in Stage 1 to the best configuration. Best in bold.

LIBERO-Long (10)											
Policy	Soup/Sauce Basket	Cheese/Butter Basket	Soup/Cheese Basket	Two Moka Pots	Stove & Moka	Bowl to Drawer	Mugs on Plates	Book to Caddy	Mug & Pudding	Mug to Micro	Avg.
Flow	14	32	38	22	54	74	12	58	24	62	39.0
$\text{Flow}_{\text{past}}$	32	54	82	52	80	84	62	74	58	74	65.2
DP	26.0	18.8	24.8	52.4	56.8	62.8	20.0	18.4	29.6	56.0	36.6
DP_{past}	28	32	34	60	58	64	38	24	28	68	43.4

Table 6: Past-action conditioning transplanted into continuous policies on LIBERO-Long (success rate, %). The "past" variants add the same enriched-past context as Past2Next (raw window with acceleration and jerk features) and keep their original noise sources.

The past action order selection. We use a raw history of $K_{\text{hist}} = 7$ actions and compute first- and second-order differences from its three most recent samples (Fig. 2(d)). These features describe acceleration- and jerk-level trends up to fixed sampling and normalization factors. Their order reflects a tradeoff between local extrapolation accuracy and noise amplification. For a predictor built from K past samples, Proposition 4.2 gives the error bound $\rho^K \|a\|_\infty$ under the band-limited assumption. The $\rho^K/K!$ Taylor bound would instead require exact derivatives and does not apply to this sample-based extrapolator.

On LIBERO-Long demonstrations, we measure $f_{\max} = 1.58\,\text{Hz}$ as the 99% spectral-energy cutoff of a Welch power spectral density estimate Youngworth et al. (2005) over the continuous pose channels. At $\Delta t = 0.05\,\text{s}$, this gives $\rho \approx 0.50$ (Table 7). Under the corresponding band-limited approximation, the relative error bound is approximately 50%, 25%, and 12.5% for $K = 1, 2, 3$, respectively. These are conditional bounds, not measured prediction errors. The cutoff retains 99% of spectral energy; the stricter 99.9% cutoff gives $\rho > 1$, so the geometric contraction guarantee depends on the approximation.

Higher-order differences amplify recording noise. With independent per-step noise of variance σ^2, the m-th backward difference has noise variance $\binom{2m}{m}\sigma^2$ (Lemma D.2). The corresponding variances are $2\sigma^2$, $6\sigma^2$, and $20\sigma^2$ for orders one, two, and three; derivative scaling adds a factor Δt^{-2m}. This growth supports using

low-order features, but the bound alone does not identify a unique optimal order or establish an absolute signal-to-noise threshold.

The raw window complements the differences by preserving absolute action levels and a longer temporal context. Differences computed from the latest three samples expose local trends directly, while the seven-step history retains information about the preceding command sequence. The policy learns how to combine these inputs rather than following a fixed extrapolation rule.

Spectral-energy cutoff q	95%	99%	99.9%
$f_{\max}$ (Hz)	0.76	**1.58**	4.08
$\rho = 2\pi f_{\max}\Delta t$	0.24	**0.50**	1.28

Table 7: Spectral-energy cutoffs of LIBERO-Long demonstration actions and the corresponding ρ at $\Delta t = 0.05\,\mathrm{s}$.

