

MemoryVLA++: Temporal Modeling via Memory and Imagination in Vision-Language-Action Models

Hao Shi[†], Weiye Li, Bin Xie[†], Yulin Wang, Renping Zhou, Tiancai Wang, Xiangyu Zhang,
Ping Luo, *Senior Member, IEEE*, Gao Huang *Member, IEEE*,[§]

Abstract—Temporal modeling is essential for robotic manipulation, as effective control requires both memory of past interactions and imagination of future states. However, most VLA models rely primarily on the current observation and therefore struggle with long-horizon, temporally dependent tasks. Cognitive science suggests that humans rely on working memory to buffer short-lived context, the hippocampal system to preserve episodic memory of past experience, and internal models to imagine possible future state evolution. Inspired by these mechanisms, we propose MemoryVLA++, a full temporal modeling framework that equips VLA models with memory and imagination for robotic manipulation. A pretrained VLM encodes the current observation into perceptual and cognitive tokens, forming working memory. These tokens query a Perceptual-Cognitive Memory Bank to retrieve relevant historical context. This bank stores low-level details and high-level semantics from past interactions, and is updated through redundancy-aware consolidation. A world model imagines future states in a denoising latent space, and the imagined latents are integrated under memory guidance to form full temporal-aware tokens. The resulting tokens condition a diffusion action expert to predict temporally consistent action sequences. We conduct extensive experiments on 5 simulation benchmarks and 3 categories of real-robot tasks across 3 robots, covering general manipulation, long-horizon temporal tasks, robustness, and generalization. Our method achieves strong performance across Libero, SimplerEnv, Mikasa-Robo, Calvin, Libero-Plus, and diverse real-robot tasks, validating the effectiveness of full temporal modeling with memory and imagination. For example, on real robots, it achieves +9% gains on general manipulation, +26% on long-horizon memory-dependent tasks, and +28% on long-horizon imagination-dependent tasks.

Index Terms—Embodied AI, vision-language-action (VLA) models, memory, world model, robotic manipulation.

I. INTRODUCTION

VISION-LANGUAGE-ACTION (VLA) models [1]–[3], powered by large-scale cross-embodiment robotic datasets [4]–[7] and pretrained Vision-Language Models (VLMs) [8], [9], have achieved remarkable progress in robotic manipulation. However, typical VLA models such as OpenVLA [1] and π_0 [2] rely solely on the current observation,

thereby overlooking temporal dependencies and performing poorly on long-horizon temporal tasks. As shown in Fig. 1(a), *Button Pressing* tasks require memory, since the observations before and after pressing exhibit almost no visual difference, making it difficult to determine whether the button has already been pressed. Meanwhile, *Dynamic-Conveyor Grasping* tasks require imagination, as perceiving how objects will move on a dynamic conveyor enables the robot to grasp at a more appropriate time. These examples highlight that effective robotic manipulation requires both remembering past interactions and anticipating how the scene will evolve in the future.

A straightforward way is to expand the VLA input with additional frame-level observations from both the past and the future. Specifically, the most recent N history frames [10]–[12] and predicted future video frames [13]–[16] are fed into the VLA model. However, simply expanding the visual observation sequence is inefficient for robotic manipulation. On the past side, concatenating consecutive history frames incurs quadratic self-attention cost and introduces substantial temporal redundancy, limiting the usable context length and impairing long-term dependency modeling. This issue is especially pronounced in robotic manipulation, where slow motions stretch decision-relevant interactions over longer time spans. On the future side, future video prediction is computationally expensive and often focuses on pixel-level fidelity rather than control-relevant dynamics. Directly feeding predicted frames into the VLA model may further propagate visual prediction errors to action generation. This motivates a temporal modeling framework that efficiently compresses past experience into long-term memory and captures future dynamics through compact, decision-relevant imagination.

Research in cognitive science [17]–[21] suggests that humans handle manipulation tasks by using working-episodic memory to retain past experiences and internal models to anticipate future state evolution, as illustrated in Fig. 1 (b). Multimodal sensory information is encoded into perceptual and cognitive representations, which are temporarily maintained in working memory for immediate decision-making. In parallel, episodic memory, a long-term memory system closely associated with the hippocampus, stores past experiences as both verbatim representations that preserve precise details and gist representations that capture abstract semantics. Beyond memory, internal models further enable humans to anticipate possible future states before action execution. During execution, working memory retrieves decision-relevant contexts from episodic memory and integrates them with current representations and future anticipation, forming a temporal repre-

Hao Shi is with the Department of Automation, BNRist, Tsinghua University, Beijing, China and also with the Department of Computer Science, The University of Hong Kong, Hong Kong, China.

Weiye Li, Yulin Wang, Renping Zhou, and Gao Huang are with the Department of Automation, BNRist, Tsinghua University, Beijing, China.

Bin Xie and Tiancai Wang are with Dexmal, Beijing, China.

Xiangyu Zhang is with StepFun, Beijing, China.

Ping Luo is with the Department of Computer Science, The University of Hong Kong, Hong Kong, China.

[†] denotes project leaders, and [§] denotes corresponding author.

E-mail: shi-h23@mails.tsinghua.edu.cn; gaohuang@tsinghua.edu.cn.

Project page: <https://shihao1895.github.io/MemoryVLA-PP-Web>.

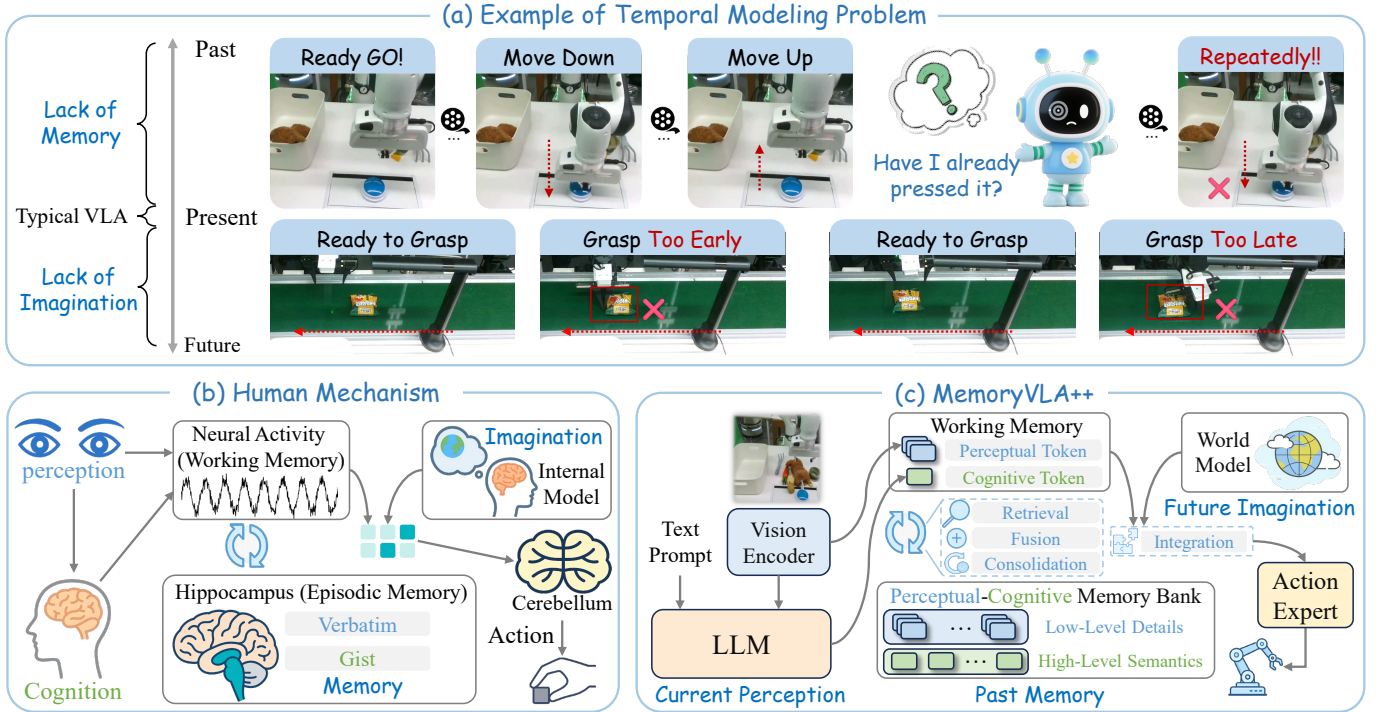


Fig. 1. Motivation of MemoryVLA++. (a) Button pressing shows the need for memory: similar visual states before and after pressing make it hard to know whether the button has already been pressed. Dynamic-conveyor grasping shows the need for imagination: predicting future object motion helps grasp at the right time. (b) Humans leverage the hippocampal system to maintain working-episodic memory, and use internal models to imagine future state evolution. (c) Inspired by these, MemoryVLA++ enables full temporal modeling in VLA models by combining present perception, past memory, and future imagination.

sensation that guides motor execution through the cerebellum, while new experiences are consolidated into episodic memory.

Drawing on cognitive science insights, we propose MemoryVLA++, a full temporal modeling framework that equips VLA models with memory and imagination for robotic manipulation, as shown in Fig. 1 (c). A vision encoder extracts perceptual tokens from the current observation, while an LLM reasons over these tokens and language tokens to produce cognitive tokens using its commonsense prior. Perceptual and cognitive tokens jointly form the working memory. To capture long-term temporal context, a Perceptual-Cognitive Memory Bank (PCMB) stores low-level perceptual details and high-level cognitive semantics from past interactions. Working memory queries the PCMB to retrieve decision-relevant historical contexts, which are adaptively fused with current tokens through a gating mechanism. Meanwhile, the PCMB is updated through redundancy-aware consolidation, merging temporally adjacent and semantically similar entries to keep memory compact. To anticipate future state evolution, a video-generation world model performs partial denoising in the latent space to obtain imagined future tokens. Guided by memory-augmented tokens, these imagined tokens are integrated into full temporal tokens that combine present perception, past memory, and future imagination. The resulting tokens condition a diffusion action expert, where cognitive tokens provide high-level semantic guidance and perceptual tokens supply fine-grained visual details, producing temporally consistent robotic actions. Fig. 2 compares the main idea of MemoryVLA++ with typical VLAs and MemoryVLA, illus-

trating full temporal modeling with memory and imagination.

We conduct extensive experiments on 5 simulation benchmarks and 3 categories of real-robot tasks across 3 robots, covering nearly 200 tasks with diverse variations. In simulation, MemoryVLA++ achieves success rates of 98.4% on Libero [22] and 74.0% on SimplerEnv [23], consistently outperforming strong baselines with a maximum gain of 16.7 percentage points on SimplerEnv. On long-horizon and temporal tasks, MemoryVLA++ achieves a 44.4% success rate on Mikasa-Robo [24] and a score of 4.29 on Calvin [25], improving over the baseline by 15.0 percentage points on Mikasa-Robo. For robustness and generalization, MemoryVLA++ further achieves an 82.7% success rate on Libero-Plus under diverse task and environment variations. In real-robot experiments, we evaluate our method on general manipulation, long-horizon memory-dependent tasks, and long-horizon imagination-dependent tasks. It achieves scores of 85%, 83%, and 77%, outperforming the baseline by 9, 26, and 28 percentage points, respectively. These results validate the effectiveness of full temporal modeling with memory and imagination for robotic manipulation.

This work substantially extends the conference version MemoryVLA [26] with several important new contributions, as summarized below.

- We advance VLA temporal modeling from past-only memory to full temporal modeling over the past, present, and future, equipping VLA models with both memory and imagination (Fig. 2). Inspired by cognitive science, the framework extends the previous perceptual-cognitive

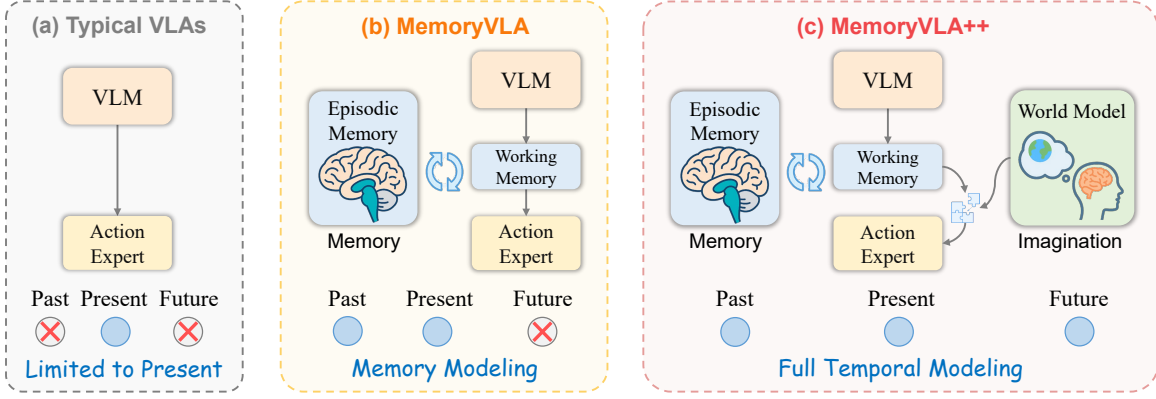


Fig. 2. Comparison of the main ideas of three paradigms: Typical VLAs are reactive and rely only on the present observation, MemoryVLA introduces a working memory-episodic memory mechanism to capture past temporal dependencies, and MemoryVLA++ further extends this by incorporating future imagination via a world model for full temporal modeling.

memory mechanism with a world-model-based imagination mechanism, forming MemoryVLA++ (Fig. 3).

- To leverage the world model for future imagination, we introduce two key components: (1) a latent-space imagination generation that uses the world model to capture decision-relevant future dynamics via partial denoising, avoiding costly pixel-level prediction; (2) a memory-guided imagination integration module that adaptively integrates imagined future latents with memory-augmented tokens to produce full temporal-aware representations. (Sec. III-D, Fig. 5).
- The experimental evaluation has been substantially expanded. We report updated MemoryVLA++ results on five simulation benchmarks, including Libero, SimplerEnv, Mikasa-Robo, Calvin, and Libero-Plus, covering nearly 200 tasks with diverse variations (Tabs. I, II, III, IV, and V). Among them, Calvin and Libero-Plus are newly added to evaluate long-horizon performance and robustness & generalization, respectively.
- We further expand the real-robot evaluation with long-horizon imagination-dependent tasks, demonstrating MemoryVLA++’s effectiveness in practical deployment (Tab. VI). The evaluation is also extended to a dual-arm ARX5 platform, in addition to the previously used single-arm robots, covering three robots in total (Fig. 8).
- Additional analytical results are provided, including ablation studies of new components (Tab. VIII), analysis of efficiency (Tab. X), analysis of the world model (Fig. 12, Tab. IX), exploration of stronger Qwen [9] VLM backbones with dexbotic [27] pretraining (Tab. XI), and new visualization results for real-robot tasks (Fig. 11).

II. RELATED WORK

A. Vision-Language-Action Models

Driven by advances in visual foundation models [28]–[35], robot imitation learning has progressed rapidly, yet remains confined to small, task-specific policies with limited generalization [36]–[39]. To overcome these limitations, the success of VLMs [9], [40], [41] and large-scale robot datasets,

such as OXE [4] and Agibot [7], has spawned the vision-language-action (VLA) paradigm [1], [2], [42]–[44]. RT-2 [45] and OpenVLA [1] tokenize continuous actions into discrete tokens and use VLMs for autoregressive prediction as if generating language. In contrast, π_0 [2], CogACT [3], and GR00T-N1 [46] adopt diffusion-based policies [36] as action heads, leveraging iterative denoising to sample continuous control trajectories and capture diverse multimodal behaviors. However, most existing VLA models mainly rely on current observations, without explicitly modeling past interaction history or future state evolution, and therefore struggle with long-horizon temporal tasks.

B. Memory Modeling for Robotic Manipulation

Memory modeling has been extensively studied in computer vision and autonomous driving [47], [48], yet it has not been fully explored in robotic manipulation. Octo [49], RoboVLMs [50], and Interleave-VLA [51] organize past observations into interleaved image-text sequences, while TTF-VLA [10] and ContextVLA [11] directly concatenate neighboring frames. Although these approaches provide some temporal context, they are computationally expensive and limited to short horizons. Another line of work encodes history into latent representations. CronusVLA [52], 4D-VLA [53], and HAMLET [54] concatenate neighboring latent frames, SAM2Act [55] augments latent features with heatmap masks, and HiF-VLA [56] captures dynamics through motion cues. Although these methods use historical information efficiently, they rely on sliding windows, cover only short-horizon local history, and often emphasize either perceptual details or high-level semantics rather than both. Beyond direct encoding of historical observations, some methods exploit abstract or indirect temporal cues. TraceVLA [57] paints historical states as trajectories on the current frame, potentially missing fine-grained details. PTP [58] introduces supervision on past action tokens, providing implicit short-term consistency. Language planning methods such as MemER [59], Mem-0 [60], and MEM [61] rely on external VLMs and non-end-to-end pipelines, introducing substantial overhead and complicating coordination between high-level planning and low-level

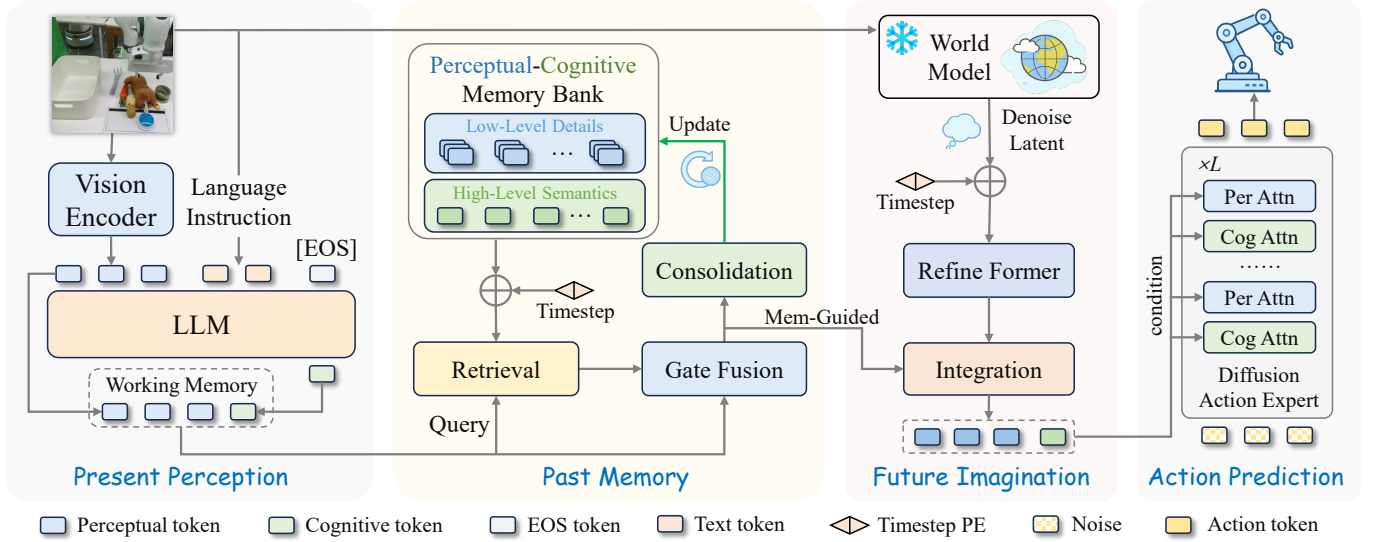


Fig. 3. Overall architecture. The current RGB observation and language instruction are encoded by a 7B VLM into perceptual and cognitive tokens, forming working memory. The working memory queries a Perceptual-Cognitive Memory Bank (PCMB) to retrieve relevant historical context with high-level semantics and low-level details. The retrieved context is adaptively fused with current tokens, while the PCMB is updated by merging the most similar neighbors. A world model imagines future states in a denoising latent space, and the imagined latents are integrated under memory guidance to form full temporal-aware tokens. These tokens condition a diffusion action expert to predict temporally coherent action sequences.

VLA. In contrast, our method captures long-horizon memory through an end-to-end framework, jointly preserving high-level cognitive semantics and low-level perceptual details, while incorporating imagination for full temporal modeling.

C. Future Modeling for Robotic Manipulation

Generative video models [62]–[64] provide a promising direction for robotic future modeling, as they encode rich spatio-temporal priors and implicit physical dynamics from large-scale video data. One line of work explicitly predicts future visual states to guide action prediction. SuSIE [16] generates subgoal images for goal-conditioned control, while UniPi [15] formulates policy learning as video generation followed by inverse dynamics. Recent works such as Fore-Act [13], VISTA [14], and $\pi_{0.7}$ [65] further extend this paradigm to VLA models by jointly modeling future visual prediction and subtask language planning. However, explicit visual prediction can be computationally expensive and may struggle to capture fine-grained physical dynamics. Another line of work models visual dynamics and action prediction in a shared latent space. VPP [66] and Mimic-Video [67] adapt video diffusion models to robot data and use their representations as visual policy features. Seer [68] builds an end-to-end framework that jointly learns future-state prediction and inverse dynamics for action prediction. Although these methods efficiently capture fine-grained future dynamics in compact latent spaces, their predictions may contain decision-irrelevant noise, and the lack of explicit memory modeling leads to incomplete temporal representations. In contrast, we extract compact future dynamics from partially denoised latents and selectively integrate them under memory guidance, suppressing decision-irrelevant noise while unifying past, present, and future temporal cues.

