Rethinking Progression of Memory State in Robotic Manipulation: An Object-Centric Perspective

Nhat Chung[1*], Taisei Hanyu[2*], Toan Nguyen[1], Huy Le[1], Frederick Bumgarner[2], Duy Minh Ho Nguyen[3,7,8], Khoa Vo[2], Kashu Yamazaki[4], Chase Rainwater[2], Tung Kieu[5], Anh Nguyen[6], Ngan Le[2],

[1]FPT Software AI Center
[2]University of Arkansas
[3]University of Stuttgart
[4]Carnegie Mellon University
[5]Aalborg University
[6]University of Liverpool
[7]German Research Center for Artificial Intelligence (DFKI)
[8]Max Planck Research School for Intelligent Systems (IMPRS-IS)
Correspondence: nhatchung14@gmail.com, {thanyu, thile}@uark.edu

Abstract

As embodied agents operate in increasingly complex environments, the ability to perceive, track, and reason about individual object instances over time becomes essential, especially in tasks requiring sequenced interactions with visually similar objects. In non-Markovian settings, critical decision cues lie in object histories rather than the current scene. Without persistent memory of prior interactions (what was used, where it was placed, or how it changed), visuomotor policies may fail, repeat past actions, or overlook completed ones. To surface this challenge, we introduce LIBERO-Mem, a non-Markovian task suite for stress-testing robotic manipulation under object-level partial observability. It combines short- and long-horizon object tracking with temporally sequenced subgoals, requiring reasoning beyond the current frame. However, vision-language-action (VLA) models often struggle in such settings, with token scaling quickly becoming intractable even for tasks spanning just a few hundred frames. We propose Embodied-SlotSSM, a slot-centric VLA framework built for temporal scalability. It maintains spatio-temporally consistent slot identities and leverages them through two mechanisms: (1) slot-state-space modeling for reconstructing short-term history, and (2) a relational encoder to align the input tokens with action decoding. Together, these components enable temporally grounded, context-aware action prediction. Experiments show Embodied-SlotSSM's baseline performance on LIBERO-Mem and general tasks, offering a scalable solution for non-Markovian reasoning in object-centric policies.

Extended Version — https://libero-mem.github.io

Introduction

Humans effortlessly recall past interactions with specific objects, such as where they last placed a salt bottle or whether they have already sprinkled salt into a pot of soup a number of times, enabling them to carry out long-horizon, multiple-step tasks with precision. This object-level memory plays a vital role in avoiding redundant actions or missing steps, especially in tasks involving repetitive steps, visually similar items, and extended temporal dependencies. Such object-level challenges are inherently *partially observable Markov decision processes (POMDP)* (illustrated in Fig. 1), where the agent's current observation does not fully capture the true state of the environment, and optimal decisions depend on the history of interactions associated with specific objects.

Conversely, robotic visuomotor policies typically rely solely on the most recent sensory inputs to determine subsequent actions (Kim et al. 2024; Li et al. 2024c; Bharadhwaj et al. 2024; Brohan et al. 2023; Octo Model Team et al. 2024; Zawalski et al. 2024; Wang et al. 2024; Yang et al. 2024; Tian et al. 2024), lacking mechanisms to encode and recall object-centric history (Walke et al. 2023; O'Neill et al. 2024; Khazatsky et al. 2024; James et al. 2020; Liu et al. 2023; Nasiriany et al. 2024). Likewise, most robotic benchmarks primarily focus on evaluating policies in short-horizon or atomic tasks (James et al. 2020; Liu et al. 2023; Nasiriany et al. 2024; Mu et al. 2021; Gu et al. 2023; Tao et al. 2024). Although several benchmarks emerged to evaluate long-horizon tasks that leverage memory modeling (Fang et al. 2025; Cherepanov et al. 2025; Morad et al. 2023; Chevalier-Boisvert et al. 2023; Pleines et al. 2025), they largely overlooked the challenges of object-centric memory in POMDP settings.