Past-action conditioning decides the sign of codebook scaling. Enlarging the codebook fivefold moves the baseline and the conditioned policy in opposite directions. For OAT, $|V| : 1000 \to 5000$ *costs* roughly ten points at either chunk size ($56.3 \to 46.9$ and $56.4 \to 46.9$): the finer codebook lowers reconstruction distortion, but the AR head must now discriminate among more admissible codes per step than its observation-only context can resolve, and the estimation error swamps the distortion gain. Under enriched-past conditioning the same enlargement *gains* $+4.6$ at $H{=}32$ ($68.8 \to 73.4$) and $+10.8$ at $H{=}16$ ($68.6 \to 79.4$): success rate increases with $|V|$ at both chunk sizes, and at the ($H{=}16$, $|V|{=}5000$) configuration conditioning alone is worth $+32.5$ points. We attribute this sign flip, obtained with the tokenizer family held fixed, to the contraction of Proposition 4.2: conditioning shrinks the per-step set of plausible codes, so the finer codebook's lower distortion is no longer offset by harder token prediction. We note that the baseline's degradation is not codebook under-utilization—FSQ keeps utilization near 100%—but a prediction-side failure, consistent with our diagnosis that the bottleneck is conditional entropy rather than reconstruction quality.

$SO(3)$ **augmentation adds an orthogonal gain.** On the strongest conditioned configuration ($H{=}16$, $|V|{=}5000$), $SO(3)$ augmentation of the tokenizer adds a further $+6.1$ points ($79.4 \to 85.5$), yielding the full Past2Next result. The gain is consistent with the augmentation acting on the distortion axis—broader codebook coverage of the rotation manifold—orthogonal to, and therefore stacking with, the conditioning axis.

B Experiment Details

LIBERO-Long (10)											
Policy	Soup/Sauce Basket	Cheese/Butter Basket	Soup/Cheese Basket	Two Moka Pots	Stove & Moka	Bowl to Drawer	Mugs on Plates	Book to Caddy	Mug & Pudding	Mug to Micro	Avg.
DP	26.0 ± 3.0	18.8 ± 1.4	24.8 ± 1.9	52.4 ± 2.7	56.8 ± 3.4	62.8 ± 2.1	20.0 ± 1.7	18.4 ± 1.9	29.6 ± 2.6	56.0 ± 3.1	36.6 ± 0.2
Flow	14.0 ± 1.5	32.0 ± 3.4	38.0 ± 4.1	22.0 ± 2.4	54.0 ± 4.6	74.0 ± 3.2	12.0 ± 1.3	58.0 ± 4.4	24.0 ± 2.5	62.0 ± 4.0	39.0 ± 1.2
$\text{Flow}_{\text{past}}$	32 ± 1.2	54 ± 2.6	82 ± 3.6	52 ± 1.6	80 ± 3.3	84 ± 1.6	62 ± 2.2	74 ± 1.6	58 ± 2.4	74 ± 3.1	65.2 ± 1.4
A2A	16.4 ± 1.6	71 ± 4.2	91.5 ± 3.4	79.5 ± 6.6	81.5 ± 7.2	93.5 ± 4.4	84.0 ± 5.9	96.5 ± 3.4	50.5 ± 6.8	83.5 ± 4.1	74.8 ± 2.8
FAST	14.8 ± 1.6	6.4 ± 0.7	1.6 ± 0.7	33.6 ± 4.4	52.8 ± 1.4	50.4 ± 5.0	16.0 ± 1.7	28.4 ± 1.7	22.0 ± 3.5	4.4 ± 1.3	23.0 ± 0.5
ACT	61.5 ± 8.5	75 ± 8.2	63.5 ± 6.6	48.5 ± 3.4	74.5 ± 10.6	73.5 ± 3.4	43.5 ± 11.8	83.5 ± 17.1	42.0 ± 10.6	75 ± 3.5	64.1 ± 2.2
AR-VLA	10.5 ± 1.9	47.0 ± 4.2	7.0 ± 1.2	21.5 ± 1.9	58.0 ± 2.3	68.5 ± 3.0	11.0 ± 2.6	57.5 ± 1.9	20.0 ± 3.7	58.0 ± 2.3	35.9 ± 0.5
Dense	31.5 ± 8.5	54.0 ± 9.8	88.5 ± 5.5	55.5 ± 11.0	92.5 ± 3.0	96.0 ± 2.8	43.0 ± 4.7	94.5 ± 2.5	43.0 ± 4.7	78.5 ± 9.5	67.7 ± 2.9
OAT	26.8 ± 3.2	35.6 ± 2.6	51.6 ± 2.2	61.2 ± 4.3	87.6 ± 1.2	91.2 ± 1.0	31.2 ± 2.7	70.8 ± 4.5	32.0 ± 2.8	75.2 ± 3.8	56.3 ± 1.0
Past2Next	73.0 ± 2.6	71 ± 3.5	92 ± 1.6	82.5 ± 2.5	97 ± 1.2	95.5 ± 1.0	82.5 ± 3.0	91.5 ± 3.0	81.0 ± 2.6	88.5 ± 1.0	85.5 ± 0.4