III. METHOD

A. Overview

1) *Problem Formulation*: We formulate robotic manipulation with VLA models as a sequential decision-making problem, where visual observations and language instruction are mapped to actions for real-world interaction. Given RGB observations from one or multiple camera views, denoted as $I = \{I^v\}_{v=1}^V$ with $I^v \in \mathbb{R}^{H \times W \times 3}$, and a language instruction L , a parameterized policy π predicts a sequence of actions:

$$\mathcal{A} = (a_1, \dots, a_T) = \pi(I, L). \quad (1)$$

For single-arm manipulation, each action is defined as

$$a_t = [\Delta x, \Delta y, \Delta z, \Delta \theta_x, \Delta \theta_y, \Delta \theta_z, g]^\top, \quad (2)$$

which contains relative end-effector translation, relative rotation represented by Euler angles, and a binary gripper state $g \in \{0, 1\}$. For dual-arm manipulation, a_t is defined as the concatenation of the action vectors of both arms.

2) *Overall Architecture*: MemoryVLA++ is an end-to-end framework for robotic manipulation, as shown in Fig. 3. Given the current observation and language instruction, a VLM first encodes them into perceptual and cognitive tokens as working memory. To incorporate historical context, we introduce a Perceptual-Cognitive Memory Bank (PCMB) that stores both high-level semantics and fine-grained perceptual details from previous interactions. The working memory queries the PCMB to retrieve decision-relevant historical representations, which are then adaptively fused with the current representations through a gating mechanism. Meanwhile, the PCMB is updated through redundancy-aware consolidation, where temporally adjacent and semantically similar entries are merged when the memory capacity is reached. To further model future dynamics, the current observation and language instruction

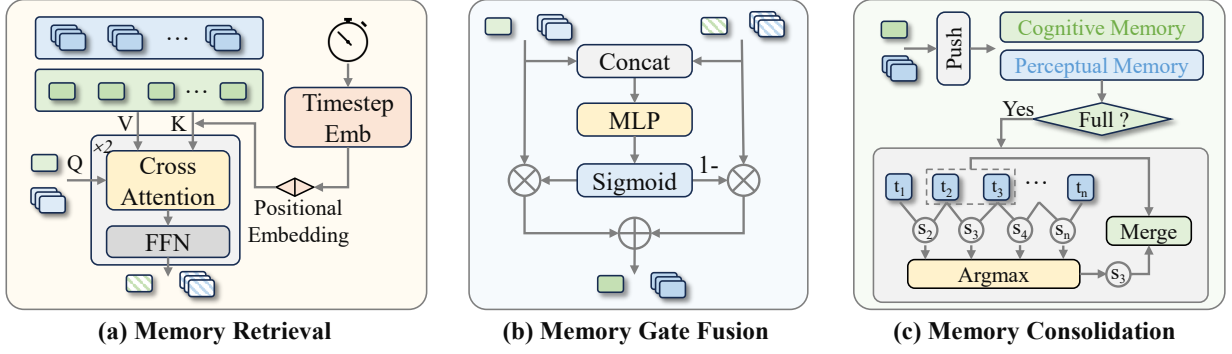


Fig. 4. Details of memory module. (a) Retrieval: current perceptual and cognitive tokens query the PCMB via cross-attention with timestep PE to fetch relevant historical context. (b) Gate fusion: current tokens and retrieved histories are adaptively fused through a gating mechanism. (c) Consolidation: the current tokens are written back to the PCMB. When the PCMB reaches capacity, the most similar adjacent entries are merged to keep the memory compact.

are fed into a pretrained world model, which performs partial denoising to generate multi-scale imagined latents. Guided by the memory-enhanced representations, these imagined latents are adaptively integrated with them to preserve decision-relevant future cues while suppressing irrelevant predictions. The resulting full temporal-aware representations are finally used to condition a diffusion action expert to predict a sequence of T future 7-DoF actions.

B. Vision-Language-Cognition Module

The vision-language-cognition module produces perceptual tokens for fine-grained visual details and cognitive token for high-level semantics from visual observations and language instructions. We build this module upon a 7B-parameter Prismatic VLM [8], which is further pretrained on the large-scale cross-embodiment real-robot dataset Open-X Embodiment [4]. Given RGB observations from one or multiple camera views, denoted as $I = \{I^v\}_{v=1}^V$, we extract visual features from each view using parallel DINOv2 [28] and SigLIP [29] encoders, and concatenate their outputs as raw visual tokens. These tokens are then processed by two parallel branches. In the perceptual branch, a SE-bottleneck-based compression module [69] reduces the channel dimension of the raw visual tokens and produces perceptual tokens $p \in \mathbb{R}^{N_p \times d_p}$, which preserve fine-grained visual cues for manipulation. In the cognitive branch, the raw visual tokens are projected into the language embedding space and concatenated with the tokenized instruction. The resulting multimodal sequence is fed into LLaMA-7B [41], and the output at the end-of-sentence (EOS) position is used as the cognitive token $c \in \mathbb{R}^{1 \times d_c}$, capturing compact high-level semantics. The perceptual tokens p and cognitive token c together form the working memory.

C. Perceptual-Cognitive Memory Modeling

The working memory is defined as

$$M_{\text{wk}} = \{p \in \mathbb{R}^{N_p \times d_p}, c \in \mathbb{R}^{1 \times d_c}\}, \quad (3)$$

where p and c denote the current perceptual tokens and cognitive token, respectively. However, this working memory mainly captures the current observation and lacks temporal

dependencies. To address this limitation, we introduce the Perceptual-Cognitive Memory Bank (PCMB):

$$M_{\text{pcmb}} = \{m^x \mid x \in \{p, c\}\}, \quad (4)$$

$$m^x = \{m_i^x \in \mathbb{R}^{N_x \times d_x}\}_{i=1}^L, \quad x \in \{p, c\}, \quad (5)$$

where $N_c = 1$ for the cognitive stream. Each perceptual entry m_i^p stores fine-grained visual details, while each cognitive entry m_i^c encodes a compact high-level semantic summary. The PCMB maintains up to L temporally ordered entries.

1) *Memory Retrieval*: At each timestep, the working memory M_{wk} , consisting of perceptual tokens p and cognitive token c , acts as the query to retrieve decision-relevant historical information from the PCMB, as illustrated in Fig. 4 (a). For each stream $x \in \{p, c\}$, we set the query as

$$q^p = p, \quad q^c = c. \quad (6)$$

Each memory entry is associated with its episode timestep via a sinusoidal timestep embedding $\text{TE}(\cdot)$, which is projected to the corresponding dimension and added as positional encoding. The keys and values for stream x are constructed as

$$K^x = [m_1^x + \text{TE}(t_1); \dots; m_L^x + \text{TE}(t_L)], \quad (7)$$

$$V^x = [m_1^x; \dots; m_L^x], \quad (8)$$

where the timestep embedding is broadcast when needed.

The perceptual memory entries are stacked into $K^p, V^p \in \mathbb{R}^{L N_p \times d_p}$, while the cognitive memory entries are stacked into $K^c, V^c \in \mathbb{R}^{L \times d_c}$. Scaled dot-product attention then retrieves historical representations for each stream:

$$\hat{H}^x = \text{softmax}\left(\frac{q^x(K^x)^\top}{\sqrt{d_x}}\right) V^x, \quad x \in \{p, c\}. \quad (9)$$

This attention operation is followed by a feed-forward network to form one Transformer layer. After applying two such layers, we obtain the final retrieved embeddings H^p and H^c .

2) *Memory Gate Fusion*: As illustrated in Fig. 4 (b), the retrieved embeddings H^p and H^c are adaptively fused with the current representations through learned gates. For each stream $x \in \{p, c\}$, the gating vector is computed from the current representation x and the retrieved embedding H^x :

$$g^x = \sigma(\text{MLP}(\text{concat}[x, H^x])). \quad (10)$$

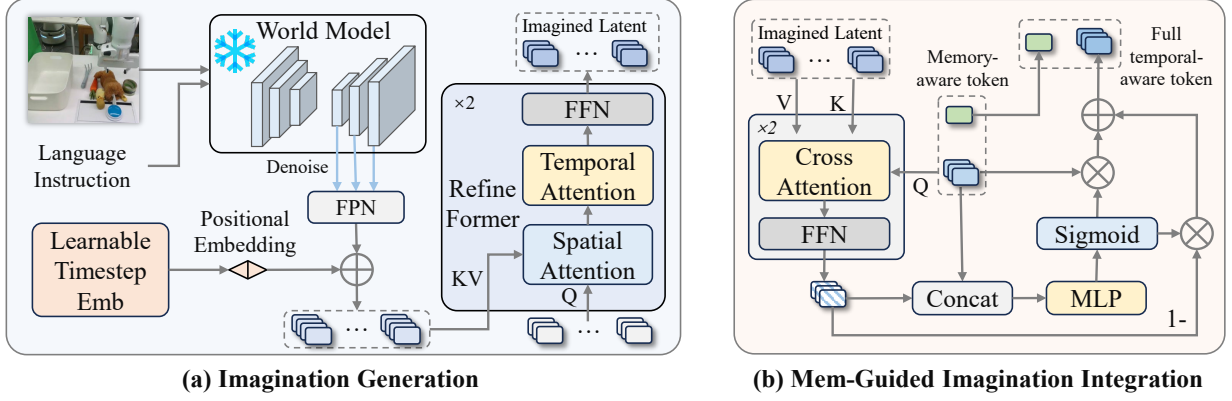


Fig. 5. Details of imagination module. (a) Imagination generation: conditioned on the current observation and instruction, the world model denoises multi-scale future latent tokens, followed by spatial and temporal attention to capture decision-relevant future state evolution. (b) Memory-guided imagination integration: memory-aware tokens attend to imagined latents and adaptively fuse future cues to form full temporal-aware tokens.

The memory-augmented representation is then obtained by

$$\tilde{x} = g^x \odot H^x + (1 - g^x) \odot x. \quad (11)$$

Here, σ denotes the sigmoid activation and $\odot$ denotes element-wise multiplication. This produces memory-augmented perceptual and cognitive representations $\tilde{p}$ and $\tilde{c}$.

3) *Memory Consolidation*: After gate fusion, the memory-augmented representations $\tilde{p}$ and $\tilde{c}$ are passed to downstream modules and simultaneously written into the PCMB. When the number of stored entries exceeds the memory capacity L , redundancy-aware consolidation is performed to keep the memory compact, as illustrated in Fig. 4 (c).

For each stream $x \in \{p, c\}$, cosine similarities are computed between temporally adjacent memory entries. The most similar adjacent pair is merged by averaging their representations:

$$\begin{aligned} i_x^* &= \arg \max_{i=1, \dots, L-1} \cos(m_i^x, m_{i+1}^x), \\ m_{i_x^*}^x &\leftarrow \frac{1}{2} (m_{i_x^*}^x + m_{i_x^*+1}^x), \quad x \in \{p, c\}. \end{aligned} \quad (12)$$

The merged neighbor $m_{i_x^*+1}^x$ is then removed from the corresponding stream. This mechanism mitigates memory bloat by merging redundant adjacent entries, while maintaining compact memory for long-term modeling.

D. World Model-Based Imagination Modeling

Although perceptual-cognitive memory provides historical context from past interactions, robotic manipulation also requires anticipating how the current scene may evolve in the near future. Instead of explicitly predicting future RGB frames, which is computationally expensive and often introduces control-irrelevant pixel details, we use a video-generation world model as a latent imagination module. The world model extracts compact future cues from partially denoised features, and a memory-guided integration module further suppresses irrelevant imagined content to produce full temporal-aware tokens for action prediction.

1) *Manipulation-Oriented World Model Adaptation*: We instantiate the world model with Stable Video Diffusion (SVD) [62], a 1.5B-parameter video diffusion model pre-trained on large-scale Internet videos. Following VPP [66],

we condition SVD on both the current observation I and instruction L , where L is encoded by CLIP and injected into the spatio-temporal UNet via cross-attention. Although SVD provides strong visual dynamics priors, we adapt it on manipulation videos to better align with robotic manipulation.

Given a training sample $(I, L, x_0) \sim \mathcal{D}_{\text{wm}}$, where x_0 denotes the target future video sequence, we sample its noisy version x_τ at diffusion timestep τ :

$$x_\tau = \sqrt{\bar{\alpha}_\tau} x_0 + \sqrt{1 - \bar{\alpha}_\tau} \epsilon, \quad \epsilon \sim \mathcal{N}(0, 1). \quad (13)$$

Here, $\bar{\alpha}_\tau$ denotes the cumulative noise schedule. The world model $\mathcal{W}_\phi$ is trained to reconstruct the clean future video sequence conditioned on I and L :

$$\mathcal{L}_{\text{wm}} = \mathbb{E}_{(I, L, x_0) \sim \mathcal{D}_{\text{wm}}, \epsilon, \tau} [\|\mathcal{W}_\phi(x_\tau, \tau, I, L) - x_0\|_2^2]. \quad (14)$$

The adaptation dataset $\mathcal{D}_{\text{wm}}$ consists of manipulation videos for learning robot-centric priors of future state evolution.

2) *Imagination Generation*: As illustrated in Fig. 5 (a), during VLA training, the manipulation-adapted world model is frozen and used only for latent imagination. Instead of decoding future RGB frames, we perform partial denoising and extract multi-scale intermediate UNet features $\{U_s\}_{s=1}^S$, where $U_s \in \mathbb{R}^{K \times C_s \times H_s \times W_s}$. Here, S is the number of feature scales and K is the number of imagined future steps. An FPN [70] is used to aggregate these features into latent tokens:

$$z = \text{FPN}(\{U_s\}_{s=1}^S), \quad z \in \mathbb{R}^{K \times N_z \times d_p}, \quad (15)$$

where N_z is the number of latent tokens per imagined step and d_p matches the perceptual-token dimension. A learnable temporal embedding $e_{\text{time}} \in \mathbb{R}^{K \times 1 \times d_p}$ is added as

$$\tilde{z} = z + e_{\text{time}}. \quad (16)$$

We then use an imagination former to compress the latent tokens. Learnable queries $q \in \mathbb{R}^{K \times N_q \times d_p}$ attend to the latent tokens through query-based spatial attention:

$$\hat{q}_k = \text{SpatAttn}(q_k, \tilde{z}_k, \tilde{z}_k), \quad k = 1, \dots, K. \quad (17)$$

The queries are further processed by temporal attention:

$$z_{\text{img}} = \text{FFN}(\text{TempAttn}(\hat{q}_{1:K})). \quad (18)$$

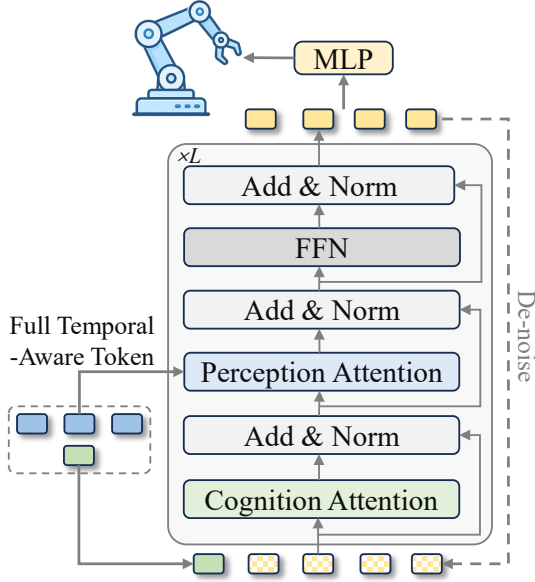


Fig. 6. Details of full temporal-aware action expert. During diffusion denoising, noisy action tokens are concatenated with cognitive token for cognition attention, while perception attention captures fine-grained details from perceptual tokens for temporally consistent action generation.

The resulting $z_{\text{img}} \in \mathbb{R}^{K \times N_q \times d_p}$ encodes compact future state evolution.

3) *Imagination Integration*: Although z_{img} captures future dynamics, it may still contain unreliable or decision-irrelevant predictions. To suppress such noise, we introduce a memory-guided imagination integration module, as illustrated in Fig. 5 (b). Specifically, the memory-augmented perceptual tokens $\tilde{p}$ query the imagined tokens z_{img} through cross-attention, followed by an FFN:

$$h = \text{FFN}(\text{CrossAttn}(\tilde{p}, z_{\text{img}}, z_{\text{img}})). \quad (19)$$

The representation h is then adaptively fused with $\tilde{p}$:

$$g = \sigma(\text{MLP}(\text{concat}[h, \tilde{p}])), \quad (20)$$

$$\bar{p} = g \odot \tilde{p} + (1 - g) \odot h, \quad (21)$$

where σ denotes the sigmoid activation and $\odot$ denotes element-wise multiplication. Finally, $\bar{p}$ and $\tilde{c}$ are combined as the full temporal-aware tokens:

$$F_{\text{temp}} = \{\bar{p}, \tilde{c}\}. \quad (22)$$

E. Full Temporal-Aware Action Expert

As illustrated in Fig. 6, the full-temporal-aware tokens $F_{\text{temp}} = \{\bar{p}, \tilde{c}\}$ are used to condition the downstream action expert for action prediction. Since robotic control lies in a continuous multimodal action space, we adopt a diffusion-based Transformer (DiT) [71] implemented with Denoising Diffusion Implicit Models (DDIM) [72]. Starting from noisy action sequences $\mathcal{A}_\tau$, the model progressively denoises them to predict the target action sequence $\mathcal{A}$.

Specifically, at denoising timestep τ , the sinusoidal timestep embedding $\text{TE}(\tau)$ is added to the cognitive token $\tilde{c}$, and the resulting representation is concatenated with the noisy action

sequence $\mathcal{A}_\tau$. A cognition-attention layer then performs self-attention over the concatenated tokens to provide high-level semantic guidance:

$$h_c = \text{CogAttn}([\tilde{c} + \text{TE}(\tau); \mathcal{A}_\tau]). \quad (23)$$

The cognition-aware features further attend to the perceptual tokens $\bar{p}$ through a perception-attention layer to inject fine-grained visual details, followed by an FFN:

$$\hat{\mathcal{A}}_0 = \text{FFN}(\text{PerAttn}(h_c, \bar{p}, \bar{p})). \quad (24)$$

The model is trained with mean squared error (MSE) loss between the predicted and target action sequences, and the final denoised representations are projected through an MLP to generate continuous 7-DoF robotic actions.

IV. EXPERIMENTS

We conduct extensive experiments to evaluate the effectiveness of MemoryVLA++. Secs. IV-B, IV-C, and IV-D report simulation results on general manipulation tasks, long-horizon temporal tasks, and robustness settings, respectively. Sec. IV-E presents real-robot results on general manipulation tasks, long-horizon memory-dependent tasks, long-horizon imagination-dependent tasks, and robustness settings. Finally, Secs. IV-F, IV-G, and IV-H provide ablation studies, analytical results, and qualitative visualizations.

A. Experimental Setup

Fig. 7 provides an overview of our simulation and real-world experiments. Overall, our experiments span 3 robots, 5 simulation benchmarks, and 3 categories of real-robot tasks, covering nearly 200 tasks with extensive variations. CogACT [3] is used as the primary baseline, and the strongest reported method is used when CogACT results are unavailable.

1) *Simulation Benchmarks*: We evaluate our method on 5 simulation benchmarks, covering general manipulation, long-horizon temporal manipulation, robustness, and generalization.

Libero [22] is a manipulation benchmark built with Franka robot. It consists of 5 suites: Spatial, Object, Goal, Long-10, and Long-90, totaling 130 tasks. Each task provides 50 demonstrations, covering general and long-horizon tasks.

SimplerEnv [23] provides real-to-sim evaluation environments for general robot manipulation. The policy is trained on BridgeData-v2 [73], a large-scale real-robot dataset containing approximately 60,000 teleoperated trajectories collected on WidowX robots across diverse tabletop scenes, and directly evaluated in simulation.

Mikasa-Robo [24] focuses on temporal robotic manipulation with Franka robot. It includes 5 memory-dependent tasks, each with 250 demonstrations.

Calvin [25] evaluates long-horizon language-conditioned manipulation, requiring the robot to complete 5 consecutive instructions by composing multiple skills. We follow the ABC→D protocol, training on environments A, B, and C and testing exclusively on environment D.

Libero-Plus [74] extends Libero with additional variations for robustness and generalization evaluation, including camera view, robot initialization, language instruction, lighting, background, noise, and layout variations.

