FibVLA: AN EFFICIENT TEMPORAL VISION-LANGUAGE-ACTION MODEL WITH FIBONACCI SAMPLING

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-language-action models (VLAs), which leverage the cognition of multi-modal information to infer physical-world actions, provide a generalized solution for embodied AI applications. Conventional VLAs usually concentrate on current digital cognition. While some efforts are made to enhance VLAs' reasoning capabilities by capturing temporal information, encoding the long-context history causes an efficiency-decreasing issue. To reconcile the conflict between capturing temporal information and maintaining inference efficiency in VLAs, this paper introduces FibVLA, an efficient framework featuring temporal perception of long-context history. Specifically, we leverage logarithmic hindsight sampling to both proprioceptive states and visual frames to capture long-term temporal dependencies with minimal redundancy. For the action expert, we introduce the flow matching to produce action distributions, and the Fibonacci recurrent inference strategy to generate long-range planning steps based on real-time closed-loop feedback. Experiments demonstrate that FibVLA significantly improves action smoothness and success rates without retraining large-scale visual encoders. Efficiency analysis demonstrates superior real-time responsiveness compared to video-based baselines in real-world evaluations.

1 INTRODUCTION

Vision-Language-Action models (VLAs) serve as a foundational paradigm for embodied agents by grounding language instructions into executable robotic control tokens (Zitkovich et al., 2023; Yu et al., 2025). Conventional VLAs (Black et al., 2024; Liu et al., 2025; Kim et al., 2025b) primarily rely on limited traceback observations, which are insufficient for long-horizon embodied tasks in which current actions depend causally on historical context (Song et al.; Lin et al., 2025). Consequently, incorporating temporal context into VLA backbones is crucial to allow embodied agents, such as robotic manipulators, to achieve reliable performance in real-world executions.

Existing temporal modeling or inference in VLAs focuses on the reasoning capabilities of models given historical observations by prompting visual frames in VLM input context (Zheng et al., 2025; Mees et al., 2024; Li et al., 2024b; Cheang et al., 2024) or predicting future subgoals to guide action generation (Zhao et al., 2025; Tian et al., 2024; Zhang et al., 2025b). Although they utilize temporal information to enable VLAs to understand the reasoning process of action generation, the encoding of history frames and state traces has to face a new efficiency problem. It is the token explosion issue, which results in prohibitive computational costs that hinder the real-time control required for inference. Therefore, to balance efficiency and the utilization of historical context, recent research primarily falls into two categories: sparse representation learning (Zheng et al., 2025; Zhang et al., 2025a) and fixed-frequency sampling methods (Shi et al., 2025; Lin et al., 2025). Sparse-representation compression depends on offline preprocessing, limiting deployment flexibility in robotic manipulation. Fixed-frequency sampling also ignores the non-uniform information density of embodied tasks. For instance, tracking task progress requires a different sampling rate than capturing the robot's proprioceptive state.

To demonstrate the non-uniform information density inherent within embodied tasks, we investigate the latent characteristics of robotic manipulation data. As shown in Figure 1, our empirical analysis reveals that representations derived from high and low sampling frequencies form distinct clusters within the feature space. This phenomenon suggests that high-frequency frames are primarily

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.

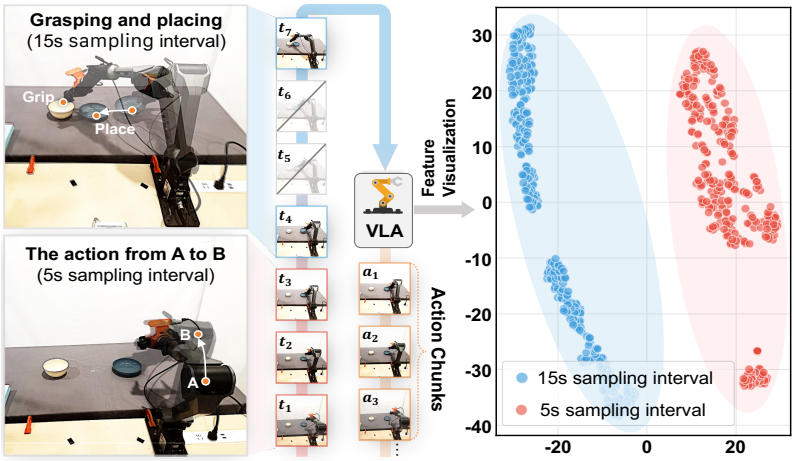


Figure 1: Action chunks generated by VLA, and visualization of features extracted by VLA at different sampling intervals.

associated with fine-grained motion control, such as changes in proprioceptive states, whereas low-frequency frames correspond to the understanding of task context, such as the progression of subtasks. Based on the above observation, we argue that integrating information from gradual-frequency sampling can enhance the VLA model's comprehensive temporal perception, enabling the robot to execute high-frequency actions while maintaining robust performance in long-horizon tasks.

Based on these insights, this work aims to design an effective and efficient temporal VLA framework for robotic perception and inference, where there are two main challenges that need to be addressed: First, *how to encode sampled robotic frames with repetitive context?* Despite gradual-frequency sampling, significant redundancy from static backgrounds and repeated features remains, requiring the model to extract key manipulation dynamics within a limited computational budget. Secondly, *how to reduce historical information maintenance overhead during inference?* Incorporating historical context at each step introduced additional retrieval and computation cost, resulting in latency that undermines the real-time requirements of robotic control.

To deal with these challenges, we propose an efficient temporal **V**ision-**L**anguage-**A**ction model with **Fib**onacci sampling, called `FibVLA`. Specifically, `FibVLA` introduces logarithmic hindsight sampling at the input stage to effectively capture both high-frequency actions and low-frequency proprioception states. Furthermore, during the encoding stage, we design a channel-wise temporal encoding that leverages temporal features along the channel dimension to filter out irrelevant background in the robot's historical frames, thereby reducing token noise. For the action expert, we introduce the flow matching method to generate action distributions rather than discriminative action sequences. Based on the action distributions, we design a Fibonacci recurrent inference strategy that exploits the recursive property of the Fibonacci sequence to align logarithmically sampled frames with preceding action chunks. Since the frames required for inference at the current timestamp are already included within the previous action chunk, avoiding redundant encoding of historical frames reduces inference overhead. Our contributions are summarized as follows:

- We propose `FibVLA`, a framework that concentrates on long-context perception for efficient temporal modeling in VLAs. It develops logarithmic hindsight sampling with Fibonacci sampling to mathematically enable the direct reuse of prior action chunks during inference.
- We introduce channel-wise temporal encoding methods to capture background-invariant historical frames and a Fibonacci recurrent inference strategy aligned with preceding action chunks, substantially reducing computational overhead in temporal VLA inference.
- We validate `FibVLA`'s superior performance on four simulation benchmarks. Moreover, we collect a real-world dataset with more than 600k frames across 15 tasks, including single/dual-arm and long/short-horizon robotic manipulation, demonstrating the effectiveness and efficiency of `FibVLA` in the physical world.

2 Related Work

To bridge the gap between the language inference of VLMs and the manipulation policies, vision-language action models (VLAs) (Yu et al., 2025; Shao et al., 2025; Ma et al., 2024) are explored to provide a direct path for processing information from vision, language, and action modalities. VLAs utilize visual foundation models (Dosovitskiy et al., 2021; Zhai et al., 2023; Radford et al., 2021) and LLMs to align vision and language embeddings, which are further trained on large-scale robot datasets (O'Neill et al., 2024; Walke et al., 2023; Khazatsky et al., 2024) to perform language-conditioned tasks. Recent works achieve success using either autoregressive discretization (e.g., RT-1 (Brohan et al., 2023), RT-2 (Zitkovich et al., 2023), and OpenVLA (Kim et al., 2025b)) or diffusion-based continuous denoising, such as π_0 (Black et al., 2024), CogACT (Li et al., 2024a), and HybridVLA (Liu et al., 2025). However, these methods often operate under the implicit assumption of the Markovian state (Jang et al., 2022; Haarnoja et al., 2018; Schulman et al., 2017), relying on instantaneous observations and overlooking the temporal dependencies arising from historical visual and proprioceptive states required for complex embodied tasks. For example, in tasks involving object occlusion, the scarcity of historical context prevents VLA models from recovering the last-known positions of target objects, thereby impeding successful task execution.

To address partial observability in long-context robotic manipulation, VLAs have evolved from basic pixel-stacking to integrated historical contexts (Li et al., 2023). While these advancements enhance the capacity of VLA models to capture temporal dependencies, they are increasingly bottlenecked by both the token explosion and the computational overhead associated with long-horizon sequences (Jiang et al., 2022; Dao et al., 2022). Although representative methods, such as TraceVLA (Zheng et al., 2025) and 4D-VLA (Zhang et al., 2025a), employ trajectory or 4D reconstructions for data reduction, their reliance on intensive offline preprocessing severely compromises real-time responsiveness and deployment flexibility. Furthermore, existing memory-based (Shi et al., 2025) and predictive strategies (Lin et al., 2025) utilize fixed-frequency sampling, overlooking the temporal non-uniformity of robotic manipulation. In practice, critical signals are often sparse and concentrated within irregular temporal windows (James et al., 2022; Mandlekar et al., 2022), yet current models expend resources on redundant frames, thereby risking the loss of pivotal transient information (Feichtenhofer et al., 2019). Thus, adaptively extracting essential historical information while maintaining inference efficiency and accuracy remains a fundamental issue in VLA temporal modeling.