To address this gap in benchmarking, we introduce **LIBERO-Mem**, a new suite of manipulation tasks specifically designed to evaluate a model's ability to retain object-centric interactions over time. Unlike prior memory benchmarks, LIBERO-Mem emphasizes object-level memory under ambiguity in object identity, location, and relational history, explicitly targeting complex POMDP scenarios. The suite includes four task types: (a) Object Motion, (b) Ob-

*These authors contributed equally.

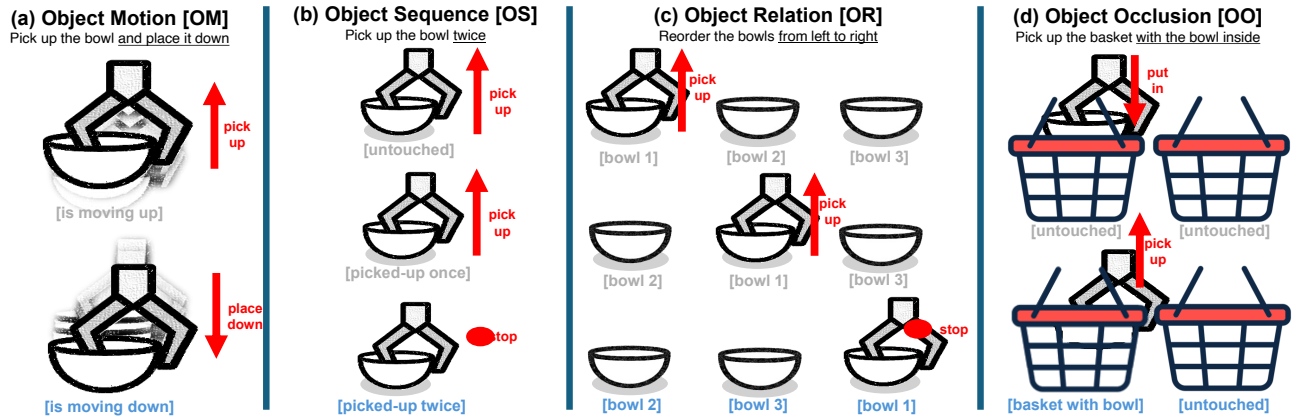

Figure 1: **LIBERO-Mem**: robotic manipulation tasks of object-level POMDP dependencies. These tasks require memory of prior actions and object-specific state tracking beyond what purely Markovian or fully observable policies can handle, highlighting the importance of persistent, object-specific memory for short- and long-horizon reasoning across visually similar inputs. In (a) Object Motion, the robot must recall its last action (e.g., pick up or place down) to act correctly. In (b) Object Sequence, success depends on remembering how many times an object has been manipulated, since visual cues are insufficient. In (c) Object Relation, the robot must track the temporal order of object relations and interactions (e.g., from left to right). In (d) Object Occlusion, occluded objects require the robot to rely on memory of past placements to identify targets.

ject Sequence, (c) Object Relation, and (d) Object Occlusion. Each task type is designed to evaluate different aspects of transient and persistent memory by introducing ambiguities that can only be resolved through temporal reasoning over the history of object interactions. By focusing on memory under uncertainty, LIBERO-Mem takes a key step toward enabling general-purpose robots to operate reliably in everyday settings, where even seemingly simple tasks, such as identifying the correct object to manipulate, become challenging without awareness of prior interactions.

As a step towards overcoming the limitations of existing visuomotor policies, we propose **Embodied-SlotSSM**, a novel VLA framework that integrates slot-based state-space modeling (SSM) (Jiang et al. 2024) to maintain structured, object-centric memory over time. Embodied-SlotSSM encodes temporal dynamics and inter-object interactions into discrete, persistent slots, enabling consistent tracking of entities throughout a task. This structured memory supports robust state estimation and informed decision-making, particularly in POMDP settings where access to interaction history is critical. By grounding both perception and action in visual memory and task intent, Embodied-SlotSSM enables adaptive behavior under real-world uncertainty.