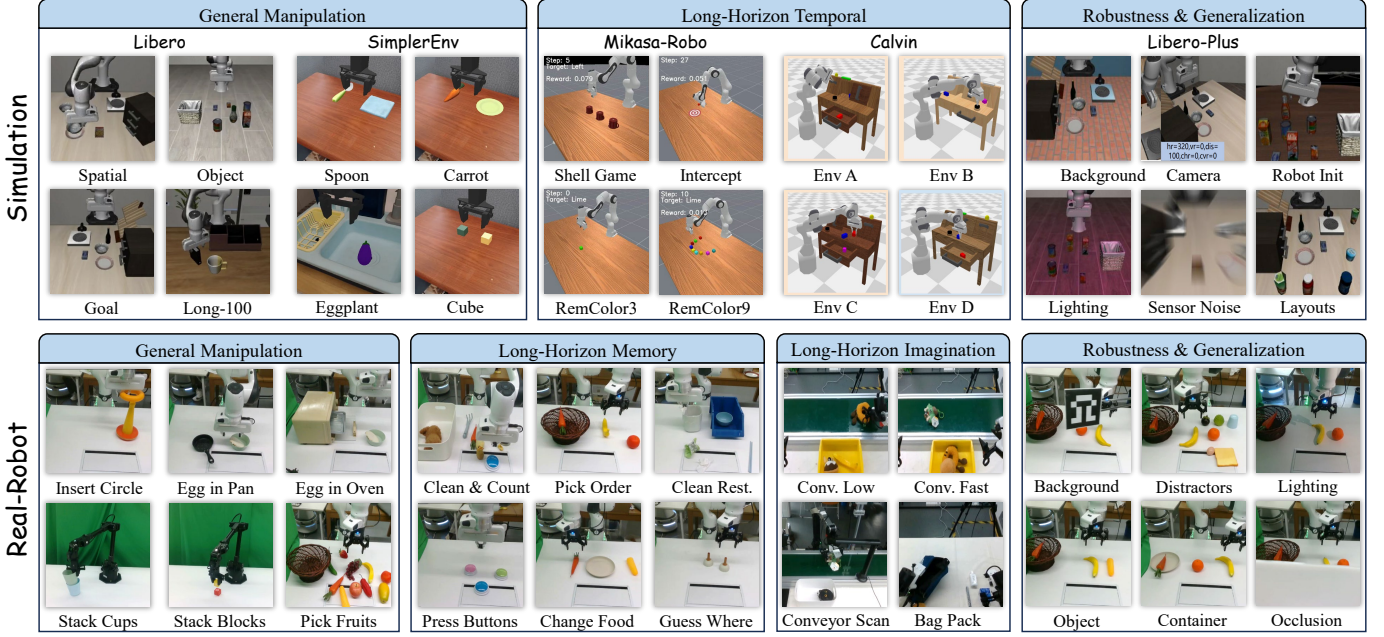


Fig. 7. Experimental setup overview. Top: simulation evaluation, covering general manipulation (Libero and SimplerEnv), long-horizon temporal manipulation (Mikasa-Robo and Calvin), and robustness & generalization evaluation (Libero-Plus). Bottom: real-robot evaluation on general tasks, long-horizon memory-dependent tasks, long-horizon imagination-dependent tasks, and robustness & generalization settings. Overall, our evaluation spans 3 robots, 5 simulation benchmarks, and 3 categories of real-robot tasks, covering nearly 200 tasks with extensive variations.

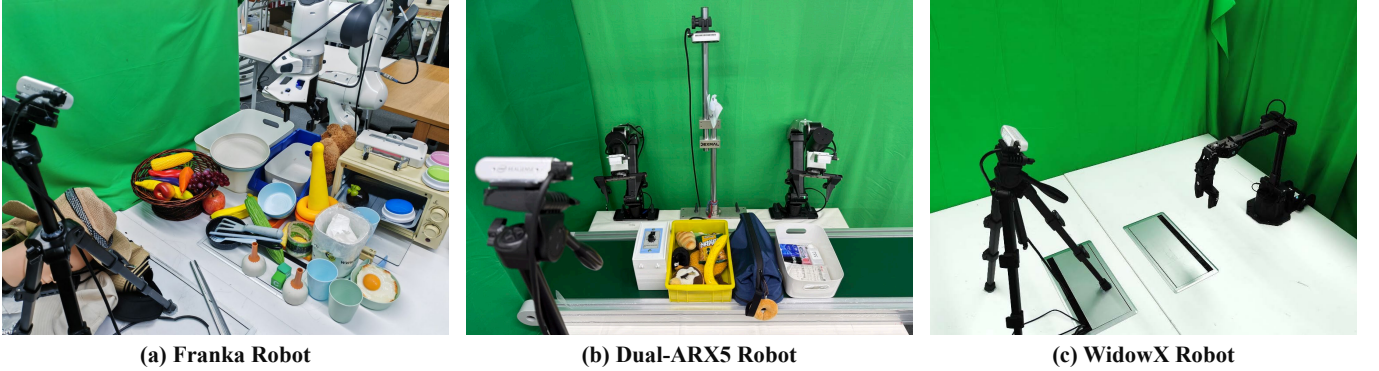


Fig. 8. Real-robot platforms used in our experiments: (a) Franka, (b) Dual-ARX5, and (c) WidowX.

2) *Real-Robot Setup*: We further evaluate our method on 3 real-robot platforms, including Franka, WidowX, and Dual-ARX5, as shown in Fig. 8. The real-robot experiments cover 3 categories: general manipulation, long-horizon memory-dependent manipulation, and long-horizon imagination-dependent manipulation. Franka and WidowX use a fixed three-view RGB setup with Intel RealSense D435 cameras, while Dual-ARX5 uses fixed RGB cameras together with an additional wrist-mounted RGB camera. RGB observations are captured at 640×480 resolution and 30 fps. Demonstrations are collected through teleoperation, and the robot system is integrated with ROS. After collection, frames are downsampled to 224×224 and temporally subsampled by retaining a frame whenever the end-effector translation since the last kept frame exceeds 0.01 m or the orientation change exceeds 0.4 rad. The processed episodes are converted into the RLDS format for downstream training.

3) *Implementation Details*: We train on 8 NVIDIA A100 or H20 GPUs with PyTorch FSDP, using 26-32 samples per GPU for a global batch of 208-256 and a learning rate of 2×10^{-5} . The LLM has 7B parameters, and the diffusion action expert contains approximately 300M parameters. During inference, we use DDIM [72] with 10 sampling steps and classifier-free guidance (CFG) [75] with a guidance scale of 1.5.

B. Simulation Evaluation on General Robotic Manipulation

1) *Training and Evaluation Details*: For Libero [22], we evaluate MemoryVLA++ across 5 suites: Spatial, Object, Goal, Long-10, and Long-90. The first 4 suites contain 10 tasks each, while Long-90 contains 90 tasks. Following OpenVLA [1], we use 50 demonstrations per task. We jointly train MemoryVLA++ on the first 4 suites for 60k steps and train Long-90 separately for 30k steps. Validation is performed every 2k steps, and results are reported at the best

TABLE I

GENERAL ROBOTIC MANIPULATION PERFORMANCE ON LIBERO [22]. SUCCESS RATES (%) ARE REPORTED ACROSS 5 SUITES. FOR METHODS WITHOUT LONG-90 RESULTS, THE AVERAGE IS COMPUTED OVER THE FIRST 4 SUITES. **BOLD** DENOTES THE BEST RESULT, AND UNDERLINE DENOTES THE BASELINE.

Method	Spatial	Object	Goal	Long-10	Long-90	Avg. Succ.
DP [36]	78.3	92.5	68.3	50.5	-	72.4
Octo [49]	78.9	85.7	84.6	51.1	-	75.1
MDT [76]	78.5	87.5	73.5	64.8	-	76.1
UniACT [77]	77.0	87.0	77.0	70.0	73.0	76.8
MaLL [78]	74.3	90.1	81.8	78.6	-	83.5
SpatialVLA [79]	88.2	89.9	78.6	55.5	46.2	71.7
TraceVLA [57]	84.6	85.2	75.1	54.1	-	74.8
OpenVLA [1]	84.7	88.4	79.2	53.7	73.5	75.9
WorldVLA [80]	87.6	96.2	83.4	60.0	-	81.8
π_0 -FAST [81]	96.4	96.8	88.6	60.2	83.1	85.0
TriVLA [82]	91.2	93.8	89.8	73.2	-	87.0
SmolVLA [83]	93.0	94.0	91.0	77.0	-	88.8
4D-VLA [53]	93.8	92.8	95.6	86.5	-	92.2
DreamVLA [84]	97.5	94.0	89.5	89.5	-	92.6
CogACT [3]	97.2	98.0	90.2	88.8	92.1	<u>93.2</u>
UniVLA [85]	96.4	98.0	90.8	89.6	-	93.7
GR00T-N1.5 [46]	94.4	97.6	93.0	90.6	-	93.9
π_0 [2]	96.8	98.8	95.8	85.2	-	94.2
MemoryVLA	98.4	98.4	96.4	93.4	95.6	96.5 (+3.3)
MemoryVLA++	99.8	100.0	98.2	96.0	97.8	98.4 (+5.2)

TABLE II

GENERAL ROBOTIC MANIPULATION PERFORMANCE ON SIMPLERENV [23]. π_0 IS REPRODUCED USING OPEN-PI-ZERO. COGACT IS EVALUATED WITH THE OFFICIAL CHECKPOINT.

Method	Spoon on Towel	Carrot on Plate	Stack Cube	Eggplant in Basket	Avg. Succ.
RT-1-X [4]	0.0	4.2	0.0	0.0	1.1
OpenVLA [1]	4.2	0.0	0.0	12.5	4.2
Octo-Base [49]	15.8	12.5	0.0	41.7	17.5
TraceVLA [57]	12.5	16.6	16.6	65.0	27.7
OpenVLA-OFT [86]	12.5	4.2	4.2	100.0	30.2
RoboVLMs [50]	45.8	20.8	4.2	79.2	37.5
SpatialVLA [79]	16.7	25.0	29.2	100.0	42.7
Magma [87]	37.5	29.2	20.8	91.7	44.8
π_0 -FAST [81]	62.5	29.2	20.8	83.3	49.0
DreamVLA [84]	45.8	45.8	25.0	87.5	51.0
VideoVLA [88]	75.0	20.8	45.8	70.8	53.1
Mimic-video [67]	41.7	54.2	29.2	100.0	56.3
CogACT [3]	58.3	45.8	29.2	95.8	<u>57.3</u>
CronusVLA [52]	66.7	54.2	20.8	100.0	60.4
GR00T-N1.5 [46]	75.3	54.3	57.0	61.3	61.9
π_0 [2]	84.6	55.8	47.9	85.4	68.4
MemoryVLA	75.0	75.0	37.5	100.0	71.9 (+14.6)
MemoryVLA++	83.3	66.7	45.8	100.0	73.9 (+16.6)

validation checkpoint. Each task is evaluated over 50 trials. For SimplerEnv [23], we train on the BridgeData-v2 dataset [73] for 50k steps, with validation performed every 2.5k steps. Results are reported at the best validation checkpoint, and each task is evaluated over 24 trials. For both benchmarks, the corresponding world model is trained for 40k steps.

2) *Results on Libero*: As shown in Tab. I, MemoryVLA++ achieves the best overall performance across all suites, with an average success rate of 98.4%. It outperforms the CogACT baseline by +5.2 points, with a particularly large gain of +7.2 points on Long-10. Across individual suites, MemoryVLA++ obtains 99.8% on Spatial, 100.0% on Object, 98.2% on Goal, 96.0% on Long-10, and 97.8% on Long-90. The gains are consistent across both general and long-horizon suites.

3) *Results on SimplerEnv*: As shown in Tab. II, MemoryVLA++ achieves an average success rate of 73.9% on Sim-

TABLE III

TEMPORAL ROBOTIC MANIPULATION PERFORMANCE ON MIKASA-ROBO [24]. NUMBERS IN PARENTHESES INDICATE THE NUMBER OF INPUT FRAMES PER FORWARD PASS. SGT, IM, RC3, RC5, AND RC9 DENOTE SHELLGAME TOUCH, INTERCEPT MEDIUM, REMEMBERCOLOR3, REMEMBERCOLOR5, AND REMEMBERCOLOR9, RESPECTIVELY.

Method	SGT	IM	RC3	RC5	RC9	Avg. Succ.
<i>Multi-frame policies</i>						
DP (4) [36]	23	-	3	-	-	-
DP (8) [36]	18	-	7	-	-	-
PTP (4) [58]	22	-	3	-	-	-
PTP (8) [58]	26	-	4	-	-	-
MaLL (4) [78]	28	-	10	-	-	-
MaLL (8) [78]	27	-	11	-	-	-
Octo (10) [49]	46	39	45	17	11	31.6
<i>Single-frame VLAs</i>						
CronusVLA (1) [52]	32	5	31	13	9	18.0
SpatialVLA (1) [79]	23	27	27	17	11	21.0
OpenVLA-OFT (1) [86]	47	14	59	16	6	28.4
π_0 (1) [2]	33	42	35	22	15	29.4
MemoryVLA (1)	88	24	44	30	20	41.2 (+11.8)
MemoryVLA++ (1)	97	40	50	19	16	44.4 (+15.0)

TABLE IV

LONG-HORIZON ROBOTIC MANIPULATION PERFORMANCE ON CALVIN [25] ABC→D SETTING. WE REPORT SUCCESS RATES OVER 1000 ROLLOUTS AND THE AVERAGE NUMBER OF COMPLETED TASKS. COGACT RESULTS FOLLOW [27].

Method	Task Success Rate					
	1	2	3	4	5	Avg. Len. ↑
<i>ABC→D Zero-Shot Setting</i>						
DP [36]	40.2	12.3	2.6	0.8	0.0	0.56
RT-1 [5]	53.3	22.2	9.4	3.8	1.3	0.90
UniPi [15]	56.0	16.0	8.0	8.0	4.0	0.92
RoboFlamingo [89]	82.4	61.9	46.6	33.1	23.5	2.47
DeerVLA [43]	89.7	70.5	51.8	44.2	35.3	2.92
GR-1 [90]	85.4	71.2	59.6	49.7	40.1	3.06
Moto [91]	89.7	72.9	60.1	48.4	38.6	3.10
OpenVLA [1]	91.3	77.8	62.0	52.1	43.5	3.27
CogACT [3]	83.8	72.9	64.0	55.9	48.0	<u>3.25</u>
OpenVLA-OFT [86]	89.1	79.4	67.4	59.8	51.5	3.47
CLOVER [92]	96.0	83.5	70.8	57.5	45.4	3.53
RoboDual [93]	94.4	82.7	72.1	62.4	54.4	3.66
UniVLA [94]	95.5	85.8	75.4	66.9	56.5	3.80
π_0 [2]	93.8	85.0	76.7	68.1	59.9	3.92
MemoryVLA	94.8	87.4	81.4	75.9	69.4	4.09 (+0.84)
MemoryVLA++	95.6	90.2	85.7	81.7	76.1	4.29 (+1.04)

plerEnv, outperforming CogACT by +16.6 points. It obtains 83.3%, 66.7%, 45.8%, and 100.0% on *Spoon on Towel*, *Carrot on Plate*, *Stack Cube*, and *Eggplant in Basket*, respectively.

C. Simulation Evaluation on Long-Horizon Temporal Tasks

1) *Training and Evaluation Details*: We evaluate long-horizon temporal tasks on Mikasa-Robo [24] and Calvin [25]. For Mikasa-Robo, we follow the standard protocol with 250 demonstrations per task, 128×128 image observations, end-effector control, and 100 evaluation episodes per task. All 5 tasks are jointly trained for 20k steps, validated every 1k steps, and reported at the best validation checkpoint. The world model for Mikasa-Robo is trained for 20k steps. As shown in Tab. III, we compare with both multi-frame policies and single-frame VLAs. For Calvin, we use the ABC→D zero-shot setting, where policies are trained on environments A, B,

TABLE V

ROBUSTNESS & GENERALIZATION PERFORMANCE ON LIBERO-PLUS [74]. OPENVLA-OFT AND OUR METHOD ARE BOTH JOINTLY TRAINED ON ALL SUITES AND USE WRIST IMAGE. “SUPERVISED FINE-TUNING” INCLUDES LIBERO-PLUS DATA IN MIXED TRAINING. AVG. SUCC. IS COMPUTED OVER ALL TRIALS ACROSS SETTINGS, RATHER THAN BY AVERAGING SETTING-LEVEL SUCCESS RATES.

Method	Camera	Robot	Language	Light	Background	Noise	Layout	Avg. Succ.
<i>Zero-Shot Setting</i>								
OpenVLA [1]	0.8	3.5	23.0	8.1	34.8	15.2	28.5	15.6
WorldVLA [80]	0.1	27.9	41.6	43.7	17.1	10.9	38.0	25.0
DP [36]	1.6	32.3	77.0	25.0	19.8	20.3	42.2	31.7
NORA [95]	2.2	37.0	65.1	45.7	58.6	12.8	62.1	39.0
UniVLA [94]	1.8	46.2	69.6	69.0	81.0	21.2	31.9	43.9
Fast-WAM [96]	16.4	44.5	68.9	78.2	53.7	37.7	60.7	51.5
π_0 [2]	13.8	6.0	58.8	85.0	81.4	79.0	68.9	53.6
π_0 -Fast [81]	65.1	21.6	61.0	73.2	73.2	74.4	68.8	61.6
OpenVLA-OFT [86]	55.6	21.7	81.0	92.7	91.0	78.6	68.7	67.9
RIPT-VLA [97]	55.2	31.2	77.6	88.4	91.6	73.5	74.2	68.4
MemoryVLA	42.7	44.9	84.4	92.8	95.0	62.1	84.7	70.2
MemoryVLA++	36.4	68.9	88.7	93.8	90.6	63.5	83.8	73.1
<i>Supervised Fine-Tuning Setting</i>								
π_0 [2]	79.6	21.1	72.5	84.7	86.2	68.3	69.4	67.4
OpenVLA-OFT [86]	92.8	30.3	85.8	94.9	93.9	89.3	77.6	<u>79.6</u>
MemoryVLA	91.4	48.6	79.4	95.2	95.3	94.0	75.7	81.9
MemoryVLA++	96.8	49.7	71.0	96.6	97.0	96.0	78.6	82.7

and C, and evaluated on the unseen environment D. Models are trained for 60k steps, evaluated every 2k steps, and reported at the best evaluation checkpoint. For the world model, we directly follow VPP [66]. We report success rates over 1000 rollouts and the average number of completed tasks.

2) *Results on Mikasa-Robo*: As shown in Tab. III, MemoryVLA++ achieves the best average success rate of 44.4%, outperforming the previous best single-frame VLA by +15.0 points. Notably, MemoryVLA++ improves ShellGameTouch by +50.0 points over the previous best single-frame VLA, demonstrating the effectiveness of full temporal modeling.

3) *Results on Calvin*: As shown in Tab. IV, MemoryVLA++ achieves the best average completed task length of 4.29, improving over CogACT baseline by +1.04. It also achieves higher success rates across all 5 task steps, with larger gains at longer horizons, showing improved temporal consistency in long-horizon task execution.

D. Simulation Evaluation on Robustness and Generalization

1) *Training and Evaluation Details*: We evaluate robustness and generalization on Libero-Plus [74]. Libero-Plus introduces 7 out-of-distribution variations, covering camera, robot, language, lighting, background, noise, and layout. In the zero-shot setting, models are trained on the 4 standard Libero suites, including Spatial, Object, Goal, and Long-10. We jointly train on these 4 suites for 60k steps and evaluate on Libero-Plus without using its training data. In the supervised fine-tuning setting, Libero-Plus data are included in mixed training, and models are trained for the same 60k steps. The world model is trained for 40k steps. For both settings, evaluation is performed every 10k steps, and we report the best evaluation result.

2) *Results on Libero-Plus*: As shown in Tab. V, MemoryVLA++ achieves an average success rate of 73.1% in the zero-shot setting, outperforming OpenVLA-OFT by +5.2 points. In the supervised fine-tuning setting, MemoryVLA++

further improves the average success rate to 82.7%, exceeding OpenVLA-OFT by +3.1 points. These results indicate that MemoryVLA++ maintains robustness and generalization under diverse distribution shifts.

E. Real-Robot Evaluation

1) *Real-Robot Tasks*: We evaluate real-robot performance across 3 task categories, as shown in Fig. 11. The general tasks evaluate short-horizon manipulation skills, including *Insert Circle*, *Egg in Pan*, *Egg in Oven*, *Stack Cups*, *Stack Blocks*, and *Pick Diverse Fruits*. The long-horizon memory-dependent tasks require the policy to use historical information over multiple sub-goals, including *Push Buttons*, *Change Food*, *Guess Where*, *Clean Table & Count*, *Pick Place Order*, and *Clean Restaurant Table*. The long-horizon imagination-dependent tasks require anticipating future state evolution, including *Conveyor Pick-Low*, *Conveyor Pick-Mid*, *Conveyor Pick-High*, *Conveyor Scan-Pick*, and *Bag Pack & Zip*.