3 Methodology

3.1 Problem Formulation

We formulate the robotic manipulation task as a sequential decision-making process. At current timestamp t, the system receives a natural language instruction $\mathcal{L}$, a current visual observation $o_t \in \mathbb{R}^{H\times W\times 3}$ with the resolution of $H \times W$, and a proprioceptive state $s_t \in \mathbb{R}^{d_s}$. The objective is to predict a future action chunk $\mathcal{A}_{t:t+L} = \{a_t, a_{t+1}, \ldots, a_{t+L}\}$, where L denotes the action chunk length and each action $a_t \in \mathbb{R}^{d_a}$. The policy π relies solely on the instantaneous observation, modeled as:

$$\mathcal{A}_{t:t+L} \sim \pi(\cdot \mid \mathcal{L}, o_t, s_t) \tag{1}$$

To capture temporal dynamics, a standard approach extends this by conditioning on a continuous history window. Let T represent the lookback window size. We define the continuous visual history as $\mathcal{O}_T = \{o_t, o_{t-1}, \ldots, o_{t-T}\}$ (where $t > T$) and the corresponding state history as $\mathcal{S}_T$. The policy is further formalized as:

$$\mathcal{A}_{t:t+L} \sim \pi(\cdot \mid \mathcal{L}, \mathcal{O}_T, \mathcal{S}_T) \tag{2}$$

3.2 Architecture Overview

In this section, we propose the `FibVLA` framework based on flow matching generation to achieve efficient temporal modeling. At the input stage of `FibVLA`, we introduce a monotonically increasing sparse temporal index set $\mathcal{K} = \{k_1, \ldots, k_N\}$, containing N sampled frames, which is used to construct the observation set $\mathcal{O}_\mathcal{K}$ and the corresponding state set $\mathcal{S}_\mathcal{K}$. Our proposed `FibVLA`, parameterized by θ, generates the action chunk by conditioning on these non-uniformly sampled frames as $\mathcal{A}_{t:t+L} = \pi_\theta(\cdot \mid \mathcal{L}, \mathcal{O}_\mathcal{K}, \mathcal{S}_\mathcal{K})$. The overall architecture of the `FibVLA` framework is shown

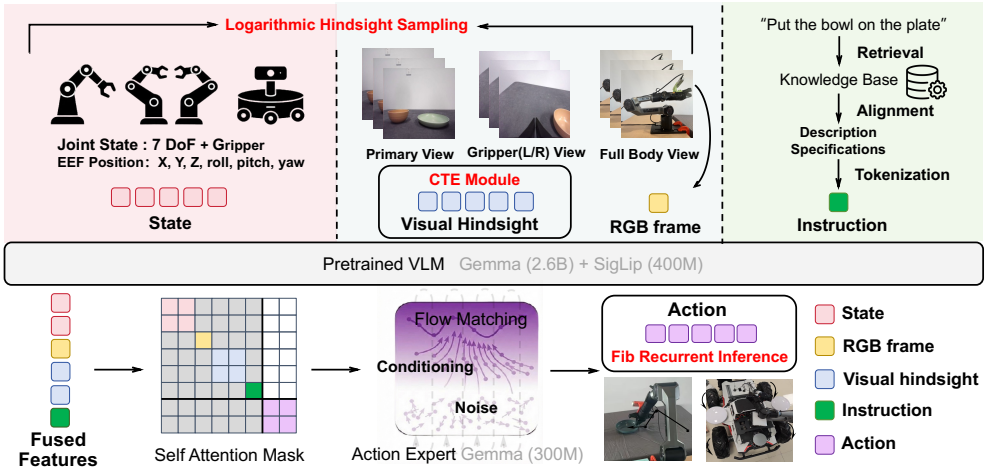


Figure 2: The overview of FibVLA, consisting of Logarithmic Hindsight Sampling, Channel-wise Temporal Encoding Module, and Fibonacci Recurrent Inference.

in Figure 2. This framework unifies the processing of multimodal inputs, including language instructions, multi-view visual frames, and state information. All inputs are first projected into a unified token sequence and then fed into the PaliGemma backbone (Beyer et al., 2024). Specifically, the input first leverages logarithmic hindsight sampling to capture temporal dependencies in both visual frames and state, while the sampled visual frames are further processed by a channel-wise temporal encoding (CTE) Module. Then, the encoded multimodal features, guided by a customized prefix attention mask, are employed to generate the action distributions through a flow-matching-based action expert. Finally, the robotic action is precisely generated using the Fibonacci recurrent inference strategy.

3.3 Logarithmic Hindsight Sampling

Based on the analysis in Figure 1, the VLA model enhances full-scale temporal perception in embodied tasks by integrating gradual-frequency sampling while balancing computational overhead. A common approach is Long-Short Term Hybrid Sampling (Wang et al., 2016; Lin et al., 2019), which maintains multiple temporal buffers to densely sample distant past states with a fixed stride. Given that distant states change slowly and provide limited information gain, this method is inefficient in allocating the sampling budget. Ideally, the sampling density should decay naturally with temporal distance, following a dense-near, sparse-far pattern. Probabilistic Quantile Sampling (Li et al., 2019; Zhou et al., 2021) partially follows this intuition, but its sampling points do not align with discrete time steps and rely on distributional assumptions and quantile computations, leading to inconsistencies between training and inference.

To address the above issue, we propose Logarithmic Hindsight Sampling, which constructs a discrete set of temporal indices $\mathcal{K}$ on a logarithmic scale. This approach preserves high-frequency information for recent time steps while efficiently covering long-term history. Formally, given a minimum sampling interval q_{min} and a growth rate $r > 1$, the sampling points are defined as:

$$k_i = \lfloor q_{min} \cdot r^i \rfloor \tag{3}$$

To mitigate index collisions caused by discretization artifacts (i.e., the floor operation), we impose a recursive sparsity constraint on the sequence:

$$k_i \geq k_{i-1} + k_{i-2}, \quad \forall i > 2 \tag{4}$$

This constraint implicitly establishes a theoretical lower bound for the growth rate r, ensuring that the sampling sequence remains strictly monotonic after discretization and inherently eliminating redundancy. In practice, the growth rate can be flexibly adjusted according to the control frequency to balance long-term coverage with the precision required to capture critical state transitions.

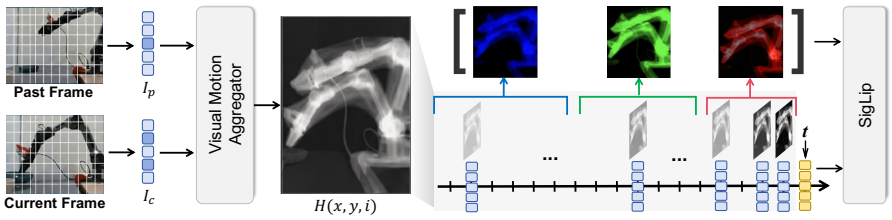


Figure 3: Illustration of the channel-wise temporal encoding module. Specifically, $H(x, y, i)$ leverages spatial discrepancies between current and past frames to identify movement, Historical frames are then integrated into the composite input via channel-wise mapping.

3.4 Channel-wise Temporal Encoding

While optimized sampling expands the temporal receptive field, scaling it to high-dimensional visual inputs remains challenging: long-horizon visual frames are often dominated by redundant backgrounds that mask critical task dynamics (Tong et al., 2022), and Transformer self-attention incurs a computational overhead that scales quadratically with sequence length (Dao et al., 2022). To overcome these issues, we introduce a channel-wise temporal encoding module to compress discrete temporal dynamics into a compact representation. Specifically, the CTE module encodes temporal dynamics by using the visual motion aggregator to process sampled sparse frames $\mathcal{O}_{\mathcal{K}}$ based on the temporal index set $\mathcal{K}$. First, the motion difference $D(x, y, i)$ for each $i \in [1, N-1]$ is derived via frame differencing on the sampled visual frames $I(x, y, t)$:

$$D(\cdot, i) = |I(\cdot, t - k_i) - I(\cdot, t - k_{i+1})| \tag{5}$$

Next, a binary motion mask $\Psi(x, y, i)$ is obtained using a predefined threshold ξ:

$$\Psi(\cdot, i) = \begin{cases} 1, & \text{if } D(\cdot, i) > \xi \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

Then, the temporal encoding frame $H(x, y, i)$ is generated through the following recursive logic:

$$H(\cdot, i) = \begin{cases} \tau, & \text{if } \Psi(\cdot, i) = 1 \\ \max(0, H(\cdot, i+1) - \delta), & \text{otherwise} \end{cases} \tag{7}$$

where τ denotes the maximum intensity duration, and δ is a decay parameter. This recursive process proceeds in the direction of decreasing temporal lag. The final output, $H(x, y, 1)$, serves as the processed motion history image (Ahad, 2012). In this representation, brighter pixels correspond to recent motion, while darker pixels retain earlier motion traces, thereby forming a "visual trail" that provides explicit temporal cues.

To fully leverage multi-scale temporal features obtained through gradual-frequency sampling, the CTE module divides the visual history into three temporal ranges: Near, Mid, and Far. As shown in Figure 3, features from these ranges are respectively mapped to the R, G, and B channels of the PaliGemma visual encoder (SigLip), forming the hindsight feature $\hat{o}_t$ (see Appendix I for examples). Finally, the current RGB frame o_t is retained as a semantic anchor and combined with $\hat{o}_t$ as input to the visual encoder, thereby integrating the embodied task's temporal information with the scene's color and texture details.