Table 8: Per-task success rate (%) ± standard error on LIBERO-Long (10).

	RoboCasa										
Policy	Close Drawer	Coffee Press Button	Turn Off Microwave	Turn Off Sink Faucet	Turn On Sink Faucet	Turn Sink Spout	Open Single Door	Open Double Door	Close Single Door	Close Double Door	Avg.
DP	52.0 ± 2.8	56.8 ± 2.8	52.8 ± 3.5	54.4 ± 1.8	43.1 ± 1.4	45.3 ± 1.7	39.0 ± 2.4	15.5 ± 4.5	52.3 ± 5.2	54.8 ± 3.4	46.6 ± 1.0
Flow	63.2 ± 1.9	49.6 ± 2.0	50.8 ± 3.1	72.8 ± 1.8	46.3 ± 1.4	50.5 ± 4.4	40.1 ± 6.2	17.2 ± 1.5	54.6 ± 2.9	58.0 ± 1.9	50.3 ± 1.0
$\text{Flow}_{\text{past}}$	65.1 ± 2.2	51.7 ± 1.6	52.8 ± 4.2	73.4 ± 2.3	48.4 ± 4.1	50.6 ± 2.4	46.2 ± 1.2	18.4 ± 2.0	57.5 ± 2.6	62.0 ± 3.4	52.6 ± 0.9
A2A	89.4 ± 4.2	78.4 ± 9.8	76.0 ± 6.4	96.2 ± 3.2	73.4 ± 8.9	77.6 ± 7.6	31.5 ± 3.8	26.5 ± 6.5	41.0 ± 1.7	35.5 ± 6.2	62.6 ± 2.0
Fast	20.8 ± 2.0	8.4 ± 1.5	16.8 ± 1.4	6.8 ± 2.1	27.1 ± 4.8	28.8 ± 1.3	23.9 ± 2.6	8.2 ± 3.0	34.9 ± 5.2	37.3 ± 6.5	21.3 ± 1.1
ACT	60.2 ± 2.8	46.1 ± 2.6	48.8 ± 3.3	62.2 ± 3.7	40.2 ± 2.7	40.0 ± 2.5	36.5 ± 2.3	14.2 ± 1.8	48.2 ± 3.0	49.8 ± 3.3	44.6 ± 0.9
AR-VLA	55.0 ± 7.8	38.6 ± 7.5	42.2 ± 9.4	64.4 ± 10.8	38.0 ± 8.2	52.0 ± 7.4	34.0 ± 6.4	12.9 ± 4.2	44.5 ± 8.6	49.5 ± 9.9	43.1 ± 2.6
Dense	71.4 ± 5.2	58.9 ± 4.1	59.9 ± 4.8	78.7 ± 6.2	55.6 ± 4.0	57.8 ± 4.5	51.4 ± 5.8	23.2 ± 2.8	64.4 ± 3.6	66.7 ± 5.0	58.8 ± 1.5
OAT	53.6 ± 3.2	63.6 ± 4.0	42.8 ± 3.6	80.0 ± 4.6	58.0 ± 4.0	50.0 ± 3.4	55.4 ± 3.8	26.2 ± 2.2	68.1 ± 4.2	70.3 ± 4.4	56.8 ± 1.2
Past2Next	78.0 ± 2.4	67.0 ± 3.1	68.0 ± 2.9	84.0 ± 1.9	64.0 ± 2.5	66.0 ± 2.5	60.0 ± 2.6	30.0 ± 2.4	72.0 ± 2.4	74.0 ± 2.3	**66.3** ± 0.8