To highlight the limitations of existing works and the challenges posed by our benchmark, we evaluate Embodied-SlotSSM and prior state-of-the-art VLA models on the benchmark of the LIBERO-Goal (Liu et al. 2023), and our proposed LIBERO-Mem benchmark for POMDP tasks. We also carry out studies of an oracle-supported implementation of Embodied-SlotSSM, denoted **Naive E-SlotSSM**, in capturing long-term dependencies towards action prediction, while supporting manipulation effectiveness. Our results highlight the practicality and scalability of slot-based memory, paving the way for more reliable and memory-efficient robotic systems in non-Markovian environments. Our contributions are:

- We introduce LIBERO-Mem, a novel non-Markovian robotic manipulation benchmark that systematically evaluates memory-augmented models on long-horizon tasks, emphasizing object permanence, historical reasoning, and structured memory retention.
- We present Embodied-SlotSSM, a slot-based state-space modeling framework that encodes persistent, object-centric memory representations, enabling structured tracking and decision-making under partial observability.
- We conduct experiments in both general Markovian and special non-Markovian settings via an oracle-supported implementation of Embodied-SlotSSM, denoted Naive E-SlotSSM, showing that Embodied-SlotSSM enhances stateful reasoning, long-horizon action prediction, and performance on manipulation tasks.

We hope future work extends this direction toward bridging VLA design and memory-centric robotics.

Related Works

Robotic Manipulation Benchmarks. Robotic manipulation has been widely benchmarked across several dimensions, including general task completion (Gu et al. 2023; Kumar et al. 2023; Fang et al. 2023), multimodal support (Jiang et al. 2023b; Nasiriany et al. 2024; Liu et al. 2023), and sim-to-real generalization (Tobin et al. 2017; Li et al. 2024b). However, many of these benchmarks, such as RLBench (James et al. 2020), LIBERO (Liu et al. 2023), and RoboCasa (Nasiriany et al. 2024), are constructed under the Markovian assumption, where the robot's next action can be predicted solely from the current observation, without requiring access to historical context. To address this limitation, recent efforts such as MemoryBench (Fang et al. 2025) and MIKASA-Robo (Cherepanov et al. 2025) have highlighted the importance of memory in robotic manipulation tasks. Memory-

Bench introduces a limited set of tasks focusing primarily on spatial memory within short-horizon scenarios. MIKASA-Robo expands this scope by incorporating a broader suite of memory types, including object, spatial, and sequential memory across 32 tasks. However, both benchmarks primarily operate under simplified settings and lack object-level ambiguities or temporal scaling. These efforts demonstrate the growing attention toward non-Markovian reasoning, but fall short in systematically stress-testing object-centric memory under compositional and temporally challenging conditions.

In contrast, we propose LIBERO-Mem, a new benchmark explicitly designed to evaluate long-horizon, object-level memory in partially observable robotic settings. LIBERO-Mem features simple yet composable manipulation tasks that require agents to reason over object identity, spatial configuration, and interaction history. Scenarios such as occluded object recall and sequence-dependent pick-and-place (see Fig. 1) make memory indispensable for success, going beyond what prior benchmarks test. Furthermore, LIBERO-Mem uniquely incorporates stress-testing via temporal scaling, enabling fine-grained assessment of policy robustness over extended episodes - addressing a key gap in existing benchmarks. An overview comparison between our LIBERO-Mem with other benchmarks are summarized in Table 1.

VLA Models in Non-Markovian Settings. End-to-end VLA models have shown strong performance across various embodied tasks, including scene understanding (Jatavallabhula et al. 2023; Gu et al. 2024; Yamazaki et al. 2024), navigation (Hirose et al. 2024; Sridhar et al. 2023; Li et al. 2024a; Tian et al. 2024; Shao et al. 2024), and robotic manipulation (Brohan et al. 2023; Zitkovich et al. 2023; Belkhale et al. 2024; Bharadhwaj et al. 2024; Octo Model Team et al. 2024; Kim et al. 2024; Li et al. 2024c). To enhance grounding, systems like OpenVLA (Kim et al. 2024), Octo (Octo Model Team et al. 2024), and ECoT (Zawalski et al. 2024) leverage powerful pretrained encoders such as DINOv2 (Oquab et al. 2024) and SigLIP (Zhai et al. 2023) for image understanding.