3.5 Fibonacci Recurrent Inference

During the VLA inference stage, the sampled $\mathcal{K}$ based on Eq.(4) causes misalignment between the frames used to infer the next action chunk and the historical frames when $k_i > k_{i-1} + k_{i-2}$, as shown in Figure 4. This misalignment prevents the direct reuse of historical feature tokens, forcing the model to re-encode temporal information at every step and thereby limiting inference speed. In contrast, when $k_i = k_{i-1} + k_{i-2}$, the sampling strategy perfectly aligns with the additive recursive property inherent to the Fibonacci sequence, allowing historical feature tokens to be reused during the inference of the next action chunk. Thus, we propose a Fibonacci recurrent inference strategy, leveraging the inherent additive recursive property of the Fibonacci sequence. We prove that this

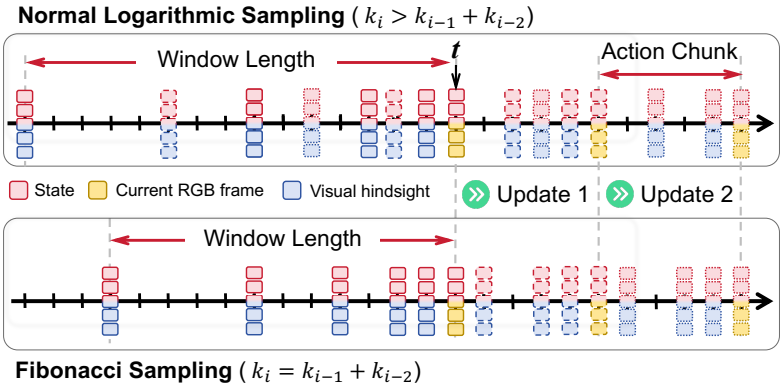


Figure 4: Schematic of Fibonacci Recurrent Inference.

strategy provides the unique analytical solution for maximizing historical information reuse under sparse sampling constraints (proof detailed in Appendix H).

Let the system receive visual input $\mathcal{O}_{\mathcal{K}}^{t}$ at time t and execute an action sequence of length L, advancing the system time to $t' = t + L$. Utilizing the Fibonacci property $k_i = k_{i-1} + k_{i-2}$, and setting the action update step to $L = k_{i-2}$, the historical frames at time t and the sampling points at time t' satisfy the following equality:

$$(t + k_{i-2}) - k_i = t - (k_i - k_{i-2}) = t - k_{i-1} \tag{8}$$

Eq.(8) indicates that by setting the update step (i.e., action chunk length) to $L = k_{i-2}$, the historical frame located k_{i-1} steps before time t corresponds to the frame located k_i before time t'. Furthermore, for sampling point k_{i-3} required at t', its effective timestamp satisfies $t' - k_{i-3} = t + (k_{i-2} - k_{i-3}) > t$, placing it strictly after the previous sampling time t. This confirms that all high-frequency updates occur strictly within the newly elapsed window $[t, t + L]$, thereby preserving the integrity of the historical structure without any misaligned sampling points.

Based on this derivation, we dynamically bind the action chunk length to specific Fibonacci terms, enabling the feature tokens in the KV cache to be precisely aligned with the sampling points at the next time step. As the sampling depth increases, the ratio between adjacent sampling intervals converges to the golden ratio, naturally maintaining a logarithmic sampling distribution.

4 Experiment

4.1 Experiment Setups

We have conducted a comprehensive series of experiments to evaluate the effectiveness of `FibVLA`, including: 1) Three simulated benchmark tests to measure its manipulation performance across diverse tasks. 2) Robotic deployment experiments in real-world environments. 3) Ablation studies to verify the contribution of each module. 4) Efficiency analysis to assess the model's inference performance.

Benchmarks & Datasets. We employ comprehensive benchmarks to evaluate task generalization and long-horizon manipulation capabilities. 1) Diverse simulation benchmarks, including **LIBERO** (Liu et al., 2023) and **MIKASA-Robo** (Cherepanov et al., 2025), alongside large-scale policy evaluations on the **Bridge** (Walke et al., 2023) and **Fractal** (Brohan et al., 2023) via SimplerEnv (Li et al., 2025). 2) **Real-world dataset**, which is collected from a physical platform built on the Piper robotic arm, comprising over 600k frames of data across 15 tasks, including single/dual-arm and long/short-horizon robotic manipulation. Further details on the specific configurations for these benchmarks and datasets are available in Appendix A and Appendix E.

Baselines. To verify the effectiveness of `FibVLA`, we compare it against a range of representative state-of-the-art VLA models. These baselines are primarily categorized into three types: 1) Classic VLM-based policies, including **RT-1-X** (Brohan et al., 2023), **RT-2-X** (Zitkovich et al., 2023), **OpenVLA** (Kim et al., 2025b), and **OpenVLA-OFT** (Kim et al., 2025a). 2) Generative policies

Table 1: Comparison of success rates on LIBERO. The best results are highlighted in bold, and the second-best results are underlined.

Method	Avg. SR	Spatial	Object	Goal	Long
Octo	75.1	78.9	85.7	84.6	51.1
π_0	<u>94.2</u>	96.8	**98.8**	<u>95.8</u>	85.2
OpenVLA	76.5	84.7	88.4	79.2	53.7
CogACT	93.5	<u>97.2</u>	<u>98.0</u>	90.2	<u>88.8</u>
TraceVLA	74.8	84.6	85.2	75.1	54.1
SpatialVLA	78.1	88.2	89.9	78.6	55.5
4D-VLA	88.6	88.9	95.2	90.9	79.1
FibVLA	**96.8**	**97.8**	<u>98.0</u>	**96.4**	**95.2**

Table 2: Comparison of success rates on the SimplerEnv-Bridge.

Method	Avg. SR	Task 1	Task 2	Task 3	Task 4
RT-1-X	1.1	0.0	4.2	0.0	0.0
OpenVLA	4.2	4.2	0.0	0.0	12.5
Octo	17.5	15.8	12.5	0.0	41.7
π_0	<u>55.7</u>	63.3	<u>58.8</u>	21.3	79.2
CogACT	51.3	<u>71.7</u>	50.8	15.0	67.5
TraceVLA	27.7	12.5	16.6	16.6	65.0
SpatialVLA	42.7	16.7	25.0	<u>29.2</u>	**100**
FibVLA	**67.3**	**75.0**	**65.4**	**35.2**	<u>93.7</u>

Table 3: Comparison of success rates on the SimplerEnv-Fractal, reporting success rates across Visual Matching (VM) and Visual Aggregation (VA) settings. Detailed subtask names for this and subsequent evaluations are provided in Appendix C.

Method	Visual Matching (VM)					Visual Aggregation (VA)					Overall
	Avg. SR	Task1	Task2	Task3	Task4	Avg. SR	Task1	Task2	Task3	Task4	
RT-1-X	42.4	56.7	31.7	59.7	21.3	30.2	49.0	32.3	29.4	10.1	36.3
RT-2-X	46.3	78.7	77.9	25.0	3.7	54.4	<u>82.3</u>	79.2	35.5	20.6	50.4
OpenVLA	34.3	18.0	56.3	63.0	0.0	39.3	60.8	67.7	28.8	0.0	36.8
Octo	11.0	17.0	4.2	22.7	0.0	1.2	0.6	3.1	1.1	0.0	6.1
π_0	69.1	88.0	80.3	56.0	**52.2**	–	–	–	–	–	–
CogACT	<u>74.8</u>	**91.3**	85.0	<u>71.8</u>	<u>50.9</u>	<u>61.3</u>	**89.6**	<u>80.8</u>	28.3	<u>46.6</u>	<u>68.1</u>
TraceVLA	45.8	45.0	63.8	63.1	11.1	49.8	64.3	60.6	**61.6**	12.5	47.8
SpatialVLA	56.0	79.3	**90.0**	54.6	0.0	51.8	78.7	**83.0**	39.2	6.3	53.9
FibVLA	**78.6**	<u>89.0</u>	<u>86.7</u>	**86.5**	**52.2**	**65.5**	82.1	77.9	<u>43.6</u>	**58.3**	**72.1**

based on diffusion or flow matching, such as **Octo** (Ghosh et al., 2024) and π_0 (Black et al., 2024). 3) VLA variants focusing on enhanced spatio-temporal reasoning and cognitive planning, encompassing **CogACT** (Li et al., 2024a), **TraceVLA** (Zheng et al., 2025), **SpatialVLA** (Qu et al., 2025), **4D-VLA** (Qu et al., 2025), and **HiF-VLA** (Lin et al., 2025). For detailed descriptions of all baselines and the implementation details of FibVLA, please refer to Appendix B and Appendix F, respectively.

4.2 Simulation Evaluation

Evaluation on LIBERO. As shown in Table 1, FibVLA demonstrates competitive performance across all benchmarks. Notably, it achieves a significant improvement on the LIBERO-Long suite, outperforming the second-best baseline by 7.21% and reaching a success rate of 95.2%. This improvement is primarily attributed to FibVLA's gradual-frequency sampling, which effectively bridges the gap between capturing global task goals and local motion dynamics. (Detailed success rates for individual subtasks are provided in Appendix D.1). Beyond long-horizon scenarios, FibVLA also maintains high stability in basic manipulation tasks involving spatial, object, and goal-oriented operations. Overall, FibVLA achieves an average success rate of 96.8%, demonstrating its effectiveness as a general-purpose manipulation policy in handling both complex reasoning and basic execution tasks.

Evaluation on SimplerEnv-Bridge. In the SimplerEnv-Bridge setting, we evaluate the model's manipulation capabilities on the WidowX robot, covering tasks ranging from basic pick-and-place to complex long-horizon manipulation. As shown in Table 2, FibVLA demonstrates superior performance on the Bridge benchmark, achieving an average success rate of 67.3%. Compared to CogACT and the recent SOTA model π_0, FibVLA achieves significant performance gains of 3.12% and 20.9%, respectively. These results indicate that FibVLA's temporal module effectively models dynamic object interactions, substantially enhancing the overall execution accuracy of policies in Bridge tasks.