2) *Training and Evaluation Details*: All tasks are evaluated from randomized initial states. For general tasks, each task uses 50-150 demonstrations. *Pick Diverse Fruits* contains 5 variants with 5 trials per variant, resulting in 25 trials in total, while other general tasks use 15 trials. For long-horizon memory-dependent and imagination-dependent tasks, each task uses 200-300 demonstrations. *Push Buttons* contains 3 variants with 5 trials per variant, resulting in 15 trials in total, while other long-horizon tasks use 10 trials. We use step-wise scoring for these long-horizon tasks to better reflect progress over multiple sub-goals. Training runs for approximately 5k-30k steps depending on the task. For MemoryVLA++, the world model is trained for 20k steps. MemoryVLA++ is evaluated on long-horizon imagination-dependent tasks, where future imagination is explicitly required.

3) *Results on General Tasks*: As shown in Tab. VI(a), MemoryVLA achieves an average success score of 85% on

TABLE VI

REAL-ROBOT PERFORMANCE ACROSS THREE TASK CATEGORIES. SUCCESS SCORES (%) ARE REPORTED. MEMORYVLA++ IS EVALUATED ONLY ON IMAGINATION-DEPENDENT TASKS, WHERE FUTURE IMAGINATION IS EXPLICITLY REQUIRED.

Method	Insert Circle	Egg in Pan	Egg in Oven	Stack Cups	Stack Blocks	Pick Diverse Fruits	Avg. Succ.
OpenVLA [1]	47	27	53	40	13	4	31
π_0 [2]	67	73	73	87	53	80	72
CogACT [3]	80	67	60	93	80	76	76
MemoryVLA	87	80	80	93	87	84	85 (+9)

(a) General manipulation tasks.

Method	Push Buttons	Change Food	Guess Where	Clean Table & Count	Pick Place Order	Clean Rest. Table	Avg. Succ.
OpenVLA [1]	6	3	0	15	27	0	9
π_0 [2]	25	42	24	61	82	80	52
CogACT [3]	15	47	40	67	90	84	57
MemoryVLA	58	85	72	84	100	96	83 (+26)

(b) Long-horizon memory-dependent tasks.

Method	Conveyor Pick-Low	Conveyor Pick-Mid	Conveyor Pick-High	Conveyor Scan-Pick	Bag Pack & Zip	Avg. Succ.
π_0 [2]	56	52	44	50	41	49
CogACT [3]	61	55	48	42	37	49
MemoryVLA	75	68	58	63	60	65 (+16)
MemoryVLA++	84	80	72	75	73	77 (+28)

(c) Long-horizon imagination-dependent tasks.

real-robot general tasks, outperforming CogACT by +9 points. It improves over the strong baseline on all 6 tasks, with notable gains on *Egg in Pan* (+13) and *Egg in Oven* (+20).

4) Results on Long-Horizon Memory-Dependent Tasks:

As shown in Tab. VI(b), MemoryVLA achieves an average success score of 83% on long-horizon memory-dependent tasks, exceeding CogACT by +26 points. The improvements are larger than those on general tasks, including +43 on *Seq. Push Buttons*, +38 on *Change Food*, +32 on *Guess Where*, and +17 on *Clean Table & Count*. These results indicate that explicit memory modeling is particularly beneficial for real-world tasks requiring long-term temporal context.

5) Results on Long-Horizon Imagination-Dependent Tasks:

As shown in Tab. VI(c), MemoryVLA++ achieves an average success score of 77% on long-horizon imagination-dependent tasks, outperforming CogACT by +28 points. Compared with MemoryVLA, MemoryVLA++ further improves the average score by +12 points. The improvements are especially large on *Bag Pack & Zip* (+36) and *Conveyor Scan-Pick* (+33). These results show that future imagination improves real-world tasks requiring future-state anticipation.

6) *Results on Robustness and Generalization:* We further evaluate real-world robustness and generalization under diverse out-of-distribution conditions. As shown in Fig. 9, we test two long-horizon tasks, *Pick Place Order* and *Clean Restaurant Table*, under variations in backgrounds, distractors, lighting, objects, containers, and occlusions. The model shows only minor performance drops across these variations, indicat-

TABLE VII

ABLATION STUDIES ON MEMORY MODELING. FOR MEMORY LENGTH, SMALL/DEFAULT/LARGE DENOTE 4/16/64 FOR SIMPLERENV, 8/16/32 FOR LIBERO-LONG-90, AND 64/256/512 FOR REAL-TEMPORAL. REAL-TEMPORAL REFERS TO THE CLEAN TABLE & COUNT TASK. THE DEFAULT SETTING IS HIGHLIGHTED WITH GRAY.

Length	SimplerEnv	Long-90	Real-Temporal
Small	67.7	94.2	78
Default	71.9	95.6	84
Large	67.7	95.6	81

(a) Memory length.

Fusion	SimplerEnv	Long-90	Real-Temporal
Add	67.7	93.8	78
Gate	71.9	95.6	84

(b) Memory fusion strategy.

Consolidation	SimplerEnv	Long-90	Real-Temporal
FIFO	66.7	94.9	76
Token Merge	71.9	95.6	84

(c) Memory consolidation strategy.

Retrieval	SimplerEnv	Memory Type	SimplerEnv
w/o Timestep PE	69.8	Cognitive Memory	63.5
w/ Timestep PE	71.9	Perceptual Memory	64.6
		Both	71.9

(d) Memory retrieval strategy.

(e) Memory type.

ing strong robustness to real-world distribution shifts.

F. Ablation Studies

1) *Ablation Studies On Memory Modeling:* We conduct ablations based on MemoryVLA to analyze the key designs of memory modeling in Tab. VII. As shown in Tab. VII(a), the default memory length achieves the best overall performance across SimplerEnv, Libero-Long-90, and the real-world temporal task, showing the importance of balancing temporal coverage and memory redundancy. As shown in Tab. VII(b), gated fusion consistently outperforms addition-based fusion, demonstrating the benefit of adaptive memory integration. For memory consolidation, Token Merge consistently improves over FIFO, confirming the benefit of redundancy-aware memory compression, as shown in Tab. VII(c). As shown in Tab. VII(d), timestep positional encoding further improves memory retrieval by preserving temporal order. As shown in Tab. VII(e), combining perceptual and cognitive memory performs best, outperforming either memory type alone by a clear margin. Overall, these ablations confirm the effectiveness of the proposed temporal memory modeling design.

2) *Ablation Studies On Imagination Modeling:* We conduct ablations based on MemoryVLA++ to analyze the key designs of imagination modeling in Tab. VIII. As shown in Tab. VIII(a), using a single denoising step already achieves competitive performance, while more steps bring little gain and increase inference cost. As shown in Tab. VIII(b), a longer

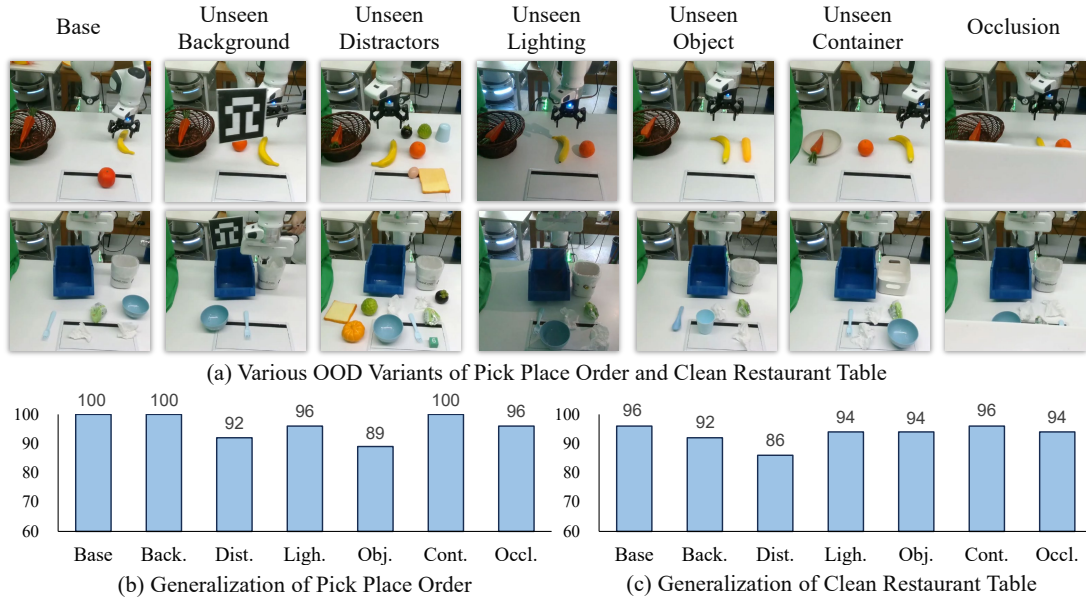


Fig. 9. Real-world robustness and generalization under OOD conditions. Our method maintains strong performance across diverse variations.

TABLE VIII
ABLATION STUDIES ON IMAGINATION MODELING. SUCCESS RATES (%) ARE REPORTED ON MIKASA-ROBO. THE DEFAULT SETTING IS HIGHLIGHTED WITH GRAY.

Denoise Step	Avg. Succ.	Horizon	Avg. Succ.
1	44.4	4	43.4
3	44.6	8	43.8
5	43.6	16	44.4

(a) Denoising step.

(b) Imagination horizon.

WM Update	Avg. Succ.	Integration	Avg. Succ.
w/o Freeze	42.8	Add	41.2
w/ Freeze	44.4	Mem-Guided	44.4

(c) World-model update.

(d) Integration strategy.

TABLE IX
EVALUATION METRICS FOR SIMULATION AND REAL-ROBOT DATASETS. PSNR AND SSIM ARE HIGHER-IS-BETTER ($\uparrow$), WHILE LPIPS, FVD, AND EPE ARE LOWER-IS-BETTER ($\downarrow$).

Datasets	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FVD $\downarrow$	EPE $\downarrow$
Libero	20.26	0.820	0.182	101.93	0.5104
Bridge	17.44	0.712	0.276	132.31	1.4999
Mikasa-Robo	26.39	0.838	0.174	189.38	0.1540
Calvin	22.22	0.833	0.185	29.69	0.2049
Real-Conv. Pick	16.91	0.764	0.250	128.18	1.6276
Real-Conv. Scan	22.34	0.822	0.182	82.92	0.4163
Real-Bag Pack	16.94	0.768	0.266	70.56	1.7672
Average	20.36	0.794	0.216	105.00	0.8829

imagination horizon improves performance, suggesting that future temporal cues are useful for long-horizon decision-making. For world-model update, freezing the world model outperforms updating it during policy training, as shown in Tab. VIII(c), indicating that the pretrained visual dynamics prior should be preserved. As shown in Tab. VIII(d), memory-guided integration clearly outperforms direct addition, showing the importance of selecting decision-relevant imagined cues through memory context. Overall, these ablations confirm the effectiveness of the proposed imagination modeling design.

G. Analytical Results

1) *Analysis of Memory Mechanism:* To provide a direct view of how the memory mechanism functions, Fig. 10 visualizes the retrieved memory elements and their attention weights on the real-world and simulation tasks. The model consistently attends to past frames that resolve decision-

relevant ambiguities absent from the current observation. In the real-world *Change Food* task, after the first food item is placed aside, the current frame contains two food items on the table, making it impossible to determine from this single observation which one should be picked next. Our method therefore attends strongly to the nearby frames reflecting the recent motion trend, as well as the last decisive frame before the ambiguity arises. In the *Shell Game Touch* task from Mikasa-Robo Simulation Benchmark, the robot is briefly shown the cube location before it is covered by cups. The model consistently attends to the initial revealing frames, which provide the only reliable cue for identifying the correct cup. These results demonstrate that our method retrieves meaningful temporal cues essential for disambiguating the next action, rather than simply recalling redundant visual history.

2) *Analysis of Imagination Mechanism:* As shown in Fig. 12, we visualize the imagined future trajectories from the world model and compare them with ground-truth future observations. Across both simulation and real-world tasks, the generated trajectories preserve the main semantic evolution of the scene. Notably, using only 1 denoising step already captures meaningful future dynamics, supporting our latent-

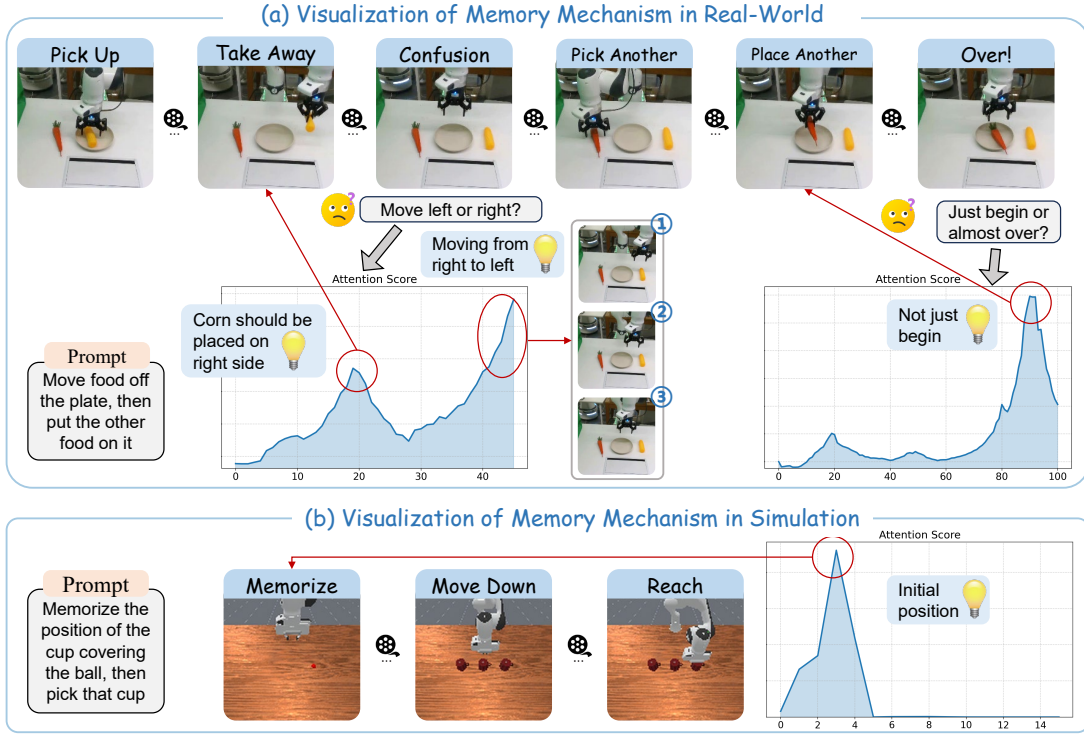


Fig. 10. Analysis of memory modeling in real-world and simulation tasks. We visualize the retrieved memory elements and their attention weights in the real-world *Change Food* task and the simulated *Shell Game Touch* task, showing that the model attends to decision-relevant past frames that resolve ambiguities absent from the current observation.

TABLE X
INFERENCE EFFICIENCY COMPARISON. LATENCY, THROUGHPUT, AND GPU MEMORY ARE MEASURED OVER 300 RUNS IN BFLOAT16 USING SINGLE-VIEW INPUTS AND AN ACTION CHUNK LENGTH OF 16 ON RTX 4090 AND HGX H20 GPUS.

Method	Latency (RTX 4090)	Throughput (RTX 4090)	Latency (H20)	Throughput (H20)	GPU Memory
Baseline	0.187 s	85.6 Hz	0.236 s	67.8 Hz	15.8 GB
MemoryVLA	0.194 s	82.5 Hz	0.246 s	65.0 Hz	16.6 GB
MemoryVLA++	0.241 s	66.4 Hz	0.301 s	53.2 Hz	21.7 GB

space imagination design for efficient robotic decision-making.

Table IX quantitatively evaluates imagination quality across simulation and real-robot datasets under the full-denoising setting. PSNR [98] measures pixel-level fidelity, SSIM [99] measures frame-level structural similarity, and LPIPS [100] measures perceptual similarity with deep features. FVD [101] evaluates video-level distributional quality, while Flow-EPE [102] measures motion consistency using optical-flow endpoint error. The results show that the world model captures future semantic and motion dynamics.

3) *Analysis of Inference Efficiency*: As shown in Tab. X, we compare inference latency, throughput, and GPU memory with the baseline on RTX 4090 and HGX H20 GPUs. All measurements are averaged over 300 runs in bfloat16 using single-view inputs and an action chunk length of 16. MemoryVLA introduces only minor overhead. On RTX 4090, the latency only increases from 0.187 s to 0.194 s, corresponding to only a $\sim 4\%$ increase, while the throughput remains close to the baseline, decreasing from 85.6 Hz to 82.5 Hz. GPU memory increases by only 0.8 GB. These results show that the memory module is lightweight and introduces negligible

extra cost during inference. MemoryVLA++ further introduces latent-space imagination during inference. Its latency increases to 0.241 s on RTX 4090 and 0.301 s on H20, while still maintaining 66.4 Hz and 53.2 Hz throughput, respectively. This moderate overhead comes from the additional world-model forward pass, but remains practical for real-robot deployment.

4) *Analysis of Stronger VLA Pretraining*: As shown in Tab. XI, replacing the LLaMA2-based VLA with a Qwen2.5-based VLA and Dexbotic pretraining improves the average success rate from 71.9% to 84.4% on SimplerEnv. On Libero, the two settings achieve comparable performance, with Qwen2.5 and Dexbotic pretraining slightly improving the average success rate from 96.7% to 97.0%. These results indicate that stronger VLA pretraining can further improve performance, especially on more challenging tasks.

H. Visualization Analysis

Fig. 11 shows real-robot tasks across general manipulation, long-horizon memory-dependent tasks, and long-horizon imagination-dependent tasks.



Fig. 11. Visualization of real-robot tasks across general manipulation, long-horizon memory-dependent tasks, and long-horizon imagination-dependent tasks.



Fig. 12. Visualization of world model imagination. We compare ground-truth future observations with imagined trajectories generated using 30 and 1 denoising steps across simulation and real-world tasks, showing that the world model captures future semantic dynamics even under partial denoising.

TABLE XI
ANALYSIS OF VLA BACKBONE AND PRETRAINING.

Backbone	Pretraining	Spoon Towel	Carrot Plate	Stack Cube	Eggplant Basket	Avg. Succ.
LLaMA2	CogACT	75.0	75.0	37.5	100.0	71.9
Qwen2.5	Dexbotic	100.0	66.7	70.8	100.0	84.4

(a) Performance on SimplerEnv.

Backbone	Pretraining	Spatial	Object	Goal	Long-10	Avg. Succ.
LLaMA2	CogACT	98.4	98.4	96.4	93.4	96.7
Qwen2.5	Dexbotic	97.2	99.2	98.4	93.2	97.0

(b) Performance on Libero.

V. CONCLUSION

We propose MemoryVLA++, a full temporal modeling framework that equips VLA models with memory and imagination for robotic manipulation. MemoryVLA++ leverages the commonsense priors of pretrained VLMs for high-level cognition. Inspired by cognitive science, it uses a hippocampus-like Perceptual-Cognitive Memory Bank that cooperates with working memory to capture past temporal dependencies. Meanwhile, a world model generates latent-space future imagination, and a full temporal-aware action expert uses memory and imagination to produce temporally consistent actions. Extensive experiments on 5 simulation benchmarks and 3 categories of real-robot tasks across 3 robots, covering nearly 200 tasks with diverse variations, demonstrate the effectiveness of MemoryVLA++. The improvements are especially clear

on long-horizon temporal tasks, validating the importance of modeling past memory and future imagination for robotic manipulation. MemoryVLA++ represents an encouraging step toward temporal modeling in generalist robot policies.

REFERENCES

- [1] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. P. Foster, P. R. Sanketi, Q. Vuong *et al.*, “Openvla: An open-source vision-language-action model,” in *Conference on Robot Learning*. PMLR, 2025, pp. 2679–2713.
- [2] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter *et al.*, “pi-0: A vision-language-action flow model for general robot control,” in *Proceedings of Robotics: Science and Systems*, 2025.
- [3] Q. Li, Y. Liang, Z. Wang, L. Luo, X. Chen, M. Liao, F. Wei, Y. Deng, S. Xu, Y. Zhang *et al.*, “Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation,” *arXiv preprint arXiv:2411.19650*, 2024.
- [4] A. O’Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain *et al.*, “Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0,” in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 6892–6903.
- [5] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu *et al.*, “Rt-1: Robotics transformer for real-world control at scale,” *Robotics: Science and Systems XIX*, 2023.
- [6] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis *et al.*, “Droid: A large-scale in-the-wild robot manipulation dataset,” *arXiv preprint arXiv:2403.12945*, 2024.
- [7] Q. Bu, J. Cai, L. Chen, X. Cui, Y. Ding, S. Feng, X. He, X. Huang *et al.*, “Agibot world colosseum: A large-scale manipulation platform for scalable and intelligent embodied systems,” in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025.