However, despite their success in reactive policy learning, most VLA models operate under an image-based Markovian assumption, treating each observation-action pair independently and have not explicitly modeled interaction history or temporal dependencies. As demonstrated in our experiments, existing VLA models perform poorly on LIBERO-Mem, where partial observability and observation aliasing challenge purely reactive, observation-driven policies. This reveals a fundamental limitation in current VLA architectures: the absence of structured memory and temporal reasoning capabilities necessary for long-horizon control. In addition to introducing LIBERO-Mem as a targeted benchmark to expose these gaps, we also propose an initial solution, grounded in object-centric modeling, that incorporates memory into the policy architecture and shows promising gains under these more realistic, memory-intensive settings.

Object-Centric Learning in Vision and Robotics. Object-centric learning has emerged as a powerful paradigm for extracting modular, interpretable representations and modeling their dynamics across space and time (Goyal et al. 2021; Jiang et al. 2023a). Both supervised and unsupervised methods (Elsayed et al. 2022; Le et al. 2025; Fan et al. 2023) aim to decompose visual inputs into discrete, entity-centric representations, enabling structured reasoning, temporal modeling, and generalization across scenes. A common design across these models involves learning a fixed set of latent "slots" or components that dynamically bind to visual entities (Locatello et al. 2020; Mondal, Cohen, and Webb 2024), facilitating object-level understanding and forecasting. Such representations have proven effective in structured visual environments and have been extended to sequential settings where modeling object dynamics over time is crucial.

Benchmark feature	LIBERO -Mem	Memory Bench	MIKASA -Robo	LIBERO	RLBench
Non-Markovian obs.	✓	✓	✓	×	×
Long-horizon trajectories	✓	✓	×	✓	×
Subgoal-aware evaluation	✓	×	×	×	×
Obj. identity ambiguities	✓	×	×	×	×
Obj. & subgoal annotations	✓	×	×	×	×
Temporal scaling stress-test	✓	×	×	×	×

Table 1: Design factors in robot manipulation benchmarks.

In robotic manipulation, object-centric representations help decompose complex scenes into manipulable entities, allowing for more structured policy learning and goal conditioning. Early approaches often relied on pose-based or category-specific object definitions (Devin et al. 2018; Migimatsu and Bohg 2020; Tyree et al. 2022), but their dependence on supervision limits generalization to novel settings. Recent unsupervised methods aim to segment visual inputs into object-like regions (Locatello et al. 2020; Heravi et al. 2022), offering more flexibility, but they still struggle in cluttered scenes or visually identical objects. Importantly, while these approaches facilitate object-aware perception and control, they are typically designed for short-horizon or episodic tasks, and do not incorporate mechanisms for tracking object identity and state over extended temporal sequences, a capability essential for reasoning in non-Markovian settings.

To address this gap, we draw on cognitive theories of persistent object reasoning (Lam et al. 2020) and slot-based architectures (Jiang et al. 2024; Le et al. 2025), and introduce Embodied-SlotSSM, a memory-centric model that explicitly tracks object-state evolution to support long-horizon manipulation and object-level memorization in POMDP settings such as those presented in our proposed LIBERO-Mem.