Table 9: Evaluation results on the RoboCasa benchmark. The final column reports the mean across ten tasks

C Regularity of the End-Effector Motion

For the joint dynamics in Eq. 1, define the state $s = (q^\top, \dot{q}^\top)^\top \in \mathbb{R}^{2n}$. The equivalent first-order system is

$$\dot{s} = f(s,t) := \begin{pmatrix} \dot{q} \\ M(q)^{-1}[\tau(t) - C(q,\dot{q})\dot{q} - g(q)] \end{pmatrix}. \tag{5}$$

Lemma C.1 (Regularity of the end-effector motion)**.** *Let $\mathcal{S} \subset \mathbb{R}^{2n}$ be compact, and assume M, C, and g are smooth on $\mathcal{S}$ with $M(q)$ positive definite there. Let $\tau \in C^k([0,T];\mathbb{R}^n)$ for some $k \geq 0$. Then for any $s(0) \in \mathcal{S}$ the system equation 5 admits a unique solution, which remains C^1 and extends over $[0,T]$ as long as the trajectory stays in $\mathcal{S}$. Moreover $q \in C^{k+2}([0,T];\mathbb{R}^n)$ and $X = \mathrm{FK}(q) \in C^{k+2}([0,T];SE(3))$. For a fixed Δt, the local coordinates of $X(t)^{-1}X(t+\Delta t)$ are C^{k+2} on $[0, T-\Delta t]$ whenever the relative rotation stays inside a smooth chart of $SO(3)$.*

Proof. **Step 1 (well-posedness).** M is continuous and positive definite on the compact set $\mathcal{S}$, so its smallest eigenvalue attains a positive minimum there and $M(q)^{-1}$ exists throughout. By Cramer's rule the entries of $M(q)^{-1}$ are the entries of the adjugate divided by $\det M(q)$; both are polynomial in the entries of M, and the determinant is bounded away from zero on $\mathcal{S}$, so $q \mapsto M(q)^{-1}$ is smooth. Consequently

$$f(s,t) = \begin{pmatrix} \dot{q} \\ M(q)^{-1}[\tau(t) - C(q,\dot{q})\dot{q} - g(q)] \end{pmatrix}$$

is continuous in (s,t) and smooth in s. A smooth function on a compact set has bounded s-derivative there, hence f is Lipschitz in s uniformly in t, and Picard–Lindelöf gives a unique local solution; the standard continuation argument extends it for as long as the trajectory remains in $\mathcal{S}$. Since $\dot{s} = f(s,t)$ with both f and s continuous, $\dot{s}$ is continuous and $s \in C^1$.

Step 2 (bootstrapping). Write $\ddot{q} = F(q,\dot{q},\tau(t))$ with $F(q,v,u) := M(q)^{-1}[u - C(q,v)v - g(q)]$, which is smooth in all three arguments. Step 1 gives $s = (q,\dot{q}) \in C^1$, hence $q \in C^2$. Suppose $q \in C^j$ for some $2 \leq j \leq k+1$. Then $(q,\dot{q}) \in C^{j-1}$ and $\tau \in C^k$ with $j-1 \leq k$, so the right-hand side satisfies $F(q,\dot{q},\tau) \in C^{\min(j-1,\,k)} = C^{j-1}$; therefore $\ddot{q} \in C^{j-1}$ and $q \in C^{j+1}$. Iterating from $j=2$ terminates at $q \in C^{k+2}$, the torque regularity capping the induction.