- [8] S. Karamcheti, S. Nair, A. Balakrishna, P. Liang, T. Kollar, and D. Sadigh, "Prismatic vlms: Investigating the design space of visually-conditioned language models," in *Forty-first International Conference on Machine Learning*, 2024.
- [9] J. Bai, S. Bai, Y. Chu, Z. Cui, K. Dang, X. Deng, Y. Fan, W. Ge, Y. Han, F. Huang *et al.*, "Qwen technical report," *arXiv preprint arXiv:2309.16609*, 2023.
- [10] C. Liu, J. Zhang, C. Li, Z. Zhou, S. Wu, S. Huang, and H. Duan, "Tf-vla: Temporal token fusion via pixel-attention integration for vision-language-action models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, no. 22, 2026, pp. 18 452–18 459.
- [11] H. Jang, S. Yu, H. Kwon, H. Jeon, Y. Seo, and J. Shin, "Contextvla: Vision-language-action model with amortized multi-frame context," *arXiv preprint arXiv:2510.04246*, 2025.
- [12] X. Wang, X. Gao, J. Fu, Z. Li, D. Fortier, G. Mullins, A. Kolobov, and B. Guo, "Lola: Long horizon latent action learning for general robot manipulation," *arXiv preprint arXiv:2512.20166*, 2025.
- [13] Z. Zhang, S. Yang, Q. Hu, L. J. Huang, J. Hou, Y. Sun, Y. Lu, and S. Han, "Foreact: Steering your vla with efficient visual foresight planning," *arXiv preprint arXiv:2602.12322*, 2026.
- [14] Q. Long, Y. Wang, J. Song, J. Zhang, P. Li, W. Wang, Y. Wang, H. Li, S. Xie, G. Yao *et al.*, "Scaling world model for hierarchical manipulation policies," *arXiv preprint arXiv:2602.10983*, 2026.
- [15] Y. Du, S. Yang, B. Dai, H. Dai, O. Nachum, J. Tenenbaum, D. Schuurmans, and P. Abbeel, "Learning universal policies via text-guided video generation," *Advances in neural information processing systems*, vol. 36, pp. 9156–9172, 2023.
- [16] K. Black, M. Nakamoto, P. Atreya, H. Walke, C. Finn, A. Kumar, and S. Levine, "Zero-shot robotic manipulation with pre-trained image-editing diffusion models," in *International Conference on Learning Representations*, vol. 2024, 2024, pp. 33 431–33 452.
- [17] A. D. Baddeley and G. J. Hitch, "Working memory (vol. 8)," *New York: GA Bower (ed), Recent advances in learning and motivation*, 1974.
- [18] E. Tulving *et al.*, "Episodic and semantic memory," *Organization of memory*, vol. 1, no. 381–403, p. 1, 1972.
- [19] V. F. Reyna and C. J. Brainerd, "Fuzzy-trace theory: An interim synthesis," *Learning and individual Differences*, vol. 7, no. 1, pp. 1–75, 1995.
- [20] K. J. W. Craik, *The nature of explanation*. CUP Archive, 1967, vol. 445.
- [21] R. Grush, "The emulation theory of representation: Motor control, imagery, and perception," *Behavioral and brain sciences*, vol. 27, no. 3, pp. 377–396, 2004.
- [22] B. Liu, Y. Zhu, C. Gao, Y. Feng, Q. Liu, Y. Zhu, and P. Stone, "Libero: Benchmarking knowledge transfer for lifelong robot learning," *Advances in Neural Information Processing Systems*, vol. 36, pp. 44 776–44 791, 2023.
- [23] X. Li, K. Hsu, J. Gu, O. Mees, K. Pertsch, H. R. Walke, C. Fu, I. Lunawat, I. Sieh, S. Kirmani *et al.*, "Evaluating real-world robot manipulation policies in simulation," in *Conference on Robot Learning*. PMLR, 2025, pp. 3705–3728.
- [24] E. Cherepanov, N. Kachaev, A. K. Kovalev, and A. I. Panov, "Memory, benchmark & robots: A benchmark for solving complex tasks with reinforcement learning," *arXiv preprint arXiv:2502.10550*, 2025.
- [25] O. Mees, L. Hermann, E. Rosete-Beas, and W. Burgard, "Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7327–7334, 2022.
- [26] H. Shi, B. Xie, Y. Liu, L. Sun, F. Liu, T. Wang, E. Zhou, H. Fan, X. Zhang, and G. Huang, "Memoryvla: Perceptual-cognitive memory in vision-language-action models for robotic manipulation," *arXiv preprint arXiv:2508.19236*, 2025.
- [27] B. Xie, E. Zhou, F. Jia, H. Shi, H. Fan, H. Zhang, H. Li, J. Sun, J. Bin, J. Huang *et al.*, "Dexbotic: Open-source vision-language-action toolbox," *arXiv preprint arXiv:2510.23511*, 2025.
- [28] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby *et al.*, "Dinov2: Learning robust visual features without supervision," *Transactions on Machine Learning Research Journal*, 2024.
- [29] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer, "Sigmoid loss for language image pre-training," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 11 975–11 986.
- [30] H. Zheng, H. Shi, Y. X. Chng, R. Huang, Z. Ni, T. Tan, Q. Peng, Y. Weng, Z. Shi, and G. Huang, "Denseg: Alleviating vision-language feature sparsity in multi-view 3d visual grounding," in *Autonomous Grand Challenge CVPR 2024 Workshop*, vol. 2, 2024, p. 6.
- [31] H. Zheng, H. Shi, Q. Peng, Y. X. Chng, R. Huang, Y. Weng, zhongchao shi, and G. Huang, "Densegrounding: Improving dense language-vision semantics for ego-centric 3d visual grounding," in *The Thirteenth International Conference on Learning Representations*, 2025.
- [32] Y. Zhang, D. Wu, H. Shi, Y. Liu, T. Wang, H. Fan, and X. Dong, "Grounding beyond detection: Enhancing contextual understanding in embodied 3d grounding," *arXiv preprint arXiv:2506.05199*, 2025.
- [33] Y. Wang, Y. Yue, Y. Yue, H. Wang, H. Jiang, Y. Han, Z. Ni, Y. Pu, M. Shi, R. Lu *et al.*, "Emulating human-like adaptive vision for efficient and flexible machine visual perception," *Nature Machine Intelligence*, pp. 1–19, 2025.
- [34] G. Huang, Y. Wang, K. Lv, H. Jiang, W. Huang, P. Qi, and S. Song, "Glance and focus networks for dynamic visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 4, pp. 4605–4621, 2022.
- [35] Y. Wang, H. Zhang, Y. Yue, S. Song, C. Deng, J. Feng, and G. Huang, "Uni-adafocus: Spatial-temporal dynamic computation for video recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 47, no. 3, pp. 1782–1799, 2024.
- [36] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," *The International Journal of Robotics Research*, p. 02783649241273668, 2023.
- [37] T. Zhao, V. Kumar, S. Levine, and C. Finn, "Learning fine-grained bimanual manipulation with low-cost hardware," *Robotics: Science and Systems XIX*, 2023.
- [38] A. Goyal, J. Xu, Y. Guo, V. Blukis, Y.-W. Chao, and D. Fox, "Rvt: Robotic view transformer for 3d object manipulation," in *Conference on Robot Learning*. PMLR, 2023, pp. 694–710.
- [39] H. Shi, B. Xie, Y. Liu, Y. Yue, T. Wang, H. Fan, X. Zhang, and G. Huang, "Spatialactor: Exploring disentangled spatial representations for robust robotic manipulation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, no. 11, 2026, pp. 8969–8977.
- [40] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altschmidt, S. Altman, S. Anadkat *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.
- [41] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.
- [42] J. Liu, H. Chen, Z. Liu, P. An, R. Zhang, C. Gu, X. Li, Z. Guo, S. Chen, M. Liu, C. Hou, M. Zhao, K. alex Zhou, P.-A. Heng, and S. Zhang, "HybridVLA: Collaborative diffusion and autoregression in a unified vision-language-action model," in *The Fourteenth International Conference on Learning Representations*, 2026.
- [43] Y. Yue, Y. Wang, B. Kang, Y. Han, S. Wang, S. Song, J. Feng, and G. Huang, "Deer-vla: Dynamic inference of multimodal large language models for efficient robot execution," *Advances in Neural Information Processing Systems*, vol. 37, pp. 56 619–56 643, 2024.
- [44] L. Sun, B. Xie, Y. Liu, H. Shi, T. Wang, and J. Cao, "Geovla: Empowering 3d representations in vision-language-action models," *arXiv preprint arXiv:2508.09071*, 2025.
- [45] B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," in *Conference on Robot Learning*. PMLR, 2023, pp. 2165–2183.
- [46] J. Bjorck, F. Castañeda, N. Cherniadev, X. Da, R. Ding, L. Fan, Y. Fang, D. Fox, F. Hu, S. Huang *et al.*, "Gr00t n1: An open foundation model for generalist humanoid robots," *arXiv preprint arXiv:2503.14734*, 2025.
- [47] S. Wang, Y. Liu, T. Wang, Y. Li, and X. Zhang, "Exploring object-centric temporal modeling for efficient multi-view 3d object detection," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 3621–3631.
- [48] T. Feng, H. Shi, X. Liu, W. Feng, L. Wan, Y. Zhou, and D. Lin, "Open compound domain adaptation with object style compensation for semantic segmentation," *Advances in Neural Information Processing Systems*, vol. 36, pp. 63 136–63 149, 2023.
- [49] Octo Model Team, D. Ghosh, H. Walke, K. Black, O. Mees, S. Dasari, J. Hejna, C. Xu, J. Luo, T. Kreiman, Y. Tan, L. Y. Chen, P. Sanketi, Q. Vuong, T. Xiao, D. Sadigh, C. Finn, and S. Levine, "Octo: An open-source generalist robot policy," in *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, 2024.
- [50] X. Li, P. Li, L. Qian, M. Liu, D. Wang, J. Liu, B. Kang, X. Ma, X. Wang, D. Guo *et al.*, "What matters in building vision-language-action models for generalist robots," *Nature Machine Intelligence*, pp. 1–15, 2026.

- [51] C. Fan, X. Jia, Y. Sun, Y. Wang, J. Wei, Z. Gong, X. Zhao, M. Tomizuka, X. Yang, J. Yan *et al.*, “Interleave-vla: Enhancing robot manipulation with interleaved image-text instructions,” in *1st Workshop on Safely Leveraging Vision-Language Foundation Models in Robotics: Challenges and Opportunities*, 2025.
- [52] H. Li, S. Yang, Y. Chen, Y. Tian, X. Yang, X. Chen, H. Wang, T. Wang, F. Zhao, D. Lin *et al.*, “Cronusvla: Transferring latent motion across time for multi-frame prediction in manipulation,” *arXiv preprint arXiv:2506.19816*, 2025.
- [53] J. Zhang, Y. Chen, Y. Xu, Z. Huang, Y. Zhou, Y.-J. Yuan, X. Cai, G. Huang, X. Quan, H. Xu *et al.*, “4d-vla: Spatiotemporal vision-language-action pretraining with cross-scene calibration,” *Advances in Neural Information Processing Systems*, vol. 38, pp. 33 914–33 937, 2026.
- [54] M. Koo, D. Choi, T. Kim, K. Lee, C. Kim, Y. Seo, and J. Shin, “HAMLET: Switch your vision-language-action model into a history-aware policy,” in *The Fourteenth International Conference on Learning Representations*, 2026.
- [55] H. Fang, M. Grotz, W. Pumacay, Y. R. Wang, D. Fox, R. Krishna, and J. Duan, “Sam2act: Integrating visual foundation model with a memory architecture for robotic manipulation,” in *International Conference on Machine Learning*. PMLR, 2025, pp. 15 925–15 942.
- [56] M. Lin, P. Ding, S. Wang, Z. Zhuang, Y. Liu, X. Tong, W. Song, S. Lyu, S. Huang, and D. Wang, “Hif-vla: Hindsight, insight and foresight through motion representation for vision-language-action models,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026, pp. 20 732–20 742.
- [57] R. Zheng, Y. Liang, S. Huang, J. Gao, H. Daumé III, A. Kolobov, F. Huang, and J. Yang, “Tracevla: Visual trace prompting enhances spatial-temporal awareness for generalist robotic policies,” in *International Conference on Learning Representations*, vol. 2025, 2025, pp. 54 277–54 296.
- [58] M. T. Villasevil, A. Tang, Y. Liu, and C. Finn, “Learning long-context diffusion policies via past-token prediction,” in *9th Annual Conference on Robot Learning*, 2025.
- [59] A. Sridhar, J. Pan, S. Sharma, and C. Finn, “Memer: Scaling up memory for robot control via experience retrieval,” *arXiv preprint arXiv:2510.20328*, 2025.
- [60] T. Chen, Y. Wang, M. Li, Y. Qin, H. Shi, Z. Li, Y. Hu, Y. Zhang, K. Wang, Y. Chen *et al.*, “Rmbench: Memory-dependent robotic manipulation benchmark with insights into policy design,” *arXiv preprint arXiv:2603.01229*, 2026.
- [61] M. Torne, K. Pertsch, H. Walke, K. Vedder, S. Nair, B. Ichter, A. Z. Ren, H. Wang, J. Tang, K. Stachowicz *et al.*, “Mem: Multi-scale embodied memory for vision language action models,” *arXiv preprint arXiv:2603.03596*, 2026.
- [62] A. Blattmann, T. Dockhorn, S. Kulal, D. Mendelevitch, M. Kilian, D. Lorenz, Y. Levi, Z. English, V. Voleti, A. Letts *et al.*, “Stable video diffusion: Scaling latent video diffusion models to large datasets,” *arXiv preprint arXiv:2311.15127*, 2023.
- [63] T. Wan, A. Wang, B. Ai, B. Wen, C. Mao, C.-W. Xie, D. Chen, F. Yu, H. Zhao, J. Yang *et al.*, “Wan: Open and advanced large-scale video generative models,” *arXiv preprint arXiv:2503.20314*, 2025.
- [64] A. Ali, J. Bai, M. Bala, Y. Balaji, A. Blakeman, T. Cai, J. Cao, T. Cao, E. Cha, Y.-W. Chao *et al.*, “World simulation with video foundation models for physical ai,” *arXiv preprint arXiv:2511.00062*, 2025.
- [65] P. Intelligence, B. Ai, A. Amin, R. Aniceto, A. Balakrishna, G. Balke, K. Black, G. Bokinsky, S. Cao, T. Charbonnier *et al.*, “pi-0.7: a steerable generalist robotic foundation model with emergent capabilities,” *arXiv preprint arXiv:2604.15483*, 2026.
- [66] Y. Hu, Y. Guo, P. Wang, X. Chen, Y.-J. Wang, J. Zhang, K. Sreenath, C. Lu, and J. Chen, “Video prediction policy: A generalist robot policy with predictive visual representations,” in *International Conference on Machine Learning*. PMLR, 2025, pp. 24 328–24 346.
- [67] J. Pai, L. Achenbach, V. Montesinos, B. Forrai, O. Mees, and E. Nava, “mimic-video: Video-action models for generalizable robot control beyond vlacs,” *arXiv preprint arXiv:2512.15692*, 2025.
- [68] Y. Tian, S. Yang, J. Zeng, P. Wang, D. Lin, H. Dong, and J. Pang, “Predictive inverse dynamics models are scalable learners for robotic manipulation,” in *International Conference on Learning Representations*, vol. 2025, 2025, pp. 92 033–92 052.
- [69] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.
- [70] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.
- [71] W. Peebles and S. Xie, “Scalable diffusion models with transformers,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4195–4205.
- [72] J. Song, C. Meng, and S. Ermon, “Denoising diffusion implicit models,” in *International Conference on Learning Representations*, 2020.
- [73] H. R. Walke, K. Black, T. Z. Zhao, Q. Vuong, C. Zheng, P. Hansen-Estruch, A. W. He, V. Myers, M. J. Kim, M. Du *et al.*, “Bridgedata v2: A dataset for robot learning at scale,” in *Conference on Robot Learning*. PMLR, 2023, pp. 1723–1736.
- [74] S. Fei, S. Wang, J. Shi, Z. Dai, J. Cai, P. Qian, L. Ji, X. He, S. Zhang, Z. Fei *et al.*, “Libero-plus: In-depth robustness analysis of vision-language-action models,” *arXiv preprint arXiv:2510.13626*, 2025.
- [75] J. Ho and T. Salimans, “Classifier-free diffusion guidance,” in *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.
- [76] M. Reuss, Ö. E. Yağmurlu, F. Wenzel, and R. Lioutikov, “Multimodal diffusion transformer: Learning versatile behavior from multimodal goals,” in *Robotics: Science and Systems*, 2024.
- [77] J. Zheng, J. Li, D. Liu, Y. Zheng, Z. Wang, Z. Ou, Y. Liu, J. Liu, Y.-Q. Zhang, and X. Zhan, “Universal actions for enhanced embodied foundation models,” in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 22 508–22 519.
- [78] X. Jia, Q. Wang, A. Donat, B. Xing, G. Li, H. Zhou, O. Celik, D. Blessing, R. Lioutikov, and G. Neumann, “Mail: Improving imitation learning with selective state space models,” in *8th Annual Conference on Robot Learning*, 2024.
- [79] D. Qu, H. Song, Q. Chen, Y. Yao, X. Ye, Y. Ding, Z. Wang, J. Gu, B. Zhao, D. Wang *et al.*, “Spatialvla: Exploring spatial representations for visual-language-action model,” in *Proceedings of Robotics: Science and Systems*, 2025.
- [80] J. Cen, C. Yu, H. Yuan, Y. Jiang, S. Huang, J. Guo, X. Li, Y. Song, H. Luo, F. Wang *et al.*, “Worldvla: Towards autoregressive action world model,” *arXiv preprint arXiv:2506.21539*, 2025.
- [81] K. Pertsch, K. Stachowicz, B. Ichter, D. Driess, S. Nair, Q. Vuong, O. Mees, C. Finn, and S. Levine, “Fast: Efficient action tokenization for vision-language-action models,” *arXiv preprint arXiv:2501.09747*, 2025.
- [82] Z. Liu, Y. Gu, S. Zheng, X. Xue, and Y. Fu, “Trivla: A unified triple-system-based unified vision-language-action model for general robot control,” *arXiv preprint arXiv:2507.01424*, 2025.
- [83] M. Shukor, D. Aubakirova, F. Capuano, P. Kooijmans, S. Palma, A. Zouitine, M. Aractingi, C. Pascal, M. Russi, A. Marafioti *et al.*, “Smolvla: A vision-language-action model for affordable and efficient robotics,” *arXiv preprint arXiv:2506.01844*, 2025.
- [84] W. Zhang, H. Liu, Z. Qi, Y. Wang, X. Yu, J. Zhang, R. Dong, J. He, H. Wang, Z. Zhang *et al.*, “Dreamvla: a vision-language-action model dreamed with comprehensive world knowledge,” *Advances in Neural Information Processing Systems*, vol. 38, pp. 24 195–24 228, 2026.
- [85] Y. Wang, X. Li, W. Wang, J. Zhang, Y. Li, Y. Chen, X. Wang, and Z. Zhang, “Unified vision-language-action model,” in *The Fourteenth International Conference on Learning Representations*, 2026.
- [86] M. J. Kim, C. Finn, and P. Liang, “Fine-tuning vision-language-action models: Optimizing speed and success,” in *Proceedings of Robotics: Science and Systems*, 2025.
- [87] J. Yang, R. Tan, Q. Wu, R. Zheng, B. Peng, Y. Liang, Y. Gu, M. Cai, S. Ye, J. Jang *et al.*, “Magma: A foundation model for multimodal ai agents,” in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 14 203–14 214.
- [88] Y. Shen, F. Wei, Z. Du, Y. Liang, Y. Lu, J. Yang, N. Zheng, and B. Guo, “Videovla: Video generators can be generalizable robot manipulators,” *Advances in neural information processing systems*, vol. 38, pp. 95 597–95 621, 2026.
- [89] X. Li, M. Liu, H. Zhang, C. Yu, J. Xu, H. Wu, C. Cheang, Y. Jing, W. Zhang, H. Liu *et al.*, “Vision-language foundation models as effective robot imitators,” in *International Conference on Learning Representations*, vol. 2024, 2024, pp. 26 703–26 721.
- [90] H. Wu, Y. Jing, C. Cheang, G. Chen, J. Xu, X. Li, M. Liu, H. Li, and T. Kong, “Unleashing large-scale video generative pre-training for visual robot manipulation,” in *International Conference on Learning Representations*, vol. 2024, 2024, pp. 10 641–10 662.
- [91] Y. Chen, Y. Ge, W. Tang, Y. Li, Y. Ge, M. Ding, Y. Shan, and X. Liu, “Moto: Latent motion token as the bridging language for learning robot manipulation from videos,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025, pp. 19 752–19 763.