Evaluation on SimplerEnv-Fractal. To validate the model's robustness against visual perturbations, we conduct a comprehensive evaluation on SimplerEnv-Fractal using two core protocols: Visual Matching, which aims to minimize the visual appearance gap between simulation and real world to

Table 4: Comparison of success rates on the MIKASA-Robo.

Method	Avg. SR	Task1	Task2	Task3	Task4
π_0	<u>33.0</u>	33.0	**42.0**	<u>31.0</u>	<u>26.0</u>
SpatialVLA	22.0	23.0	27.0	18.0	20.0
OpenVLA-OFT	26.5	<u>48.0</u>	14.0	27.0	17.0
FibVLA	**46.5**	**78.0**	<u>37.0</u>	**36.0**	**35.0**

assess transfer potential; and Visual Aggregation, serving as a stress test by introducing extensive environmental variations to evaluate adaptability to visual domain shifts. Table 3 reports the overall performance: FibVLA achieves a 5.87% improvement in average success rate compared to the second-best baseline, CogACT. Notably, under the rigorous Visual Aggregation setting, despite severe environmental changes, FibVLA maintains a stable average success rate of 65.5% across all four subtasks without any significant weaknesses. This finding indicates that FibVLA goes beyond mere memorization of training data patterns. Instead, it leverages temporal context to comprehend task semantics, thereby maintaining stable control amid lighting variations and background distractions, effectively mitigating the action jitter or execution failures often observed in baseline models lacking temporal modeling.

Evaluation on MIKASA-Robo. In the MIKASA-Robo benchmark, the environment only briefly presents the target configuration in the initial few frames as a cue, after which the scene resets to a default state. Consequently, the model must rely solely on the memory of the initial keyframes to reconstruct the target scene, as the target information is unavailable in the current observations. As shown in Table 4, FibVLA achieves superior performance with an average success rate of 46.5%, outperforming π_0 by 40.9%. Notably, the model demonstrates a significant performance advantage in specific subtasks that involve transient visual goals. This improvement is likely attributed to the design of the temporal module. In scenarios where target positions or task-relevant cues are only briefly visible at the onset, the gradual-frequency sampling mechanism prioritizes the retention of these critical initial states.

4.3 Real-world Evaluation

We have conducted extensive evaluations on the Piper robotic platform, equipped with both third-person and wrist-mounted cameras, to validate the effectiveness and robustness of FibVLA in real-world settings. The model is fine-tuned on teleoperated demonstrations covering single/dual-arm coordination across various horizons. To move beyond binary success rates, we introduce a fine-grained step-wise scoring metric to evaluate manipulation quality and execution stability. The detailed definitions and task decompositions are provided in Appendix E.

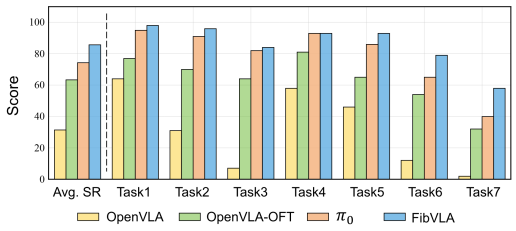


Figure 5: Real-world performance comparison across individual tasks. Detailed subtask names are provided in Appendix C.

We compare FibVLA against mainstream baselines, including OpenVLA, OpenVLA-OFT, and π_0. As shown in Figure 5, FibVLA demonstrates superior performance across all tasks, achieving an average score of 85.7, which significantly outperforms the second-best performing model π_0 (by a margin of +11.4 points). Specifically, FibVLA not only exhibits exceptional stability and fluency with near-perfect scores on the first five basic tasks, but also establishes a substantial lead on the more challenging long-horizon tasks (Task 6 and Task 7). Further details regarding the physical experimental setup and specific evaluation cases are described in Appendix G.

4.4 Ablation Study

To validate the effectiveness of FibVLA's core components, we conduct two variants: 1) **w/o Sampling**, which removes the progressive frequency sampling strategy from FibVLA. 2) **w/o CTE**, which removes the Channel-wise Temporal Encoding module. Ablation experiments have conducted on the LIBERO-Long benchmark and the real-world dataset. The experimental results are shown in Table 5. Specifically, FibVLA significantly outperforms all variants. The w/o Sampling variant suffers the most severe performance degradation. This demonstrates that an efficient historical information compression mechanism is crucial for capturing long-horizon dependencies. Furthermore, w/o CTE also causes a noticeable performance loss, validating the key role in temporal feature extraction and fusion.

Table 5: Ablation study of FibVLA components on LIBERO-Long and Real-world dataset.

Variant	LIBERO-Long (Avg. SR)	Real-world (Avg. Score)
FibVLA	**95.2**	**85.7**
–(w/o Sampling)	88.4	78.3
–(w/o CTE)	91.2	80.0

4.5 Efficiency Studiy

To evaluate the inference efficiency of different sampling strategies, we compare the inference latency of our Fibonacci recurrent inference strategy against other common mathematical sampling methods, including normal Logarithmic sampling (**w/ Logarithmic**) and long-short term hybrid sampling (**w/ Long–Short**). We also included recent state-of-the-art temporal VLA methods, TraceVLA and HiF-VLA, both of which employ traditional continuous time-window sampling mechanisms. For fair comparison, all methods have configured with a uniform historical time window size of 10 frames. As shown in Table 6, the results on the LIBERO-Long task suite demonstrate the superior computational efficiency of Fibonacci sampling. While maintaining high-level performance across these long-horizon tasks, our method requires only 177 ms per inference, indicating that the Fibonacci recurrent inference maximizes historical information coverage with minimal temporal overhead while reducing computational latency.

Table 6: Comparison of inference time across different sampling strategies, including both competitive baselines and FibVLA variants utilizing alternative sampling curves.

Sampling Method	Inference Time (ms)	Avg. SR (LIBERO-Long)
FibVLA	**177**	<u>95.2</u>
–(w/ Logarithmic)	201	94.6
–(w/ Long–Short)	235	**95.5**
TraceVLA	<u>196</u>	54.1
HiF-VLA	243	94.4

5 Conclusion

In this paper, we propose FibVLA, an efficient temporal VLA framework designed to bridge the gap between temporal perception and real-time inference. We design a novel gradual-frequency sampling strategy based on Fibonacci sampling, which allows the model to directly reuse feature caches from the previous timestep during inference without incurring additional temporal encoding overhead, thereby capturing temporal context information for embodied tasks. Experimental results on three benchmarks and the real-world dataset show that FibVLA significantly improves performance in long-horizon tasks and demonstrates real-time robustness in the real world. In future work, we plan to leverage the Fibonacci recursive principle to explore replay buffer mechanisms, enhancing FibVLA's continual learning for embodied tasks under out-of-distribution scenarios.

AI USE STATEMENT

In this draft, generative AI tools were used to assist with LaTeX template migration and formatting. The authors will update this disclosure before submission to reflect the complete and final use of AI-assisted tools. The authors take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This paper advances the foundations of Embodied AI. By improving the efficiency of vision-language-action models, this work reduces the computational overhead for robotic training and inference, mitigating the environmental footprint for large-scale deployment. Furthermore, these efficiency gains facilitate the democratization of advanced robotic control on resource-constrained platforms. As fundamental research, its specific societal consequences primarily depend on downstream applications, and we foresee no specific negative impacts.

REPRODUCIBILITY STATEMENT

We include detailed benchmark descriptions, baseline definitions, task specifications, hyperparameters, real-world setup details, and the theoretical proof in the appendix. These materials are intended to support reproducibility of the simulation and real-world evaluations.

REFERENCES

Md Atiqur Rahman Ahad. Motion history image: its variants and applications. *Machine Vision and Applications*, 23(2):255–281, 2012.

Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *Proceedings of the International Conference on Machine Learning (ICML)*, July 2021.

Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, et al. Paligemma: A versatile 3b vlm for transfer. *arXiv preprint arXiv:2407.07726*, 2024.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. pi_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alexander Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *Robotics: Science and Systems XIX*, 2023.

Chi-Lam Cheang, Guangzeng Chen, Ya Jing, Tao Kong, Hang Li, Yifeng Li, Yuxiao Liu, Hongtao Wu, Jiafeng Xu, Yichu Yang, et al. Gr-2: A generative video-language-action model with web-scale knowledge for robot manipulation. *arXiv preprint arXiv:2410.06158*, 2024.

Egor Cherepanov, Nikita Kachaev, Alexey K Kovalev, and Aleksandr I Panov. Memory, benchmark & robots: A benchmark for solving complex tasks with reinforcement learning. *arXiv preprint arXiv:2502.10550*, 2025.

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in neural information processing systems*, 35: 16344–16359, 2022.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6202–6211, 2019.

Dibya Ghosh, Homer Rich Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, et al. Octo: An open-source generalist robot policy. In *Robotics: Science and Systems*, 2024.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pp. 1861–1870. Pmlr, 2018.

Stephen James, Kentaro Wada, Tristan Laidlow, and Andrew J Davison. Coarse-to-fine q-attention: Efficient learning for visual robotic manipulation via discretisation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13739–13748, 2022.

Eric Jang, Alex Irpan, Mohi Khansari, Daniel Kappler, Frederik Ebert, Corey Lynch, Sergey Levine, and Chelsea Finn. Bc-z: Zero-shot task generalization with robotic imitation learning. In *Conference on Robot Learning*, pp. 991–1002. PMLR, 2022.

Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. Vima: General robot manipulation with multimodal prompts. *arXiv preprint arXiv:2210.03094*, 2022.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. In *Robotics: Science and Systems*, 2024.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. *arXiv preprint arXiv:2502.19645*, 2025a.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, et al. Openvla: An open-source vision-language-action model. In *Conference on Robot Learning*, pp. 2679–2713. PMLR, 2025b.

Qixiu Li, Yaobo Liang, Zeyu Wang, Lin Luo, Xi Chen, Mozheng Liao, Fangyun Wei, Yu Deng, Sicheng Xu, Yizhong Zhang, et al. Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. *arXiv preprint arXiv:2411.19650*, 2024a.

Shiyang Li, Xiaoyong Jin, Yao Xuan, Xiyou Zhou, Wenhu Chen, Yu-Xiang Wang, and Xifeng Yan. Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. *Advances in neural information processing systems*, 32, 2019.

Xinghang Li, Minghuan Liu, Hanbo Zhang, Cunjun Yu, Jie Xu, Hongtao Wu, Chilam Cheang, Ya Jing, Weinan Zhang, Huaping Liu, et al. Vision-language foundation models as effective robot imitators. *arXiv preprint arXiv:2311.01378*, 2023.

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*, 2024b.

Xuanlin Li, Kyle Hsu, Jiayuan Gu, Oier Mees, Karl Pertsch, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, et al. Evaluating real-world robot manipulation policies in simulation. In *Conference on Robot Learning*, pp. 3705–3728. PMLR, 2025.

Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 7083–7093, 2019.

Minghui Lin, Pengxiang Ding, Shu Wang, Zifeng Zhuang, Yang Liu, Xinyang Tong, Wenxuan Song, Shangke Lyu, Siteng Huang, and Donglin Wang. Hif-vla: Hindsight, insight and foresight through motion representation for vision-language-action models. *arXiv preprint arXiv:2512.09928*, 2025.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *arXiv preprint arXiv:2306.03310*, 2023.

Jiaming Liu, Hao Chen, Pengju An, Zhuoyang Liu, Renrui Zhang, Chenyang Gu, Xiaoqi Li, Ziyu Guo, Sixiang Chen, Mengzhen Liu, et al. Hybridvla: Collaborative diffusion and autoregression in a unified vision-language-action model. *arXiv preprint arXiv:2503.10631*, 2025.

Yueen Ma, Zixing Song, Yuzheng Zhuang, Jianye Hao, and Irwin King. A survey on vision-language-action models for embodied ai. *arXiv preprint arXiv:2405.14093*, 2024.

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 *Conference on Robot Learning*, pp. 1678–1690. PMLR, 2022.

Oier Mees, Dibya Ghosh, Karl Pertsch, Kevin Black, Homer Rich Walke, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, et al. Octo: An open-source generalist robot policy. In *First Workshop on Vision-Language Models for Navigation and Manipulation at ICRA 2024*, 2024.

Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6892–6903. IEEE, 2024.

Delin Qu, Haoming Song, Qizhi Chen, Yuanqi Yao, Xinyi Ye, Yan Ding, Zhigang Wang, JiaYuan Gu, Bin Zhao, Dong Wang, et al. Spatialvla: Exploring spatial representations for visual-language-action model. *CoRR*, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Rui Shao, Wei Li, Lingsen Zhang, Renshan Zhang, Zhiyang Liu, Ran Chen, and Liqiang Nie. Large vlm-based vision-language-action models for robotic manipulation: A survey. *arXiv preprint arXiv:2508.13073*, 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.

Kiwhan Song, Boyuan Chen, Max Simchowitz, Yilun Du, Russ Tedrake, and Vincent Sitzmann. History-guided video diffusion. In *Forty-second International Conference on Machine Learning*.

Yang Tian, Sizhe Yang, Jia Zeng, Ping Wang, Dahua Lin, Hao Dong, and Jiangmiao Pang. Predictive inverse dynamics models are scalable learners for robotic manipulation. *arXiv preprint arXiv:2412.15109*, 2024.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in neural information processing systems*, 35:10078–10093, 2022.

Homer Rich Walke, Kevin Black, Tony Z Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning*, pp. 1723–1736. PMLR, 2023.

Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pp. 20–36. Springer, 2016.

Zhaoshu Yu, Bo Wang, Pengpeng Zeng, Haonan Zhang, Ji Zhang, Lianli Gao, Jingkuan Song, Nicu Sebe, and Heng Tao Shen. A survey on efficient vision-language-action models. *arXiv preprint arXiv:2510.24795*, 2025.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11975–11986, 2023.

He Zhang, Lingzhu Xiang, Haitao Lin, Zeyu Huang, Minghui Wang, Dingyan Zhong, Yubo Dong, Yihao Wu, Yongming Rao, Dongsheng Zhang, et al. Hy-embodied-0.5-vla: From vision-language-action models to a real-world robot learning stack. *arXiv preprint arXiv:2606.14409*, 2026.

Jiahui Zhang, Yurui Chen, Yueming Xu, Ze Huang, Yanpeng Zhou, Yu-Jie Yuan, Xinyue Cai, Guowei Huang, Xingyue Quan, Hang Xu, et al. 4d-vla: Spatiotemporal vision-language-action pretraining with cross-scene calibration. *arXiv preprint arXiv:2506.22242*, 2025a.

Jianke Zhang, Yanjiang Guo, Yucheng Hu, Xiaoyu Chen, Xiang Zhu, and Jianyu Chen. Up-vla: A unified understanding and prediction model for embodied agent. *arXiv preprint arXiv:2501.18867*, 2025b.

Qingqing Zhao, Yao Lu, Moo Jin Kim, Zipeng Fu, Zhuoyang Zhang, Yecheng Wu, Zhaoshuo Li, Qianli Ma, Song Han, Chelsea Finn, et al. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 1702–1713, 2025.

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. In *The Thirteenth International Conference on Learning Representations*, 2025.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 11106–11115, 2021.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pp. 2165–2183. PMLR, 2023.

A Detailed Experimental Setups and Benchmarks

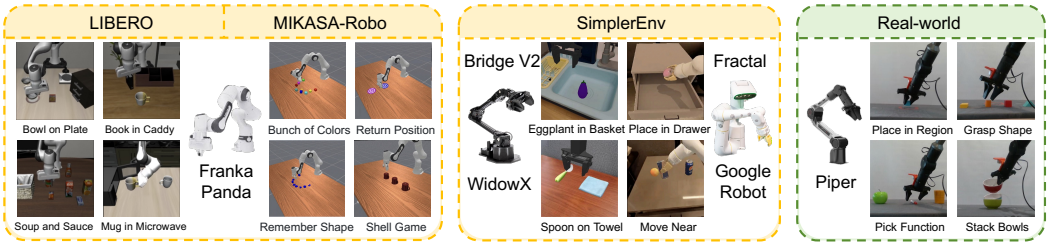


Figure 6: Overview of experimental setups and task specifications. The evaluation encompasses three simulation benchmarks and real-world robotic tasks across diverse platforms, including the **Franka Panda** (LIBERO-Long and MIKASA-Robo), **WidowX 250** (SimplerEnv-Bridge), and **Google Robot** (SimplerEnv-Fractal), and the **Piper** manipulator (Real-world evaluation).

- **LIBERO Liu et al. (2023):** A lifelong robotic learning benchmark designed to decouple knowledge transfer. Here, LIBERO-Spatial, Object, and Goal (10 tasks each) investigate the transfer of spatial relationships, object types, and motion behaviors, respectively, while LIBERO-Long comprises 10 challenging long-horizon tasks for evaluating downstream performance in complex, multi-step scenarios.

- **MIKASA-Robo Cherepanov et al. (2025):** A comprehensive benchmark comprising 32 memory-intensive manipulation tasks across 12 categories, designed to evaluate four core memory capabilities: Object Memory, Spatial Memory, Sequential Memory, and Memory Capacity. By targeting these distinct cognitive aspects, this benchmark provides a rigorous environment for assessing a model's generalization in handling complex geometries and contact-rich interactions.
- **SimplerEnv Li et al. (2025):** An open-source simulation suite for evaluating real-to-sim transfer, encompassing robotic setups from Google Robot (RT-series) and BridgeData V2. It offers a standardized Gym interface for seamless task interaction and provides integrated inference pipelines for evaluating generalist policies like RT-1 and Octo.
  - **Bridge V2 Walke et al. (2023):** A large-scale real-world dataset comprising over 60k trajectories across diverse environments using the WidowX 250 manipulator. As a key component of the Open X-Embodiment cross-embodiment dataset, Bridge V2 serves as a primary training source for verifying a model's effectiveness in open-vocabulary instruction following, with performance further validated in the aligned SimplerEnv.
  - **Fractal Brohan et al. (2023):** A large-scale real-world robotics dataset released alongside the RT-1 model, consisting of approximately 130k episodes and over 700 tasks collected by a fleet of 13 robots over 17 months. Characterized by its high diversity in objects, backgrounds, and manipulation skills, Fractal serves as a foundational data source for pre-training generalist VLA models and is a core component of the Open X-Embodiment dataset.
- **Real-world:** The real-world benchmarking is conducted on a curated custom dataset comprising over 600k frames across 15 tasks, encompassing both single- and dual-arm manipulations with varying temporal horizons. To prioritize experimental reproducibility, a standardized environment is established featuring a consistent white background and black tablecloth, with interaction objects limited to easily accessible props such as various cubes and common household items. A subset of 7 representative tasks was selected from this suite for comprehensive performance evaluation.