Step 3 (forward kinematics). For a manipulator with revolute and prismatic joints the product-of-exponentials formula gives $\mathrm{FK}(q) = e^{[\mathcal{A}_1]q_1} \cdots e^{[\mathcal{A}_n]q_n} X_0$, where each $[\mathcal{A}_i]$ is a fixed screw axis in $\mathfrak{se}(3)$. Each matrix exponential is an entire function of its scalar argument and matrix multiplication is polynomial in the entries, so $\mathrm{FK} : \mathbb{R}^n \to SE(3)$ is real-analytic. Composing a real-analytic map with $q \in C^{k+2}$ preserves the class, giving $X \in C^{k+2}([0,T];SE(3))$.

Step 4 (relative motion). Inversion in $SE(3)$ is polynomial in the entries, since $(R,p)^{-1} = (R^\top, -R^\top p)$, and so is the group product; hence $Y(t) \coloneqq X(t)^{-1}X(t+\Delta t)$ is C^{k+2} on $[0, T-\Delta t]$. If Y takes values in an open set on which a chart φ is smooth — for instance the logarithm, which is real-analytic wherever the rotation angle is strictly below π — then $\varphi \circ Y \in C^{k+2}$. □

Remark C.2 (The chart condition in practice)**.** The caveat in Step 4 is not restrictive at control rates: at $\Delta t = 0.05\,\mathrm{s}$ the relative rotation between consecutive end-effector poses is far below π throughout our demonstrations, so $Y(t)$ stays well inside the principal chart with a wide margin. The condition would bind only for motions that reorient the end-effector by more than a half-turn within a single control period.

D THE ONE-STEP EXTRAPOLATOR

Let Δt be the control period and define the backward-difference operator

$$\nabla^0 a_t = a_t, \qquad \nabla^m a_t = \sum_{j=0}^{m} (-1)^j \binom{m}{j} a_{t-j}. \tag{6}$$

Because delta actions are velocity-level, $a_t \approx \dot{x}(t)\Delta t$, the first two action differences encode pose-level acceleration and jerk trends, up to fixed control-period scaling. In the notation of the main text, $d_t^{(1)} = \nabla a_{t-1}$ and $d_t^{(2)} = \nabla^2 a_{t-1}$; the policy computes them from normalized actions (Sec. 5).

Let $\mathcal{E}_K$ be the unique polynomial of degree at most $K-1$ agreeing with the K most recent actions at the nodes $t_j = t - j\Delta t$, $j = 1, \ldots, K$, and let $\hat{a}_t^{(K)}$ denote its value one control step ahead, at time t. Newton's backward-difference formula writes this polynomial, for $x = t_1 + u\Delta t$, as $\sum_{m=0}^{K-1} \binom{u+m-1}{m} \nabla^m a_{t-1}$. Evaluating at $x = t$, that is $u = 1$, every binomial coefficient collapses to $\binom{m}{m} = 1$, so the extrapolator takes the compact form

$$\hat{a}_t^{(K)} = \sum_{m=0}^{K-1} \nabla^m a_{t-1} = \sum_{j=1}^{K} (-1)^{j+1} \binom{K}{j} a_{t-j}. \tag{7}$$

The first few orders are the familiar ones: $\hat{a}_t^{(1)} = a_{t-1}$ (zeroth-order hold), $\hat{a}_t^{(2)} = 2a_{t-1} - a_{t-2}$ (linear), and $\hat{a}_t^{(3)} = 3a_{t-1} - 3a_{t-2} + a_{t-3}$ (quadratic). Since action normalization is a per-channel affine map and the coefficients in equation 7 sum to one, extrapolating normalized actions yields the normalized extrapolation, so Eq. 2 holds for the normalized features as well. Two consequences of equation 7 are used in the main text.