- [92] Q. Bu, J. Zeng, L. Chen, Y. Yang, G. Zhou, J. Yan, P. Luo, H. Cui, Y. Ma, and H. Li, "Closed-loop visuomotor control with generative expectation for robotic manipulation," *Advances in Neural Information Processing Systems*, vol. 37, pp. 139 002–139 029, 2024.
- [93] Q. Bu, H. Li, L. Chen, J. Cai, J. Zeng, H. Cui, M. Yao, and Y. Qiao, "Towards synergistic, generalized, and efficient dual-system for robotic manipulation," *arXiv preprint arXiv:2410.08001*, 2024.
- [94] Q. Bu, Y. Yang, J. Cai, S. Gao, G. Ren, M. Yao, P. Luo, and H. Li, "Univla: Learning to act anywhere with task-centric latent actions," *arXiv preprint arXiv:2505.06111*, 2025.
- [95] C.-Y. Hung, Q. Sun, P. Hong, A. Zadeh, C. Li, U. Tan, N. Majumder, S. Poria *et al.*, "Nora: A small open-sourced generalist vision language action model for embodied tasks," *arXiv preprint arXiv:2504.19854*, 2025.
- [96] T. Yuan, Z. Dong, Y. Liu, and H. Zhao, "Fast-wam: Do world action models need test-time future imagination?" *arXiv preprint arXiv:2603.16666*, 2026.
- [97] S. Tan, K. Dou, Y. Zhao, and P. Krähenbühl, "Interactive post-training for vision-language-action models," *arXiv preprint arXiv:2505.17016*, 2025.
- [98] A. Hore and D. Ziou, "Image quality metrics: Psnr vs. ssim," in *2010 20th international conference on pattern recognition*. IEEE, 2010, pp. 2366–2369.
- [99] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [100] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.
- [101] T. Unterthiner, S. Van Steenkiste, K. Kurach, R. Marinier, M. Michalski, and S. Gelly, "Towards accurate generative models of video: A new metric & challenges," *arXiv preprint arXiv:1812.01717*, 2018.
- [102] J. Zhang, Z. Huang, C. Gu, Z. Ma, and L. Zhang, "Reinforcing action policies by prophesying," *arXiv preprint arXiv:2511.20633*, 2025.



Yulin Wang received the BS degree in automation from Beihang University, Beijing, China, in 2019, and the PhD degree in automation from Tsinghua University, Beijing, China, in 2025. He was a Visiting Student with the University of California, Berkeley, Berkeley, CA, USA, in 2018. He is currently a Postdoctoral Researcher with the University of Oxford, Oxford, U.K. His research interests include deep learning and computer vision.



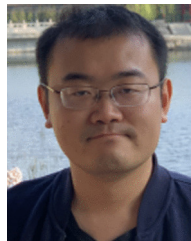
Renping Zhou received the BS degree in computer science and technology in 2025 from Tsinghua University, Beijing, China. He is currently working toward the PhD degree with the Department of Automation, Tsinghua University, Beijing, China. His research interests include embodied intelligence and computer vision.



Tiancai Wang received the MS degree from Tianjin University, Tianjin, China. He was a Research Intern with the Inception Institute of Artificial Intelligence, Abu Dhabi, United Arab Emirates, and a Senior Researcher with Megvii Research, Beijing, China. He is currently a Co-founder and Researcher with Dexmal. His research interests include embodied intelligence and computer vision.



Hao Shi received the BS degree in computer science from Tianjin University, Tianjin, China, in 2023. He is currently working toward the MS degree with the Department of Automation, Tsinghua University, Beijing, China. He is also a Visiting Student with MMLab, The University of Hong Kong, Hong Kong, China. His research interests include embodied intelligence and computer vision.



Xiangyu Zhang received the PhD degree from Xi'an Jiaotong University, Xi'an, China, in 2017. He was a Principal Researcher with Megvii Research, Beijing, China. He is currently a Co-founder and Chief Scientist with Stepfun. His research interests include computer vision and deep learning. He got the CVPR Best Paper Award in 2016. The total number of his Google Scholar citations is more than 450,000.



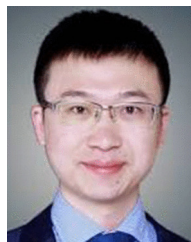
Weiye Li received the BS degree in automation from Beihang University, Beijing, China, in 2025. He is currently working toward the MS degree with the Department of Automation, Tsinghua University, Beijing, China. His research interests include embodied intelligence.



Ping Luo (Senior Member, IEEE) received the PhD degree in information engineering from the Chinese University of Hong Kong, Hong Kong, in 2014. He is currently an Associate Professor with the Department of Computer Science, The University of Hong Kong, Hong Kong, China. From 2014 to 2016, he was a Postdoctoral Researcher with the Chinese University of Hong Kong. From 2017 to 2018, he was a Principal Research Scientist with SenseTime Research. His research interests include machine learning and computer vision.



Bin Xie received the MS degree in electronic information engineering from Tianjin University, Tianjin, China, in 2025. He is currently a Researcher with Dexmal. His research interests include embodied intelligence and computer vision.



Gao Huang (Member, IEEE) received the BS degree in automation from Beihang University, Beijing, China, in 2009, and the PhD degree in automation from Tsinghua University, Beijing, China, in 2015. He was a Visiting Research Scholar with the Department of Computer Science and Engineering, Washington University in St. Louis, St. Louis, MO, USA, in 2013, and a Postdoctoral Researcher with the Department of Computer Science, Cornell University, Ithaca, NY, USA, from 2015 to 2018. He is currently an Associate Professor with the Department of Automation, Tsinghua University, Beijing, China. His research interests include machine learning and deep learning.

SUPPLEMENTARY MATERIALS

A. Additional Simulation Training and Evaluation Setup

1) *Libero*: Following OpenVLA [1], we train with 50 demonstrations per task after removing failed trajectories from the dataset. For MemoryVLA++, Spatial, Object, Goal, and Long-10 are jointly trained for 60k steps to align with the subsequent Libero-Plus evaluation, since Libero-Plus is built on these four standard suites. Long-90 is trained separately for 30k steps because it is not included in Libero-Plus. The dataloader adopts a grouped sampling strategy in which each batch is divided into multiple groups, and each group consists of several frames drawn from a single episode. Frames within a group are kept in temporal order. The memory length is set to 16. The corresponding world model is trained for 40k steps. For MemoryVLA, we follow its original training protocol [26], and we observe similar performance between separate and joint training on the Libero.

Evaluation on Libero [22] is conducted across all five suites: Spatial, Object, Goal, Long-10, and Long-90. For Spatial, Object, Goal, and Long-10, evaluation is performed every 10k steps. For Long-90, evaluation is performed every 2k steps. Each task is evaluated with 50 trials, and success rates are reported at the best evaluation checkpoint. Unlike SimplerEnv, no action ensemble strategy is used in our Libero experiments. CogACT results are reproduced using the official codebase for fairness.

2) *SimplerEnv*: On BridgeData-v2, models are trained for 50k steps with a stream dataloader. Each episode is unpacked into consecutive frames tagged with its episode ID. During training, batches are filled sequentially with frames from a single episode whenever possible. If an episode ends before the batch is complete, the remaining slots are filled with frames from the following episode. A new batch then continues from the position where the previous one stopped, ensuring that in-episode temporal order is always preserved. The memory length is fixed to 16. The corresponding world model is trained for 40k steps.

Evaluation follows the official CogACT protocol [3]. We adopt the same evaluation scripts and use the adaptive action ensemble strategy introduced in CogACT, with ensemble coefficient $\alpha = 0.1$ and ensemble horizon set to 7 for Bridge. Models are validated every 2.5k steps. Since the denoising objective of diffusion models does not reliably indicate policy quality, we report success rates at the best validation checkpoint. Each task is evaluated over 24 trials.

Since the original paper only reported per-task success rates for CogACT-Base but not for CogACT-Large, we re-evaluated the released CogACT-Large checkpoint in our setup and report those numbers for fairness. For π_0 , results are taken from the open-source reproduction open-pi-zero, which provides implementations with both *uniform* and *beta* timestep sampling strategies in flow matching. We report results under float32 precision as in the public release.

3) *Mikasa-Robo*: For Mikasa-Robo [24], we adopt the standard protocol with five tasks and train jointly on all 1,250 demonstrations for 20k steps, using 128×128 RGB observations and Δ end-effector control. We reuse the same

dataloader setup as in Libero, and set the memory length to 16. The corresponding world model is trained for 20k steps.

Validation is performed every 1k training steps, and success rates are reported at the checkpoint with the best validation performance. Each task is evaluated with 100 episodes. As in Libero, we do not use any action ensemble strategy in our Mikasa-Robo experiments.

4) *Calvin*: For Calvin [25], we adopt the ABC→D zero-shot setting, where policies are trained on environments A, B, and C and evaluated on the unseen environment D. We use 17,870 training trajectories from environments A, B, and C. The dataloader follows the same grouped sampling strategy as Libero. Models are trained for 60k steps. For the world model, we directly follow VPP [66].

Models are evaluated every 2k steps and reported at the best evaluation checkpoint. We report success rates over 1000 rollouts and the average number of completed tasks.

5) *Libero-Plus*: For Libero-Plus [74], we evaluate both zero-shot and supervised fine-tuning settings. Libero-Plus introduces 7 out-of-distribution variations, covering camera, robot, language, lighting, background, noise, and layout. In the zero-shot setting, models are trained on the 4 standard Libero suites, including Spatial, Object, Goal, and Long-10, and evaluated on Libero-Plus without using its training data. In the supervised fine-tuning setting, Libero-Plus data are mixed with the 4 standard Libero suites for joint training. Both settings use the same grouped dataloader as Libero and are trained for 60k steps. OpenVLA-OFT and our method are both jointly trained on all suites and use wrist-view images in this benchmark. The corresponding world model is trained for 40k steps.

Evaluation is performed every 10k steps, and results are reported at the best evaluation checkpoint.

B. Additional Real-Robot Training and Evaluation Setup

Models are trained for 5k-30k steps depending on task complexity and dataset size. The general tasks contain 50-150 demonstrations per task, while long-horizon memory-dependent and imagination-dependent tasks use 200-300 demonstrations per task. The memory length is set to 16 for general tasks and 256 for long-horizon temporal tasks. For MemoryVLA++, the world model is trained for 20k steps and used for long-horizon imagination-dependent tasks.

Evaluation uses 15-25 trials for General tasks and 10-15 trials for Long-horizon Temporal tasks. For General tasks, *Pick Diverse Fruits* contains five variants (apple, orange, banana, chili, grape), each evaluated with 5 trials (25 total). All other General tasks are evaluated with 15 trials each, and we report task-level success rates.

For long-horizon memory-dependent tasks, *Seq. Push Buttons* includes three button orders (blue-pink-green, blue-green-pink, green-blue-pink), each tested with 5 trials. All other tasks are evaluated with 10 trials, and step-wise scoring is adopted to capture partial progress. The scoring rules are as follows:

- **Seq. Push Buttons**: pressing each correct button yields 30, with a bonus of 10 if all three are correct. Loose matching is allowed (slight contact counts as a press).

- **Change Food:** lifting and removing the initial food (30), grasping the new food (30), and placing it on the plate (30), with a 10 bonus for full success.
- **Guess Where:** grasping the cover (30), covering the block (30), and uncovering it (40).
- **Clean Table & Count:** five objects in total. For each object, clearing yields 10 points and pressing the counter yields 10. Small counting errors (incomplete press / one extra press) earn 5; major errors (missed count / multiple extras) earn 0. Empty grasps with clear counting intent incur a 5-point penalty.
- **Pick Place Order:** carrot, banana, and orange must be picked and placed in sequence. Each correct step earns 30, with a 10 bonus for full completion. Any order violation terminates the attempt.
- **Clean Restaurant Table:** five objects in total. Each correctly sorted into trash bin or storage bin scores 20. Misplacement earns 10, and merely lifting without correct placement earns 5.

For long-horizon imagination-dependent tasks, evaluation uses 10 consecutive trials per task. Step-wise scoring is adopted to capture partial progress within each trial. The scoring rules are as follows:

- **Conveyor-Pick:** five objects in total. Low-, mid-, and high-speed settings use the same scoring rule. For each object, successful grasping yields 10 points and placing it into the designated box yields 10.
- **Conveyor Scan-Pick:** five objects in total. For each object, successful grasping yields 6 points, successful scanning yields 6, and placing it into the box yields 6. A 2-point bonus is awarded if the object is fully completed.
- **Bag Pack & Zip:** reaching into the bag with the left arm yields 10 points, opening the bag yields 10, and placing each of the five objects into the bag yields 10. A 10-point bonus is awarded if all five objects are successfully packed. Grasping the zipper pull with the right arm yields 10, and fully closing the zipper yields 10.

C. Data Augmentation

We apply standard per-frame augmentations to the third-person RGB stream during training. Augmentations are applied in a fixed order: random resized crop, random brightness, random contrast, random saturation, and random hue. The crop samples 90% of the image area with aspect ratio 1.0 and resizes to 224×224 . Brightness is perturbed with magnitude 0.2, contrast and saturation are scaled in $[0.8, 1.2]$, and hue is shifted by up to 0.05. All augmentations are disabled at evaluation.

D. Data Length Statistics

Tab. XII reports the maximum, minimum, median, and average action lengths across all task suites, including SimplierEnv Evaluation (Bridge, Fractal), LIBERO (Spatial/Object/Goal and 10/90 task suites), and both real-world general and temporal tasks. For the real-world tasks, we additionally provide filtered statistics based on a motion-magnitude threshold (translation > 1 cm or rotation > 0.4 rad between consecutive

frames) to remove frames where the end-effector motion is small.

TABLE XII
ACTION LENGTH STATISTICS FOR SIMULATION AND REAL-WORLD TASK SUITES. FOR CALVIN, WE REPORT THE LENGTH OF A 5-SUBTASK SEQUENCE BY MULTIPLYING THE PER-SUBTASK ACTION LENGTH BY 5. FOR REAL-WORLD TASKS, “FILTERED” REMOVES FRAMES WITH NEGLIGIBLE END-EFFECTOR MOTION (TRANSLATION < 1 CM AND ROTATION < 0.4 RAD).

Task Suite	Max	Min	Median	Average
Libero-Spatial / Object / Goal	270	75	131	130
Libero-10 / 90	505	58	144	156
SimplierEnv	200	80	117	119
Mikasa-Robo	90	60	60	72
Calvin	325	170	325	300
Libero-Plus	505	75	138	156
Real-General (Original)	1575	281	575	575
Real-General (Filtered)	213	40	81	84
Real-Memory (Original)	7704	412	981	1672
Real-Memory (Filtered)	902	72	236	288
Real-Imagination (Original)	5737	1796	2809	2935
Real-Imagination (Filtered)	987	258	467	463

E. Real-world Tasks Details

Our real-world evaluation includes 17 tasks, divided into *General*, *Long-horizon Memory-dependent*, and *Long-horizon Imagination-dependent* suites.

a) *General Tasks.*: *Insert Circle*: insert a circle onto a vertical pillar, requiring accurate positioning and insertion.

Egg in Pan: place an egg into a shallow frying pan, testing grasp stability and gentle placement.

Egg in Oven: put an egg into a small oven container, involving more constrained placement than the pan.

Stack Cups: stack one plastic cup on top of another, evaluating vertical alignment and balance.

Stack Blocks: stack a yellow block on top of a red block, focusing on precise spatial alignment.

Pick Diverse Fruits: pick a specified fruit from a tabletop with more than ten different fruit types and place it into a basket, testing semantic understanding, visual diversity, and instruction following.

b) *Long-horizon Temporal Tasks.*: *Seq. Push Buttons*: push three buttons in a specified color sequence, stressing ordered memory and resistance to temporal confusion.

Change Food: remove a food item from a plate and replace it with another, requiring multi-step sequencing and correct temporal ordering.

Guess Where: cover a block with a container and later uncover it, testing reversible actions and consistent tracking over time.

Clean Table & Count: clear items from the table one by one while pressing a counter button after each removal, combining manipulation with explicit progress monitoring.

Pick Place Order: pick up carrot, banana, and orange in a fixed order and place them into a basket, enforcing sequence-sensitive planning under temporal dependencies.

Clean Restaurant Table: sort table items by category, placing trash into a trash bin and tableware into a storage bin,

representing a long-horizon task with semantic reasoning and complex multi-stage sequencing.

Conveyor-Pick: pick objects arriving sequentially on a moving conveyor belt and place them into a designated box, requiring anticipatory prediction of future object positions and stable grasping under continuous motion.

Conveyor Scan-Pick: pick objects from the conveyor belt, place each object under a scanner for identification, and then deposit it into a box, requiring anticipatory prediction for moving objects and memory for scan across the subsequent placement stage.

Bag Pack & Zip: grasp objects one by one from the table, place them into a deformable bag, and finally grasp the zipper pull to close the bag, evaluating long-horizon planning, bimanual coordination, deformable-object manipulation, and fine-grained zipper alignment.

F. Additional Qualitative Results

We provide additional qualitative results across simulation and real-world benchmarks. Figs. 13, 14, 15, 16, 17, 18, and 19 show results on Libero, SimplerEnv, Mikasa-Robo, Calvin, and Libero-Plus. Figs. 20, 21, 22, 23, and 24 show results on real-world general, memory-dependent, and imagination-dependent tasks. Figs. 25, and 26 show qualitative results of world-model-based future imagination.

G. Supplementary Videos

We also provide supplementary videos to better illustrate the behavior of MemoryVLA++ across simulation and real-world tasks. These videos are included in the supplementary materials and are also available on the project website: <https://shihao1895.github.io/MemoryVLA-PP-Web>.

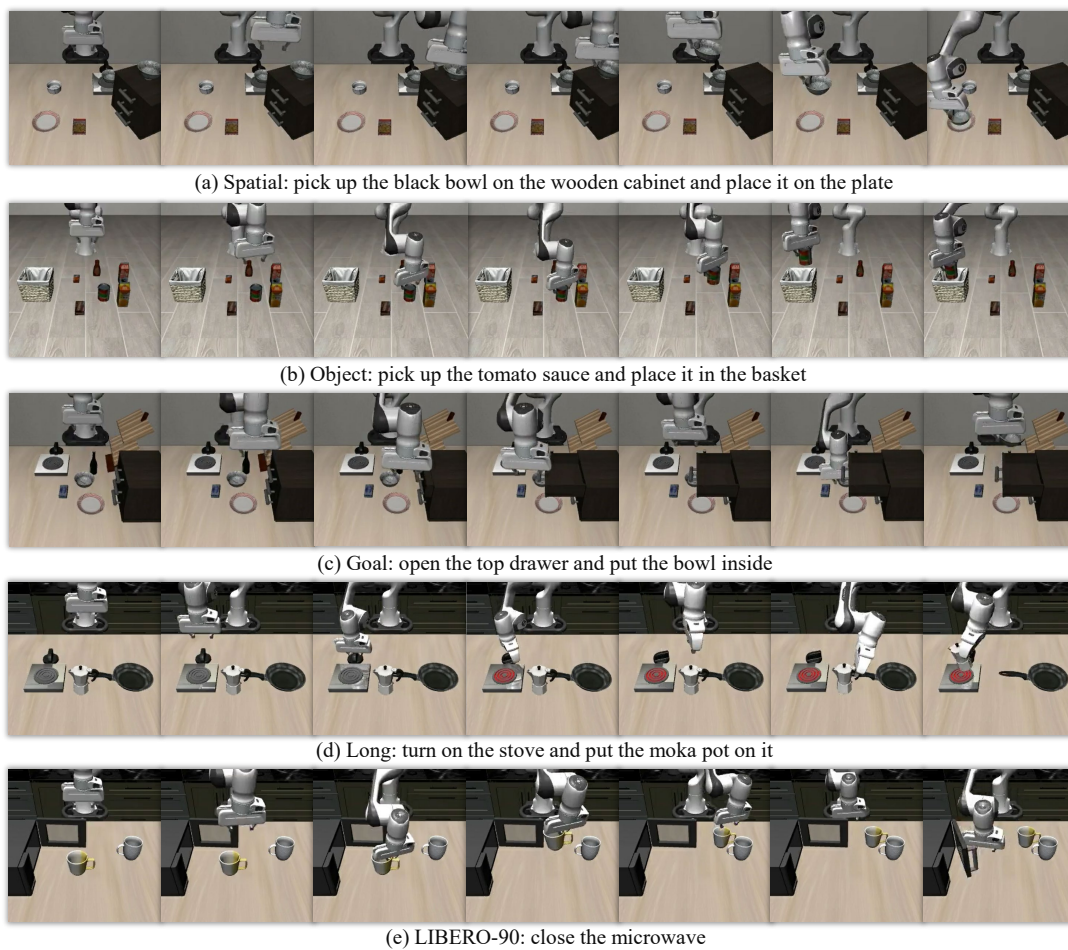


Fig. 13. Qualitative results on Libero.