B Extended Description of Baselines

- **RT-1-X Brohan et al. (2023) & RT-2-X Zitkovich et al. (2023): RT-1-X** is a transformer-based robot action model that uses EfficientNet as a vision encoder to output discretized action tokens. **RT-2-X** is a representative VLA model that transforms a Vision-Language Model (VLM) into a closed-loop robotic policy through co-fine-tuning on large-scale internet data and robotic trajectories.
- **OpenVLA Kim et al. (2025b) & OpenVLA-OFT Kim et al. (2025a): OpenVLA** is a Llama 2-based model using Prismatic encoders to generate actions via discrete token quantization. Its successor, **OpenVLA-OFT**, optimizes this framework by adopting a continuous action head with L_1 regression and action chunking. This transition from autoregressive tokenization to direct regression enables higher control frequencies and improved inference efficiency while maintaining the original model's strong generalization.
- **Octo Ghosh et al. (2024):** A transformer-based generalist robot policy pre-trained on the Open X-Embodiment dataset. Unlike the RT series, Octo employs a diffusion head to output multi-modal action distributions rather than discrete tokens.
- π_0 **Black et al. (2024):** A foundation model introduced by Physical Intelligence. Similar to `FibVLA`, it leverages flow matching techniques to model continuous action distributions. As a powerful generalist policy, it represents the frontier of generative action modeling.
- **CogACT Li et al. (2024a):** A VLA model designed to distinguish "cognition" from "action." By decoupling Chain-of-Thought (CoT) reasoning from specific action execution, it enhances task planning and execution capabilities for complex long-horizon tasks.
- **TraceVLA Zheng et al. (2025):** This method explicitly generates "visual action traces" via visual prompts to guide the VLA model in understanding the end-effector's motion path, thereby improving manipulation precision.
- **SpatialVLA Qu et al. (2025):** Focuses on addressing the lack of spatial awareness in VLA models. It injects geometric knowledge into the VLM backbone by co-training action prediction with 3D spatial reasoning tasks, such as depth prediction or 3D bounding box regression.

- **4D-VLA Zhang et al. (2025a):** Further introduces the temporal dimension on top of 3D perception. By processing dynamic 3D point cloud video streams, it enables the model to understand dynamic changes of objects in space-time, generating spatiotemporally consistent action trajectories.
- **HiF-VLA Lin et al. (2025):** A unified framework that leverages motion representations for bidirectional temporal reasoning, encompassing hindsight priors and foresight anticipation. It addresses temporal myopia in VLA models by capturing inter-state dynamics to filter static background noise effectively.

C Detailed Task Specifications

Table 7: Mapping between task IDs and language instructions for LIBERO-Long and real-world evaluations.

Task ID	LIBERO-Long	Real-world
Task 1	Put soup and box in basket	Place Bowl (Test)
Task 2	Put box and butter in basket	Place Bowl (Color)
Task 3	Turn on stove and put pot	Push Cube (Shape)
Task 4	Put bowl in drawer and close	Pick Cube
Task 5	Put mugs on left and right plates	Dual-Arm Handover
Task 6	Pick book and place it in back	Stack Bowls (Instruction)
Task 7	Put mug on plate, pudding right	Stack Bowls (Visual)
Task 8	Put soup and sauce in basket	–
Task 9	Put both pots on stove	–
Task 10	Put mug in microwave and close	–

Table 8: Mapping between task IDs and language instructions for simulation benchmark evaluations.

Task ID	SimplerEnv-Bridge	SimplerEnv-Fractal	MIKASA-Robo
Task 1	PutSpoonOnTableCloth	GraspSingleOpenedCokeCan	ShellGameTouch
Task 2	PutCarrotOnPlate	MoveNearGoogleBakedTex	InterceptMedium
Task 3	StackGreenCubeOnYellowCubeBakedTex	Open/CloseDrawer	TakeItBack
Task 4	PutEggplantInBasket	OpenTopDrawerAndPlaceApple	RememberShape5

D Additional Analysis on LIBERO-Long

D.1 Per-Task Performance Analysis

To complement the average LIBERO-Long results reported in the main paper, we provide per-task success rates across all ten subtasks. This breakdown helps examine whether the improvement of `FibVLA` comes from isolated easy tasks or from consistent gains across diverse long-horizon manipulation scenarios.

Table 9: Comparison of success rates on LIBERO-Long subtasks.

Method	Avg. SR	Task1	Task2	Task3	Task4	Task5	Task6	Task7	Task8	Task9	Task10
OpenVLA	54.0	35.0	95.0	65.0	45.0	40.0	80.0	60.0	45.0	20.0	55.0
OpenVLA-OFT	91.0	82.0	96.0	96.0	94.0	90.0	96.0	92.0	**100**	70.0	**94.0**
π_0	72.0	86.0	86.0	60.0	86.0	82.0	52.0	84.0	82.0	26.0	76.0
UniVLA	63.0	64.0	82.0	76.0	96.0	58.0	98.0	24.0	74.0	32.0	26.0
HiF-VLA	94.4	94.0	**98.0**	**100**	**100**	94.0	**100**	90.0	98.0	76.0	**94.0**
`FibVLA`	**95.2**	**96.0**	**98.0**	96.0	98.0	**96.0**	96.0	**98.0**	**100**	**88.0**	86.0

D.2 Ablation on Visual Encoders and Adaptive Routing

We further evaluate the generality of our temporal sampling strategy on LIBERO-Long. Specifically, we apply the same historical sampling formulation to different visual temporal encoders, including a Hy-VLA-based visual encoder (Zhang et al., 2026) and a TimeSformer encoder (Bertasius et al., 2021). This comparison examines whether the proposed sampling strategy is tied to a specific visual encoder, or whether it can be combined with different temporal visual representation backbones. All results in Table 10 are official runs with 10 tasks and 25 episodes per task; pilot runs, cancelled tasks, and misconfigured tests are excluded.

Beyond fixed Fibonacci sampling, we also study adaptive sampling variants built upon the same temporal sampling principle. These variants do not abandon the original sampling intervals entirely; instead, they preserve the coarse temporal structure of Fibonacci-style history selection while biasing part of the sampled historical context toward task-relevant state transitions, such as gripper opening/closing, fast end-effector motion, or other rapid interaction changes. The results show that the effect of adaptive sampling depends on both the visual encoder and whether recent context is retained. In particular, preserving the most recent frames improves performance under the TimeSformer encoder, while combining the Hunyuan 3D VAE encoder with adaptive recent achieves the best overall result.

Table 10: Official LIBERO-Long ablation results with different visual temporal encoders and sampling strategies. For methods with multiple seeds, we report the better official seed result.

Visual Encoder	Sampling / Routing	Successes	Avg. SR	T0	T1	T2	T3	T4	T5	T6	T7	T8	T9
$\pi_{0.5}$ original	–	234 / 250	93.6	100	96.0	96.0	100	100	96.0	88.0	100	84.0	76.0
Hunyuan 3D VAE	Fixed Fibonacci	239 / 250	95.6	100	100	100	100	92.0	96.0	96.0	96.0	88.0	88.0
Hunyuan 3D VAE	Standard Adaptive	240 / 250	<u>96.0</u>	100	100	100	100	92.0	96.0	96.0	92.0	92.0	92.0
Hunyuan 3D VAE	Adaptive Recent	244 / 250	**97.6**	100	96.0	100	96.0	100	96.0	100	96.0	96.0	96.0
TimeSformer	Fixed	216 / 250	86.4	76.0	92.0	76.0	100	92.0	96.0	88.0	84.0	80.0	80.0
TimeSformer	Standard Adaptive	219 / 250	<u>87.6</u>	84.0	96.0	84.0	100	92.0	96.0	80.0	92.0	80.0	72.0
TimeSformer	Adaptive Recent	222 / 250	**88.8**	80.0	100	92.0	100	80.0	96.0	80.0	96.0	88.0	76.0

E Scoring Rubric for Real-world Experiments

To comprehensively evaluate the error-recovery capabilities and execution fluency of our model in long-horizon tasks, we propose a quantitative scoring framework as follows:

- **Task Decomposition**: Each long-horizon task is decomposed into five key sub-steps (i.e., action primitives), representing the critical stages of the manipulation sequence.
- **Scoring Criteria**: Each sub-step is evaluated based on execution quality using a tiered scoring system:
  - **2 Points (Fluent Execution)**: The robot successfully completes the sub-step in a single, continuous motion without hesitation or significant deviation.
  - **1 Point (Corrective Execution)**: The robot encounters minor deviations or stutters but successfully completes the sub-step through autonomous self-correction.
  - **0 Point (Execution Failure)**: The robot fails to complete the sub-step or encounters an irreversible error, such as a collision or dropping the object.
- **Statistical Protocol**: To ensure statistical significance, each task is conducted over 10 independent trials. The maximum possible score for a single task is calculated as:

$$S_{max} = 5 \text{ sub-steps} \times 2 \text{ points} \times 10 \text{ trials} = 100 \tag{9}$$

Table 11: List of Self-collected Real-world Evaluation Tasks with Decomposed Action Steps