Lemma D.1 (Extrapolation error is a backward difference)**.** *For every $K \geq 1$, $a_t - \hat{a}_t^{(K)} = \nabla^K a_t$.*

Proof. Split the $j = 0$ term out of equation 6: $\nabla^K a_t = a_t - \sum_{j=1}^{K} (-1)^{j+1} \binom{K}{j} a_{t-j}$, and the sum is exactly $\hat{a}_t^{(K)}$ by equation 7. □

Lemma D.2 (Noise amplification of the difference features)**.** *If the recorded actions carry independent per-step noise of variance σ^2, then $\mathrm{Var}[\nabla^m a_t] = \binom{2m}{m}\sigma^2$, so the derivative estimate $\nabla^m a_t / \Delta t^m$ has noise standard deviation $\sigma\sqrt{\binom{2m}{m}}\,\Delta t^{-m}$.*

Proof. The coefficients in equation 6 are $(-1)^j \binom{m}{j}$ and the noise terms are independent, so the variance is $\sigma^2 \sum_{j=0}^{m} \binom{m}{j}^2 = \binom{2m}{m}\sigma^2$ by Vandermonde's identity. □

Remark D.3 (The dynamics features span the extrapolator)**.** By equation 7, $\hat{a}_t^{(3)}$ is a fixed linear combination of a_{t-1}, ∇a_{t-1}, and $\nabla^2 a_{t-1}$. Conditioning the policy on the raw past action together with the scaled first and second differences therefore supplies a basis for the order-2 extrapolator rather than a heuristic summary of the history: the network can realize $\hat{a}_t^{(3)}$ as a linear map of its inputs, and is free to deviate from it where the data warrant.

E Proof of the Past-Action Contraction

We restate the result for convenience.

Proposition E.1 (Past-action contraction, restated)**.** *Let* $\rho := 2\pi f_{\max}\Delta t$. *Under Assumption 4.1,* $\left\| a_t - \hat{a}_t^{(K)} \right\|_\infty \le \rho^K \|a\|_\infty$.

Proof. Work componentwise and take the maximum at the end. By Lemma D.1 the error equals the K-th backward difference, $a_t - \hat{a}_t^{(K)} = \nabla^K a_t$, so it suffices to bound the latter.

For equispaced nodes $t - K\Delta t, \dots, t$ with spacing Δt, the divided difference satisfies $a[t - K\Delta t, \dots, t] = \nabla^K a_t / \big(K!\,(\Delta t)^K\big)$, and the mean-value form of the divided difference gives some $\xi \in (t - K\Delta t,\, t)$ with $a[t - K\Delta t, \dots, t] = a^{(K)}(\xi)/K!$. Equating the two expressions, the factorials cancel and

$$\nabla^K a_t = a^{(K)}(\xi)\,(\Delta t)^K. \tag{8}$$

Assumption 4.1 makes a the restriction of a bounded function whose Fourier transform is supported in $[-2\pi f_{\max}, 2\pi f_{\max}]$; such a function is entire of exponential type $2\pi f_{\max}$, and Bernstein's inequality gives $\|a'\|_\infty \le 2\pi f_{\max} \|a\|_\infty$. Since a' is band-limited to the same interval, iterating K times yields

$$\left\| a^{(K)} \right\|_\infty \le (2\pi f_{\max})^K \|a\|_\infty . \tag{9}$$

Substituting equation 9 into equation 8,

$$\left| a_t - \hat{a}_t^{(K)} \right| \;\le\; (2\pi f_{\max})^K (\Delta t)^K \|a\|_\infty \;=\; \rho^K \|a\|_\infty ,$$

and taking the maximum over the six continuous components gives the claim. □

Remark E.2 (Why the rate is geometric, not factorial)**.** The $K!$ appearing in the divided-difference remainder is cancelled in equation 8 by the product of node spacings, $\prod_{j=1}^{K} j\Delta t = K!\,(\Delta t)^K$. A predictor given the exact derivatives $a^{(1)}(t - \Delta t), \dots, a^{(K-1)}(t - \Delta t)$ would instead achieve the Taylor rate $\rho^K/K!$; a policy conditioned on past *samples* has to estimate those derivatives by finite differences, and Lemma D.2 quantifies what that costs. The gap between the two rates is the price of working from samples alone.