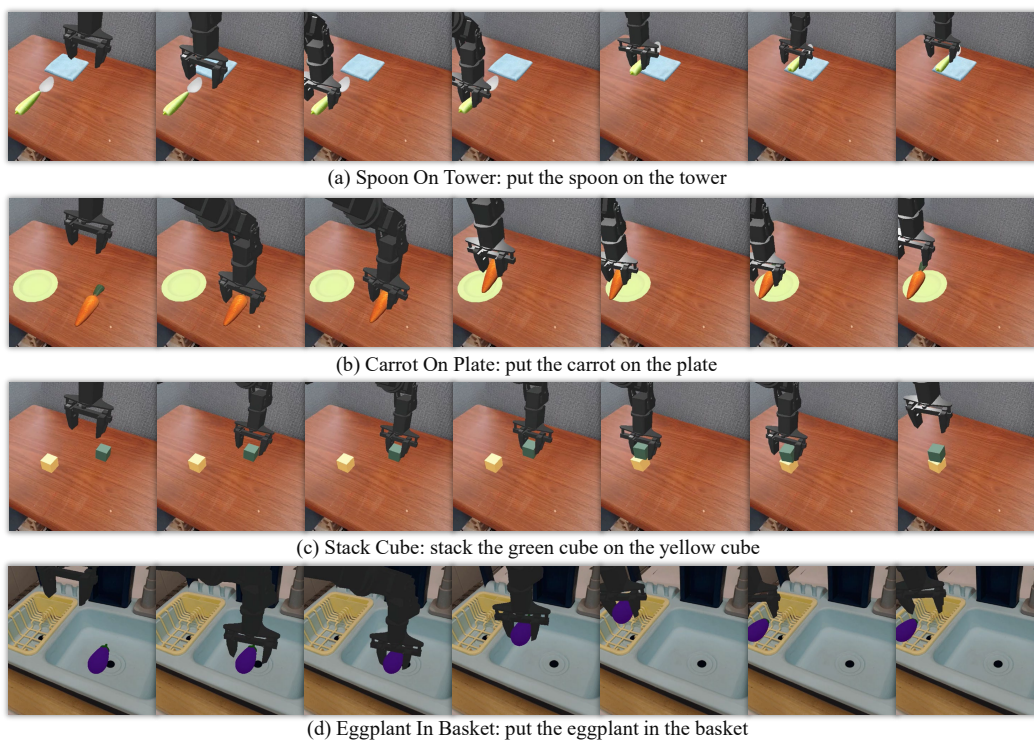


Fig. 14. Qualitative results on SimplerEnv.

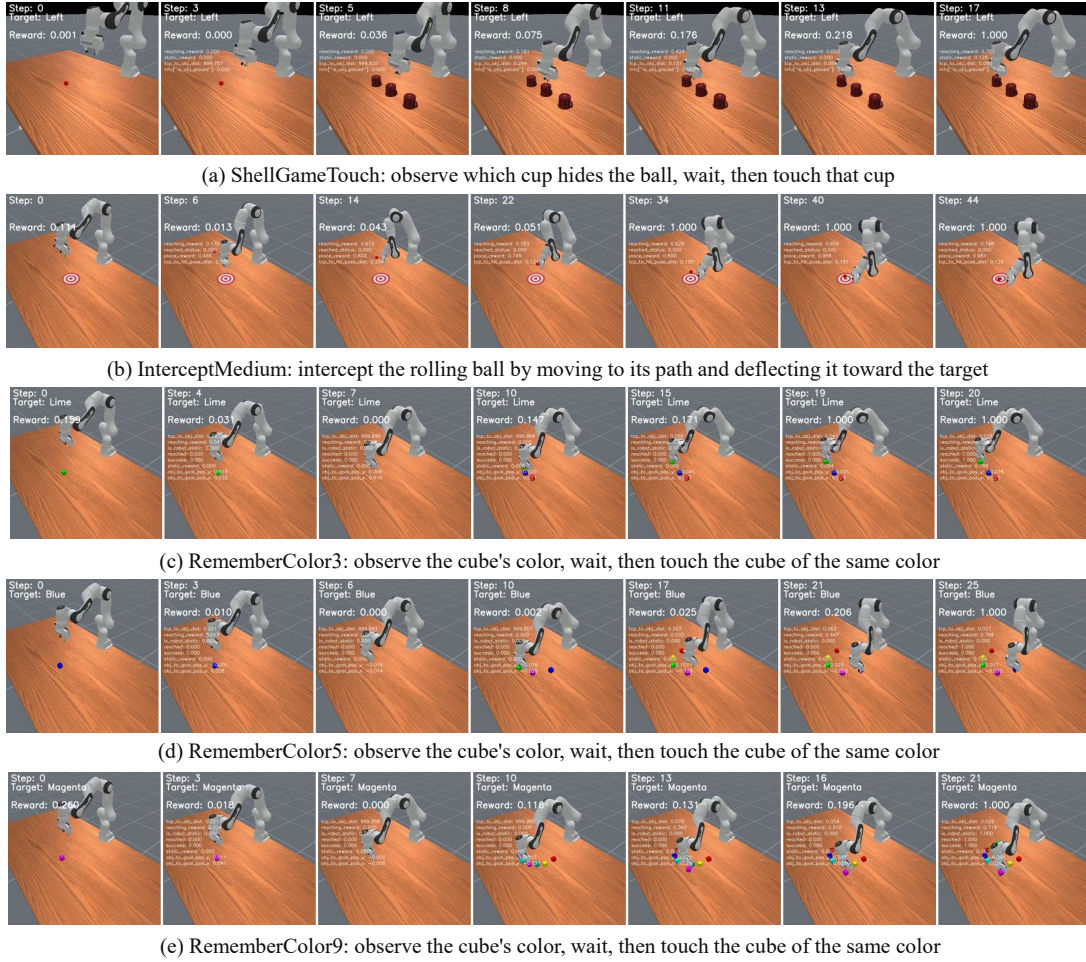
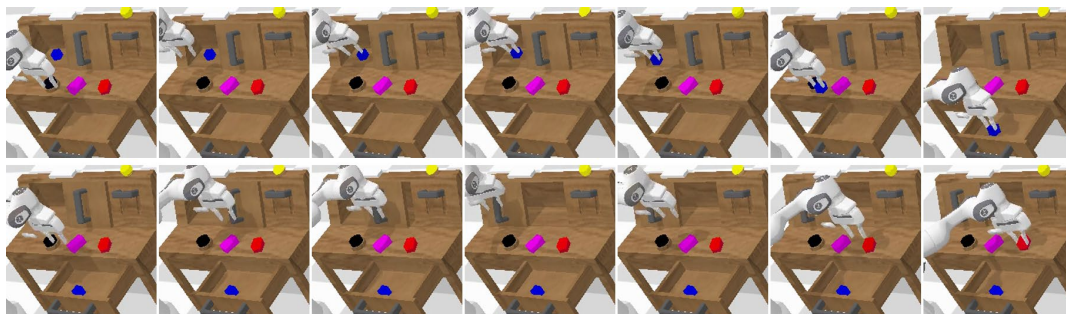


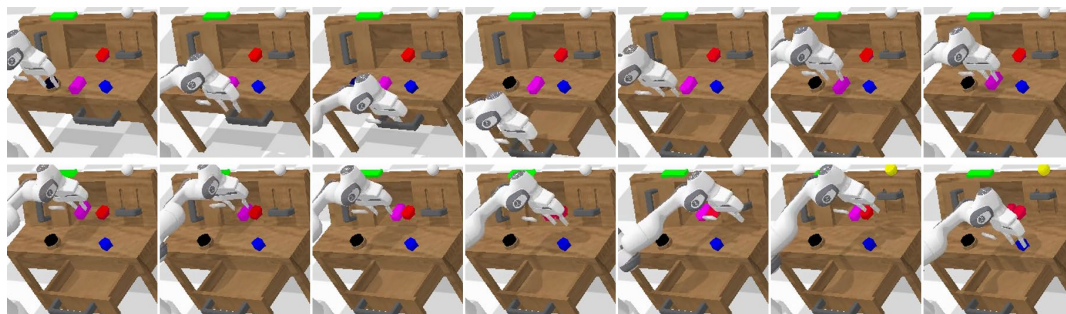
Fig. 15. Qualitative results on Mikasa-Robo.



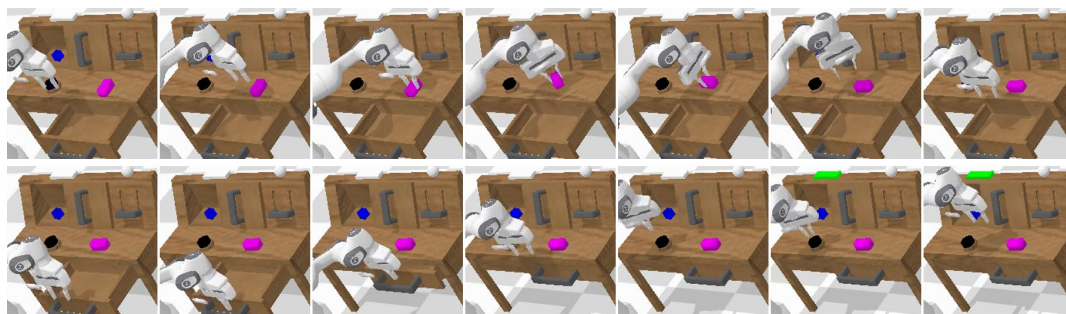
Fig. 16. Qualitative results on Calvin (1).



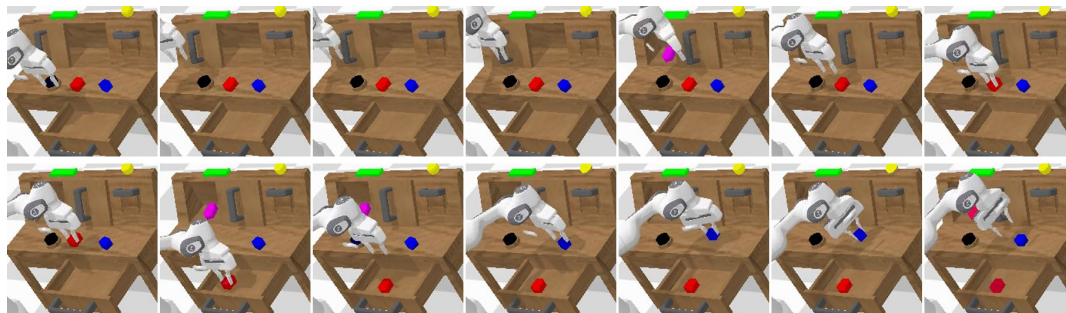
(e) Lift Blue Block Slider→Place In Drawer→Move Slider Left→Rotate Red Block Right→Lift Blue Block Drawer



(f) Open Drawer→Lift Pink Block Table→Place In Slider→Turn On Lightbulb→Rotate Blue Block Left



(g) Rotate Pink Block Left→Close Drawer→Turn On Led→Move Slider Left→Lift Red Block Slider



(h) Move Slider Right→Lift Red Block Table→Place In Drawer→Rotate Blue Block Left→Turn Off Led

Fig. 17. Qualitative results on Calvin (2).

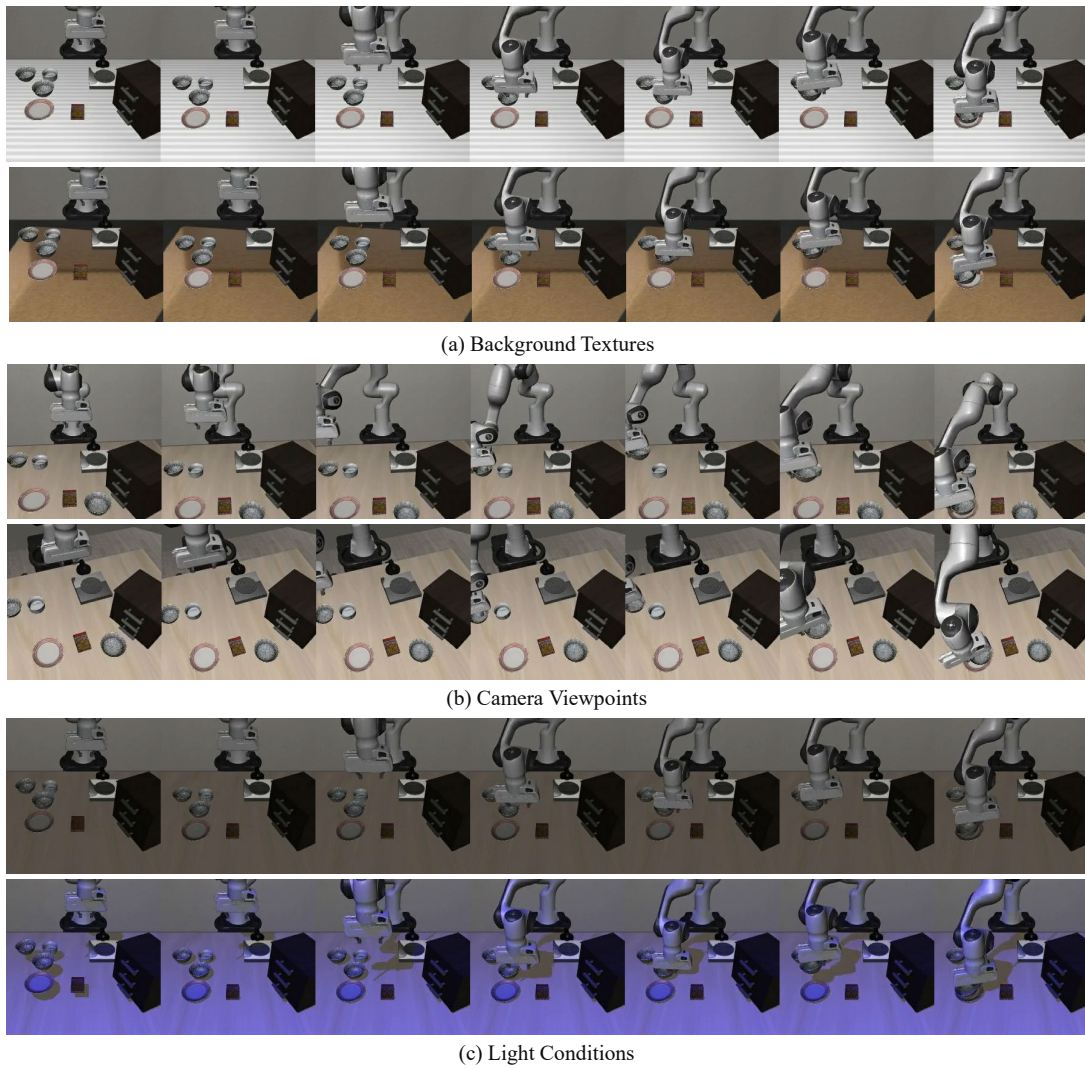
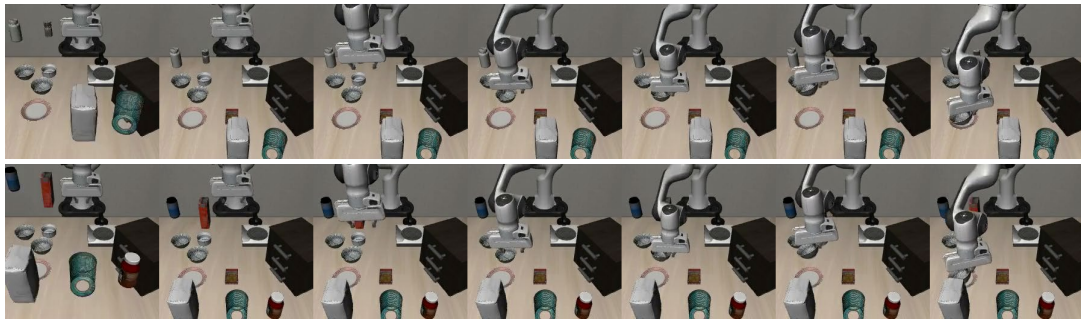
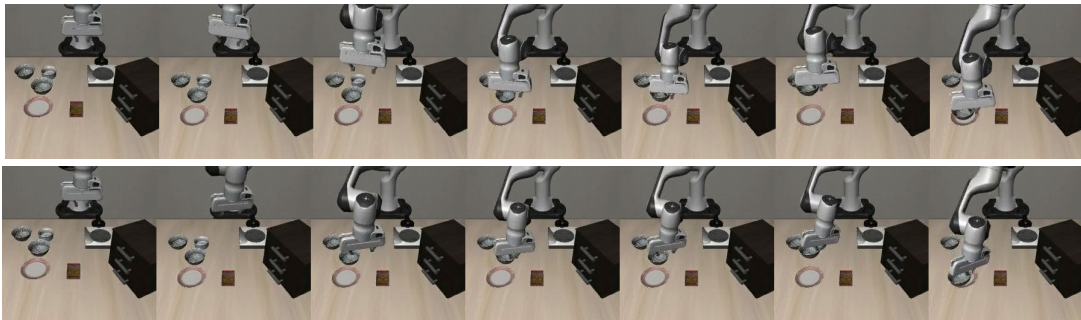


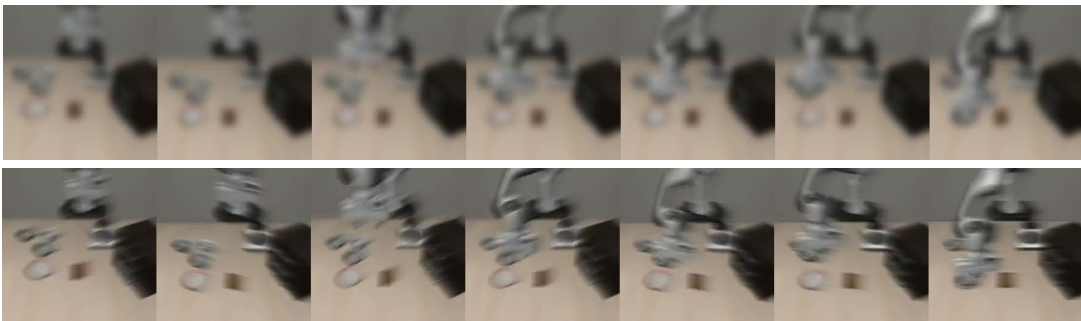
Fig. 18. Qualitative results on Libero-Plus (1).



(d) Objects Layout



(e) Robot Initial States



(f) Sensor Noise

Fig. 19. Qualitative results on Libero-Plus (2).

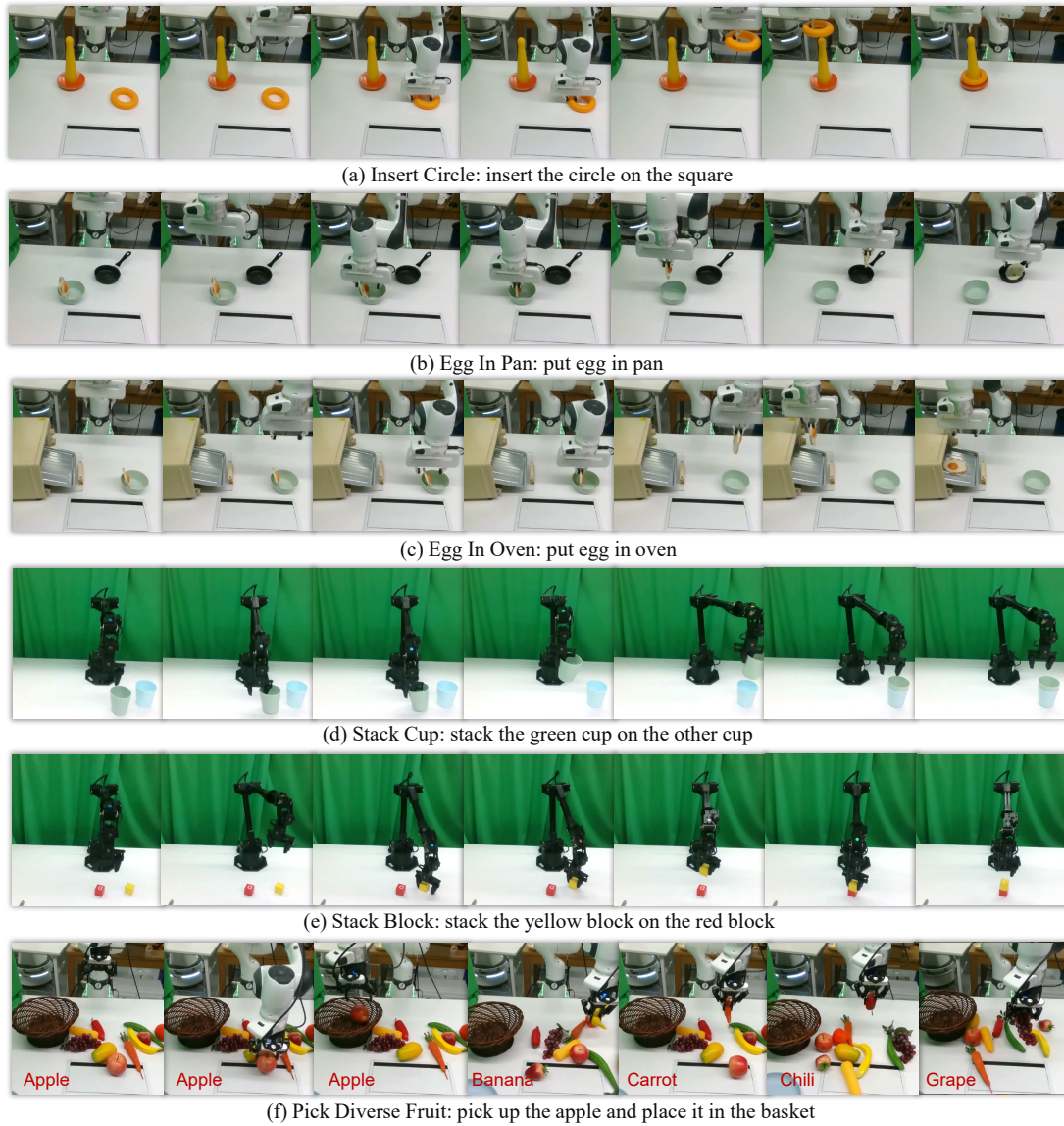


Fig. 20. Qualitative results on real-world general manipulation tasks.