Task & Instruction	Evaluation Steps (5 Sub-steps)	Config	Horizon
Place Bowl (Test): Put the blue plate on desk, pick up the white bowl, put the bowl on the plate.	1. Move to hover directly over the white bowl. 2. Descend vertically and close the gripper to clamp the bowl. 3. Lift the bowl and move horizontally to the plate. 4. Descend until the bowl bottom touches the plate. 5. Open the gripper and retract vertically.	Single	Short
Place Bowl (Spatial): Put the blue plate on desk, pick up the white bowl, put the bowl on the plate.	1. Move to the bowl's random position and align the wrist. 2. Descend and close the gripper upon contact. 3. Lift and reorient the wrist in mid-air to level the bowl. 4. Move carrying the bowl to the coordinates of the plate. 5. Lower the bowl onto the plate and open the gripper.	Single	Short
Place Bowl (Color): Put the bowl on the color [X] plate.	1. Move to hover over the white bowl. 2. Descend, close the gripper, and lift the bowl. 3. Move the arm to the position of the [Color] plate. 4. Descend vertically to place the bowl into the plate. 5. Open the gripper and retract the arm.	Single	Short
Push Cube (Shape): Push the shape [X] block into the red region.	1. Move directly over the specific [Shape] block. 2. Descend and align with the contact surface behind the object. 3. Rotate the gripper to the pushing angle. 4. Push the block horizontally into the red region. 5. Lift the arm vertically to finish the task.	Single	Short
Pick Cube: Pick up the left/middle/right one among the three cubes and put it into the bowl.	1. Move directly above the cube matching the [Position]. 2. Descend and close the gripper to grab the cube. 3. Lift the cube and move above the bowl. 4. Open the gripper to let the cube fall into the bowl. 5. Return the arm to the initial home position.	Single	Short

... Continued from previous page

Task & Instruction	Evaluation Steps (5 Sub-steps)	Config	Horizon
Pick Functional Object: Put the object used for [X] on the plate.	1. Move to the center of mass of the target functional object. 2. Rotate the wrist to a suitable grasping angle and close gripper. 3. Lift the object vertically off the table. 4. Move horizontally to the plate's position. 5. Lower until contact is made and open the gripper.	Single	Short
Wipe Whiteboard: Wipe the whiteboard clean.	1. Move to the eraser, descend, and close the gripper tightly. 2. Move the eraser to the starting corner of the writing. 3. Press down to ensure contact between eraser and board. 4. Execute a wiping motion (zigzag or linear) across the marks. 5. Lift the eraser vertically off the board.	Single	Short
Stack Bowls (Instruction): Stack bowls in the order of color [X], [Y], [Z] from top to bottom.	1. Grasp bowl [Z], place it in the workspace center, release. 2. Grasp bowl [Y], move it above [Z], align, and release. 3. Grasp bowl [X], move it above [Y]. 4. Align carefully and lower [X] onto [Y]. 5. Open gripper and retreat to complete the tower.	Single	Long
Stack Bowls (Visual): Stack bowls in the same/reverse order as the stack on the left.	1. Grasp the bottom-matching bowl and place on the target spot. 2. Grasp the middle-matching bowl from the supply. 3. Stack it precisely onto the bottom bowl and release. 4. Grasp the top-matching bowl from the supply. 5. Stack it onto the middle bowl and release.	Single	Long
Dual-Arm Handover: Pick up cylinder with left arm, pass to right arm.	1. Left arm moves to the cylinder and closes the gripper. 2. Left arm lifts object to center; Right arm moves to meet it. 3. Right arm closes gripper on the free end of the cylinder. 4. Left arm opens gripper to release the object. 5. Right arm moves away with the object; Left arm retracts.	Dual	Short

... *Continued from previous page*

Task & Instruction	Evaluation Steps (5 Sub-steps)	Config	Horizon
Classification: Put all the fruits into the bowl.	1. Move to the first fruit, grasp it, and lift. 2. Move to the bowl and open gripper to release. 3. Move to the second fruit, grasp it, and lift. 4. Move to the bowl and open gripper to release. 5. Repeat until no fruits remain, then return to home.	Single	Long
Mid-air Stacking: Place the bowl on the plate in mid-air.	1. Left arm grasps the plate; Right arm grasps the bowl. 2. Arms lift and meet in the center; Left holds plate flat. 3. Right arm positions the bowl directly above the plate. 4. Right arm lowers gently until the bowl touches the plate. 5. Right arm opens gripper and moves away.	Dual	Short
Uncap Marker: Remove the cap from the marker.	1. Left arm moves to hold the marker body firmly on the table. 2. Right arm moves to the marker cap. 3. Right arm closes gripper to clamp the cap. 4. Right arm pulls horizontally along the axis to detach the cap. 5. Right arm places the cap on the table.	Dual	Short
Make Coffee: Make a cup of instant coffee.	1. Grip the powder cup, pour into the empty cup, and return it. 2. Move to the spoon and grip the handle. 3. Insert the spoon vertically into the filled cup. 4. Perform a circular stirring motion with the wrist. 5. Lift the spoon out of the cup.	Dual	Long
Fold Clothes: Fold the clothes on the table.	1. Move to the left sleeve, grip, fold inward, and release. 2. Move to the right sleeve, grip, fold inward, and release. 3. Move to the bottom hem of the shirt. 4. Grip and lift the hem towards the collar. 5. Release to complete the fold and retract arms.	Dual	Long

F HYPERPARAMETER SETTINGS AND DATASET STATISTICS

Table 12: Hyperparameter settings and data statistics for different evaluation environments.

Category	Hyperparameter	LIBERO	MIKASA-Robo	SimplerEnv-Bridge	SimplerEnv-Fractal	Real-world
Optimization	Optimizer	AdamW	AdamW	AdamW	AdamW	AdamW
	Optimizer Betas	$(0.9, 0.95)$	$(0.9, 0.95)$	$(0.9, 0.95)$	$(0.9, 0.95)$	$(0.9, 0.95)$
	Peak LR	2.5×10^{-5}	2.5×10^{-5}	2.5×10^{-5}	2.5×10^{-5}	2.5×10^{-5}
	Min LR	2.5×10^{-6}	2.5×10^{-6}	2.5×10^{-6}	2.5×10^{-6}	2.5×10^{-6}
	Weight Decay	1.0×10^{-10}	1.0×10^{-10}	1.0×10^{-10}	1.0×10^{-10}	1.0×10^{-10}
	Gradient Clipping	1	1	1	1	1
	LR Scheduler	Cosine Decay	Cosine Decay	Cosine Decay	Cosine Decay	Cosine Decay
	Warmup Steps	1,000	1,000	1,000	1,000	1,000
	Training Steps	30,000	70,000	100,000	70,000	30,000
	Batch Size	16	16	16	32	16
Data	Episodes	1,693	6,000	25,460	26,152	700
	Frames	273,465	585,000	864,292	1,067,618	288,594
	Action Dim	7	8	7	7	7 / 14
	Cameras	1×Wrist +1×3rd	1×Wrist +1×3rd	1×3rd	1×3rd	2×Wrist +1×3rd
	State Dim	8	25	–	–	7 / 14
	Instruction	Natural	Natural	Natural	Templated	Natural

G REAL-WORLD EXPERIMENTAL SETUP AND TASK EXECUTION

The physical experimental platform is centered around the Agilex Piper robot arm, integrated with the Agilex Pika teleoperation kit for demonstration data collection. Visual feedback is provided by wrist-mounted and third-person Intel RealSense D435 cameras, supporting both manual teleoperation and autonomous model inference. To ensure the reproducibility of our experiments, we deliberately designed the environment using readily available components. The setup employs a standardized white background board and a black tablecloth to minimize environmental interference, while all task objects are common household items to facilitate the benchmarking of `FibVLA` in accessible settings.

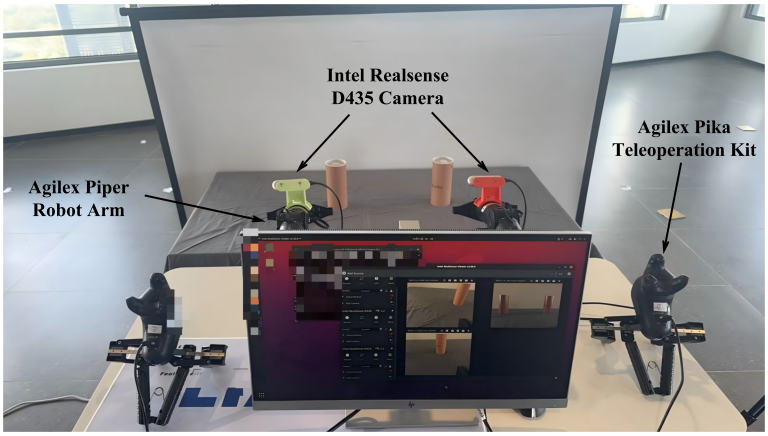
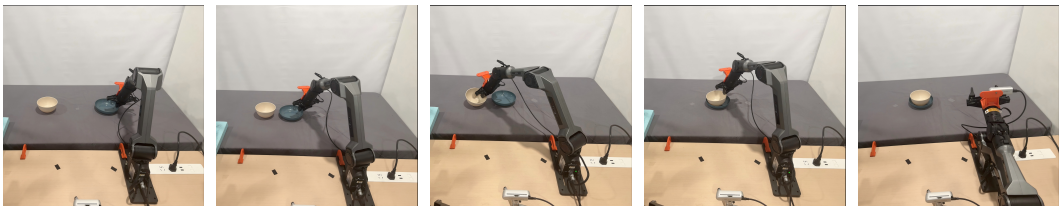


Figure 7: Real-world experimental scenario

We use the "Place Bowl (Test)" task as an illustrative example to demonstrate the real-world inference performance of `FibVLA`. To enhance generalization, the demonstration data include balanced initial configurations (e.g., swapping the left-right positions of the bowl and plate as shown in Fig. 8a), and the model successfully executes the task regardless of the initial layout. Crucially, `FibVLA` exhibits a deep understanding of temporal logic rather than mere imitation learning. When external interventions revert the task state during inference (e.g., resetting from Fig. 8c back to Fig. 8a, or Fig. 8e back to Fig. 8c), the model autonomously perceives the environmental shift and re-executes the necessary preceding actions. Such dynamic error recovery proves that `FibVLA` explicitly models temporal information to achieve robust long-horizon reasoning.



(a) Approach and grasp plate (b) Place plate on center (c) Approach and grasp bowl (d) Place bowl on plate (e) Return to home position

Figure 8: Key-step snapshots of the "Place Bowl (Test)" task during real-world inference.

H Proof of Upper Bound on Coincidence Count for Logarithmic Hindsight Sampling

Definition H.1 (Logarithmic Sampling Sequence)**.** Let $S = \{x_0, x_1, x_2, \dots\}$ be a strictly increasing integer sequence with anchor $x_0 := 0$. For $i \geq 1$, the elements are generated by a base parameter $q_{\min} > 0$ and a growth rate $r > 1$:

$$x_i = \lfloor q_{\min} \cdot r^i \rfloor. \tag{10}$$

Definition H.2 (Sparse Sampling Constraint)**.** The sequence S is said to satisfy the *sparse sampling constraint* if any three consecutive terms satisfy the second-order recurrence inequality

$$x_i \geq x_{i-1} + x_{i-2}, \quad \forall i \geq 2. \tag{11}$$

Definition H.3 (Coincidence Count)**.** For a given step size $\Delta \in \mathbb{Z}^+$, define the coincidence count function

$$H(\Delta) = \left|\left\{(j, i) \in \mathbb{Z}^2_{\geq 0} \;\middle|\; j > i,\ x_j - x_i = \Delta\right\}\right|. \tag{12}$$

Proposition H.4 (Bounded Coincidence Count)**.** *Let S be any sequence satisfying the sparse sampling constraint. Then for all $\Delta > 0$,*

$$H(\Delta) \leq 3. \tag{13}$$

Moreover, equality can only occur if the sequence attains the Fibonacci boundary $x_i = x_{i-1} + x_{i-2}$ and $\Delta \in S$.

Proof. We consider solutions to the equation $x_j - x_i = \Delta$ with $j > i$.

Any sequence satisfying $x_i \geq x_{i-1} + x_{i-2}$ dominates the Fibonacci sequence term-wise. Consequently, for any fixed index pair (j, i), the difference $x_j - x_i$ is minimized in the tight boundary case where equality holds, i.e.,

$$x_i = x_{i-1} + x_{i-2}. \tag{14}$$

Hence, the maximal possible value of $H(\Delta)$ is achieved under the Fibonacci recurrence, which corresponds to the minimal growth rate compatible with the sparse sampling constraint. It therefore suffices to analyze this extremal case. Without loss of generality, consider $\Delta = x_k \in S$ for some $k \geq 1$. If $\Delta \notin S$, monotonicity immediately reduces the number of admissible solutions. We enumerate all possible solutions to $x_j - x_i = x_k$:

Case 1 ($i = 0$)**:** Since $x_0 = 0$, the trivial solution $(j, i) = (k, 0)$ always exists.

Case 2 ($j = i + 1$)**:** From the Fibonacci recurrence,

$$x_{i+1} - x_i = x_{i-1}.$$

Setting $x_{i-1} = x_k$ yields $i = k + 1$, giving the solution $(j, i) = (k + 2, k + 1)$.

Case 3 ($j = i + 2$)**:** We have

$$x_{i+2} - x_i = (x_{i+1} + x_i) - x_i = x_{i+1}.$$

Setting $x_{i+1} = x_k$ yields $i = k - 1$, giving the solution $(j, i) = (k + 1, k - 1)$.

Case 4 ($j - i \geq 3$)**:** We show that no solutions exist in this regime.

- If $j \geq k + 2$, then for all admissible i,

$$x_j - x_i \geq x_{k+2} - x_{k-1} = 2x_k > x_k.$$

- If $j = k + 1$ and $i \leq k - 2$,

$$x_{k+1} - x_{k-2} = 2x_{k-1} > x_k,$$

since $x_k = x_{k-1} + x_{k-2}$.

Thus, $x_j - x_i$ strictly exceeds x_k, and no solutions arise.

Collecting all cases, there are at most three admissible solutions. For sequences with strict inequality $x_i > x_{i-1} + x_{i-2}$, Cases 2 and 3 are no longer attainable, and $H(\Delta)$ is further reduced. Therefore, for all sequences satisfying the sparse sampling constraint,

$$H(\Delta) \leq 3.$$

□

Remark (Connection to Logarithmic Sampling) Although the Fibonacci sequence is defined by a linear recurrence, its closed-form expression given by Binet's formula is

$$F_i = \frac{\phi^i - \psi^i}{\sqrt{5}}, \tag{15}$$

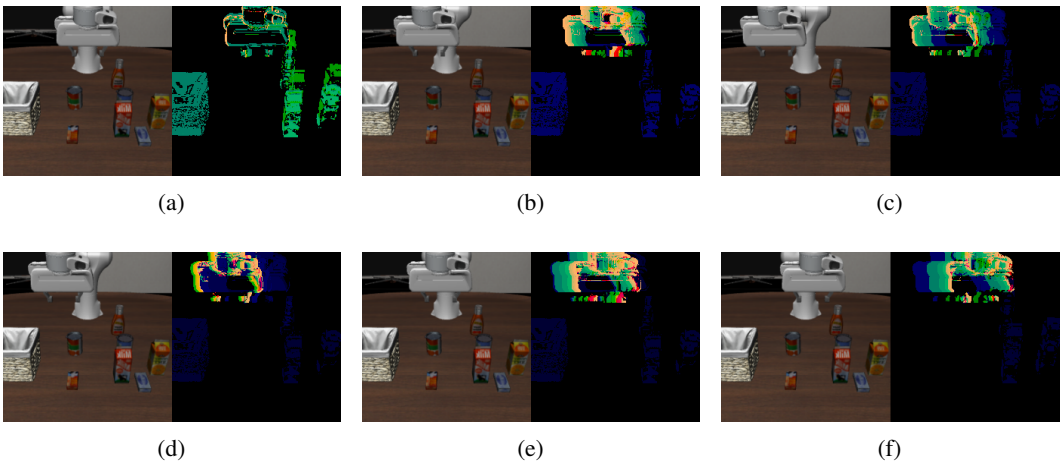
where $\phi = \frac{1+\sqrt{5}}{2}$ and $\psi = \frac{1-\sqrt{5}}{2}$. Since $|\psi| < 1$, the term ψ^i decays exponentially, yielding the asymptotic relation

$$F_i \sim \frac{1}{\sqrt{5}}\phi^i. \tag{16}$$

Thus, the Fibonacci sequence can be viewed as an asymptotic logarithmic sampling sequence with growth rate $r = \phi$. This observation justifies treating the Fibonacci recurrence as the extremal boundary case when analyzing coincidence bounds for logarithmic hindsight sampling sequences.

I Visualization of Channel-wise Temporal Encoding

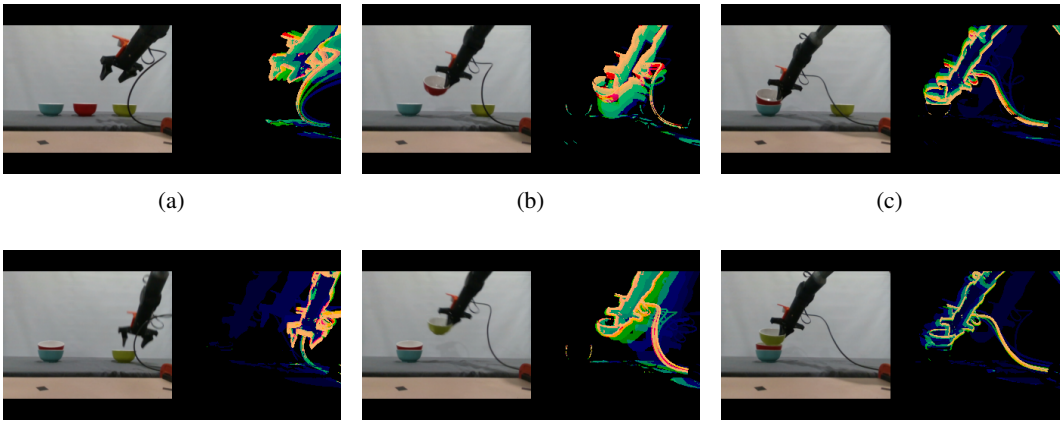
The image hindsight effectively represents the temporal evolution of the task within the latent space. A notable phenomenon in these visualizations is the variation in the morphology of the feature "trails." In the LIBERO environment, which operates at a lower sampling frequency of 10Hz, the processed output exhibits elongated trails due to the larger inter-frame displacement. Conversely, in the real-world experiments conducted at 30Hz, the encoding appears more compact and remains closely aligned with the object boundaries. Rather than being visual artifacts, these variations in trail length and distribution serve as a direct mapping of the object's motion dynamics, such as velocity and trajectory. This demonstrates that the CTE module successfully integrates the temporal progression of the task into the current visual representation.



(a) (b) (c)

(d) (e) (f)

Figure 9: Visualization of visual inputs and corresponding image hindsight in the LIBERO simulation environment (10Hz). Each subfigure (a–f) displays the raw RGB camera observation on the left and the processed output from the channel-wise temporal encoding (CTE) module on the right. The Image Hindsight captures the temporal progression of the task within the latent representation.



(a) (b) (c)

(d) (e) (f)

Figure 10: Visualization of visual inputs and image hindsight in real-world Piper robotic arm experiments (30Hz). Similar to the simulation, the left panels show raw physical observations, while the right panels display the corresponding processed image hindsight.