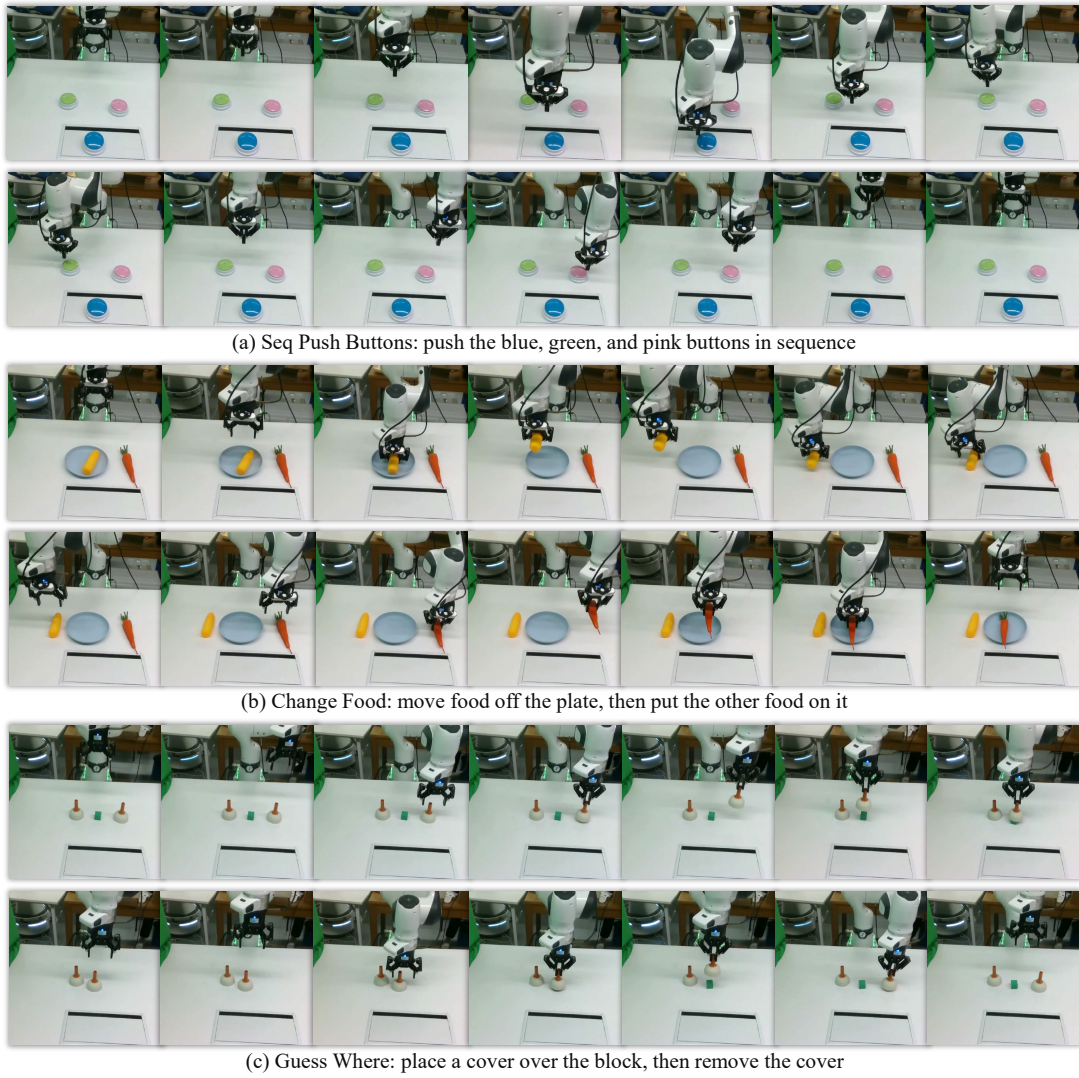
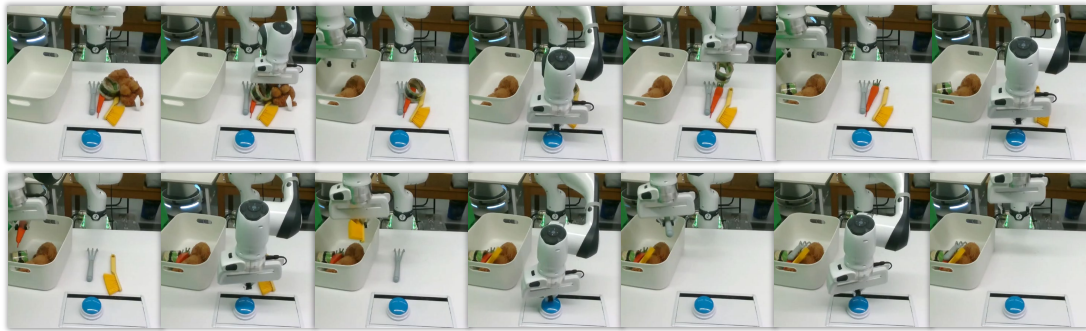
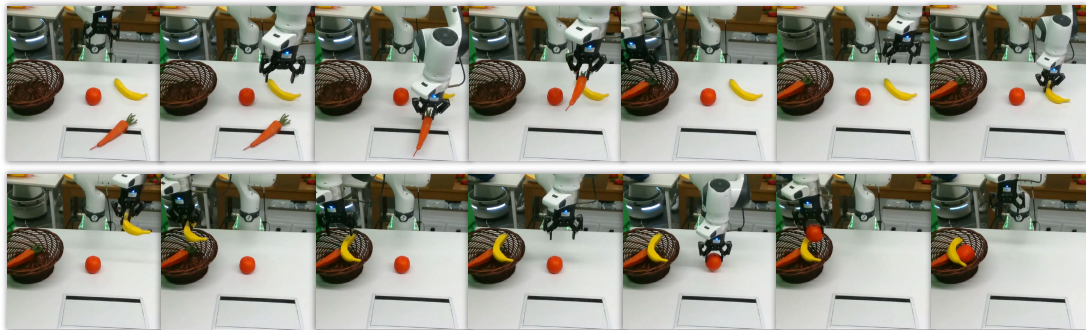


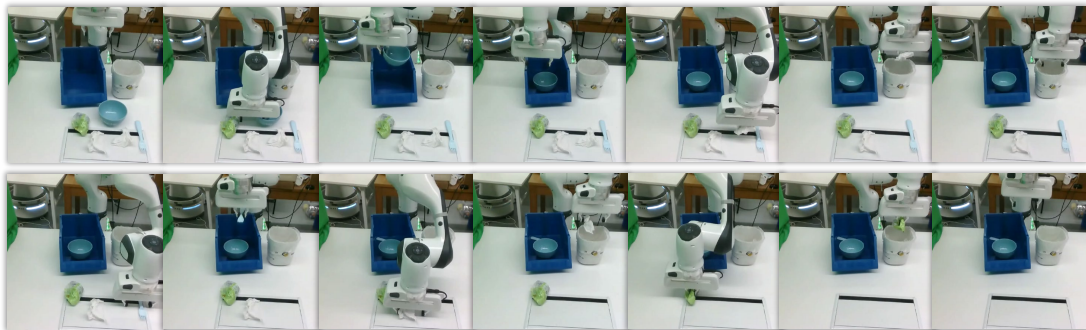
Fig. 21. Qualitative results on real-world memory-dependent tasks (1).



(d) Clean Table & Count: clean the table one by one, and press the button for each item cleaned



(e) Pick Place Order: pick up carrot, banana, and orange in order and place them in the basket

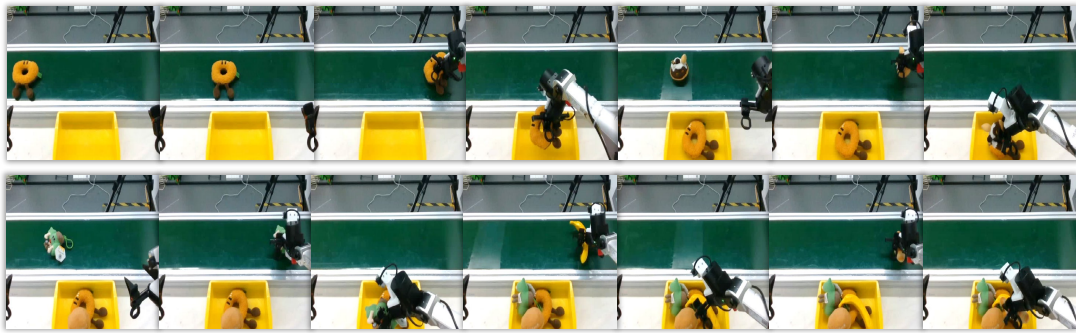


(f) Clean Restaurant Table: place all trash items into the trash bin and all tableware into the storage bin

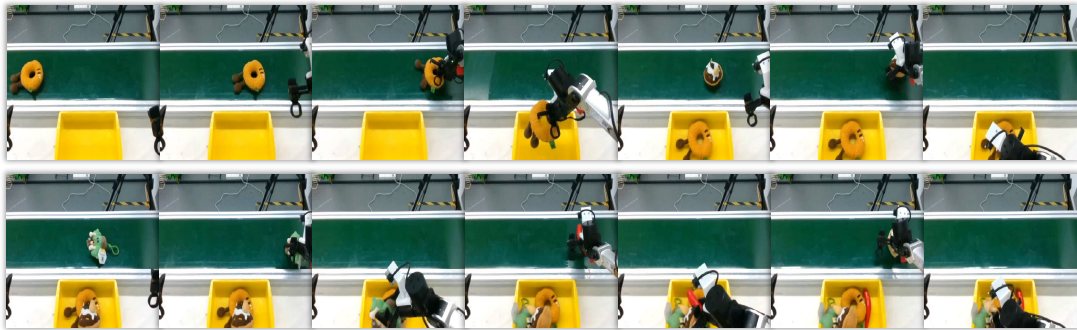
Fig. 22. Qualitative results on real-world memory-dependent tasks (2).



(a) Conveyor Pick (Low): continuously pick objects from the conveyor belt and place them into the box

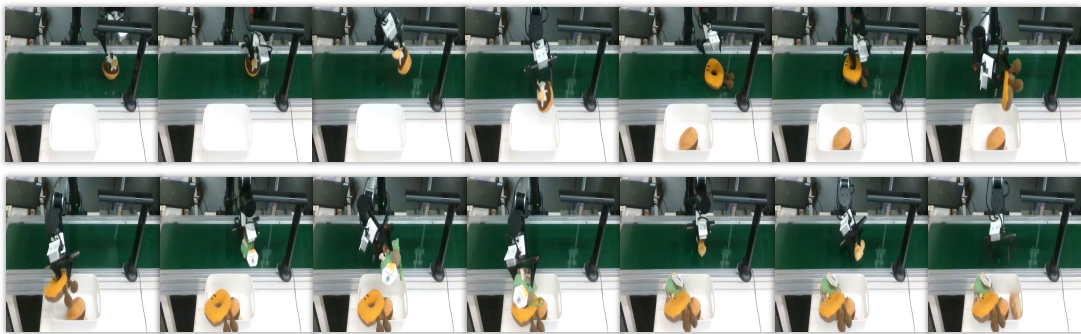


(b) Conveyor Pick (Mid): continuously pick objects from the conveyor belt and place them into the box

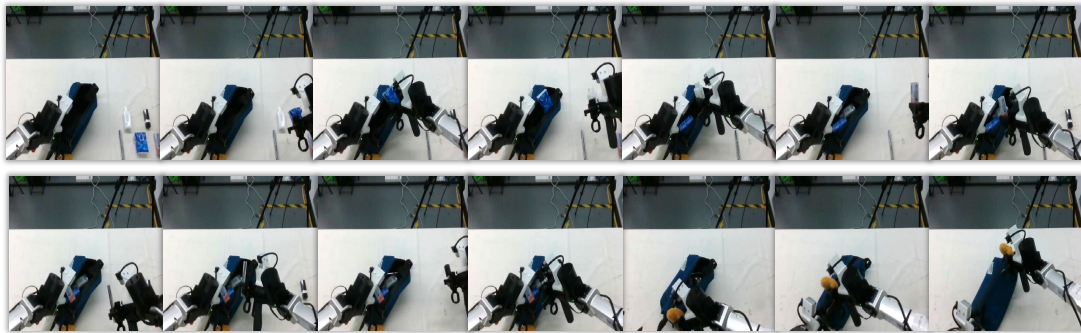


(c) Conveyor Pick (High): continuously pick objects from the conveyor belt and place them into the box

Fig. 23. Qualitative results on real-world imagination-dependent tasks (1).



(d) Conveyor Scan-Pick: pick up each object from the conveyor belt, scan it, and place it in the box



(e) Bag Pack & Zip: Put the objects into the crossbody bag one by one, and then close the zipper

Fig. 24. Qualitative results on real-world imagination-dependent tasks (2).

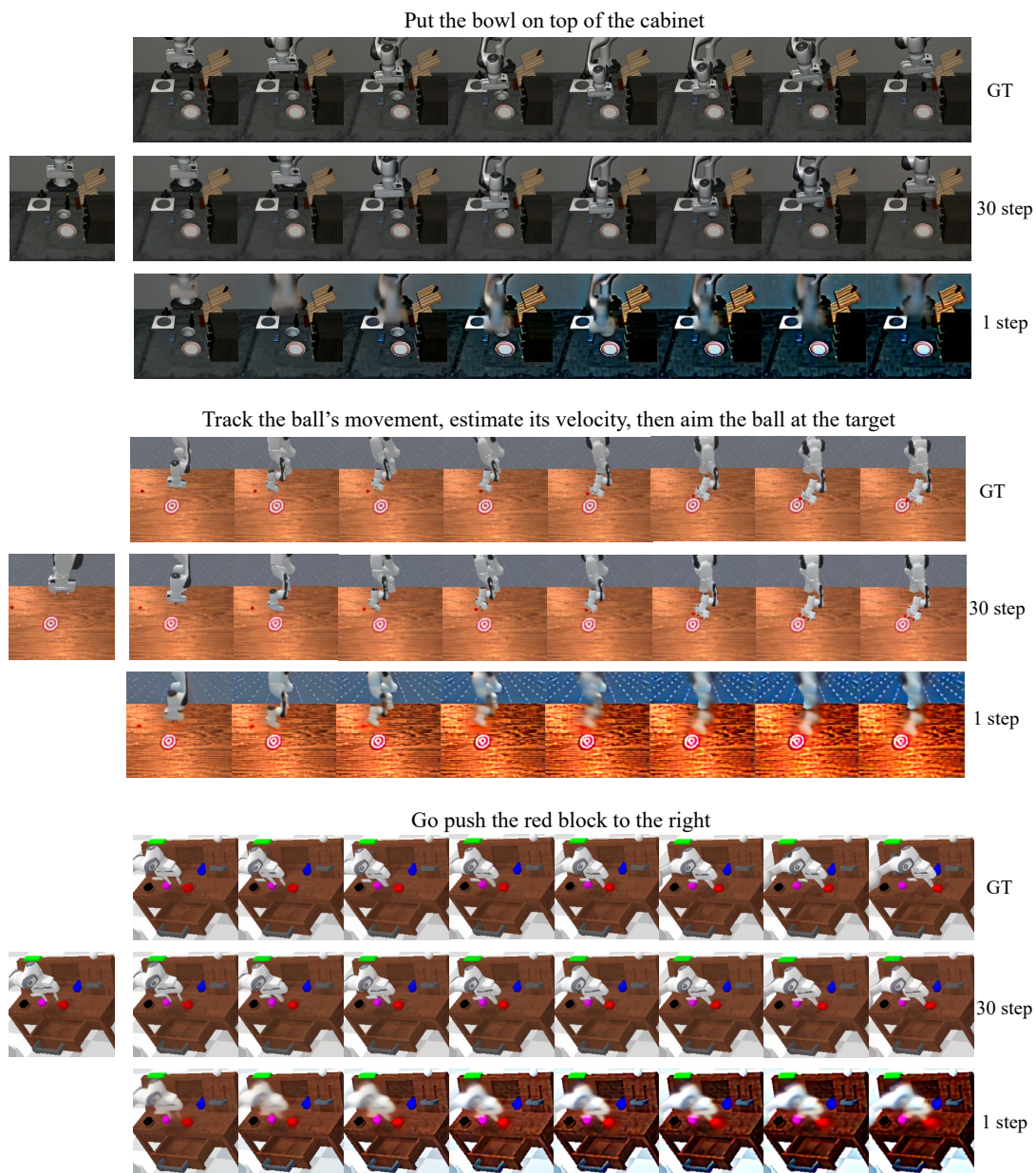


Fig. 25. Qualitative results of world-model-based future imagination (1).

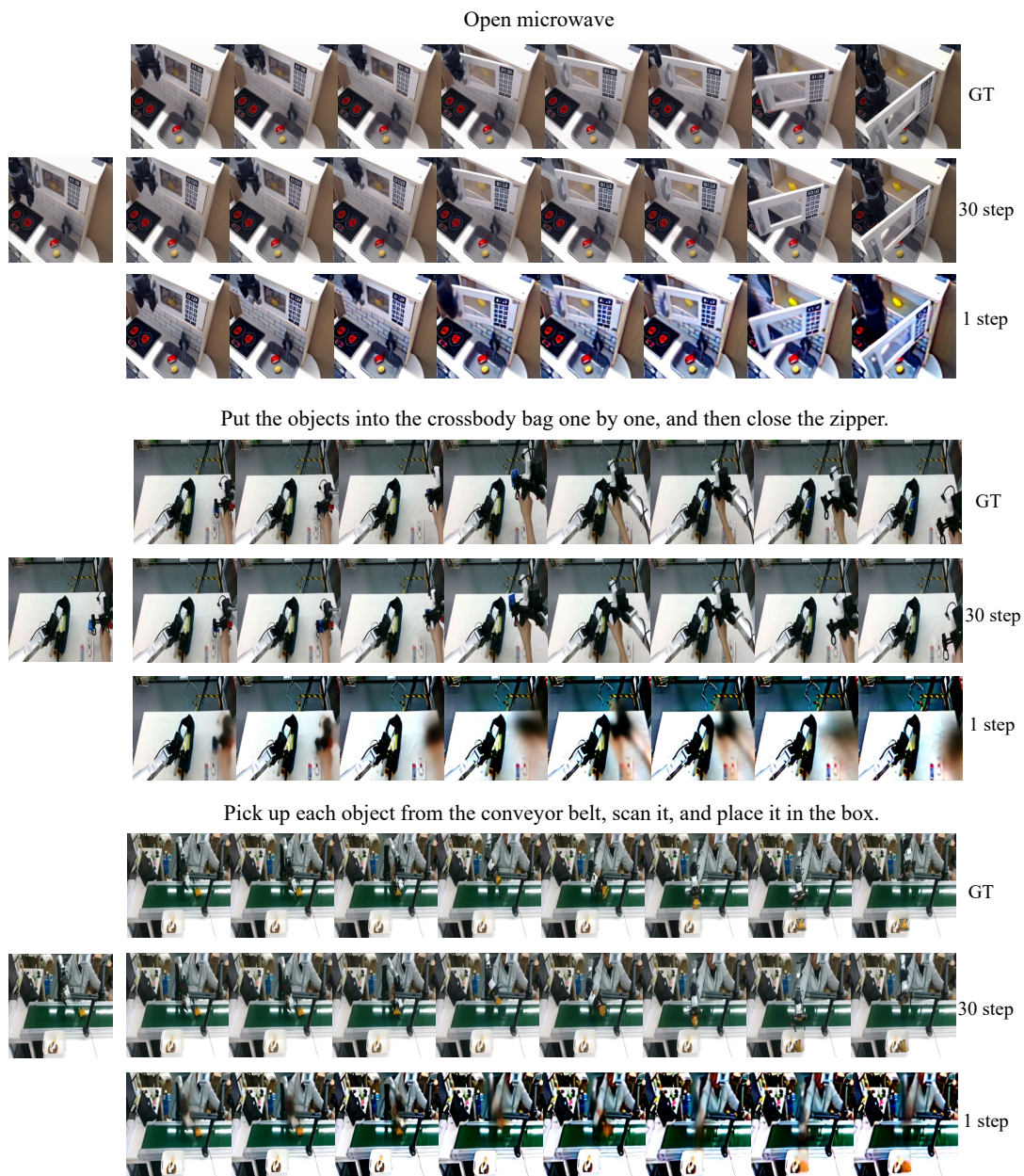


Fig. 26. Qualitative results of world-model-based future imagination (2).