Non-Markovian Robot Manipulation

Preliminaries. Instruction-conditioned robotic policies are typically trained to map a sequence of sensory inputs and a language goal into a sequence of actions. A common formulation assumes that the agent can infer the optimal next action from its current visual observation and the instruction alone. While this simplification enables efficient learning and inference, it neglects the temporal structure inherent in many real-world tasks, where past interactions and object histories are essential for disambiguating the current state. Formally, we can consider VLA learning via a demonstration dataset,

$$\mathcal{D} = \{l, \mathbf{a}_{1:T}, \mathbf{v}_{1:T}\}, \quad (1)$$

where l is a natural language instruction, $\mathbf{a}_{1:T} = \{\mathbf{a}_1, \ldots, \mathbf{a}_T\}$ is a sequence of actions, and $\mathbf{v}_{1:T} = \{\mathbf{v}_1, \ldots, \mathbf{v}_T\}$ is a sequence of visual observations. Typically, VLA models (Kim et al. 2024; Zawalski et al. 2024) predict actions from only the current observation and instruction:

$$\hat{\mathbf{a}}_t \sim P_\theta\big(\mathbf{a}_t \mid \mathbf{v}_t, l\big), \tag{2}$$

which implicitly assumes a Markovian process:

$$P\big(\mathbf{a}_t \mid \mathbf{v}_{1:t}, l\big) = P\big(\mathbf{a}_t \mid \mathbf{v}_t, l\big). \tag{3}$$

However, in many practical scenarios, such as cooking, laboratory automation, and industrial assembly, this assumption fails. Robots frequently operate under non-Markovian settings with partial observability (POMDP), where identical visual inputs can correspond to different semantic states depending on prior actions. For instance, a robot may need to remember whether it has already added salt into a container once or twice. This violation of the Markov assumption can be formally expressed by the existence of two timesteps t_1 and t_2 such that $\mathbf{v}_{t_1} \approx \mathbf{v}_{t_2}$, but the correct actions differ due to distinct histories: $P\big(\mathbf{a}_{t_1} \mid \mathbf{v}_{1:t_1}, l\big) \neq P\big(\mathbf{a}_{t_2} \mid \mathbf{v}_{1:t_2}, l\big)$. To act effectively in these settings, an agent must reason over its accumulated experience $(\mathbf{v}_{1:t}, \mathbf{a}_{1:t-1})$ rather than relying solely on instantaneous observations. In this research, we model the problem as an object-centric POMDP, where visually similar object appearances differ in latent history, motivating object-specific memory for temporal reasoning.

LIBERO-Mem: non-Markovian Benchmark

Task designs for object-centric POMDP. LIBERO-Mem consists of 10 tasks spanning four object-centric memory dimensions: Object Motion (OM), Object Sequence (OS), Object Relation (OR), and Object Occlusion (OO). Each task presents temporal dependencies and ambiguity requiring structured memory beyond instantaneous observations. Formally, let $\mathcal{T}_i = \{o_t^i\}_{t=1}^T$ denote a trajectory for task i, where o_t^i is the visual observation at time t. For each task i, there exist at least two trajectories $\mathcal{T}_i^{(1)}$, $\mathcal{T}_i^{(2)}$ such that $\exists\, t_1, t_2$ with $o_{t_1}^{(1)} = o_{t_2}^{(2)}$, yet their underlying task states or required actions differ. This guarantees that visual observations are not uniquely predictive of the correct behavior, thereby enforcing the need for structured memory across time. Please see extended version for full asset details.

Short- and long-horizon data collection process. Expert demonstrations are collected via smooth keyboard control with multi-key tracking. Each task contains 200–700 frames supporting both short- and long-horizon evaluation. Each task is collected to 120 trajectories, where 100 are refined as training data, and 20 are left for validation.

Subgoal-aware evaluation. Each task is decomposed into symbolic subgoals using Sequence ($\rightarrow$) and Or ($\vee$) operators for fine-grained evaluation.

Object identity ambiguities. Visually identical bowls and plates differ only in asset ID, requiring agents to resolve object identity from temporal interaction history.

Object & subgoal annotations. Each timestep includes object instance IDs, masks, subgoal completion flags per object.

Task Num: Description	Subtask Goals	Types
`T1`: robot to pick up the bowl and place it back on the plate	bowl lifted $\rightarrow$ bowl on plate	OM
`T2`: robot to lift the bottle and put it down on the plate	bottle lifted $\rightarrow$ bottle on plate	OM
`T3`: robot to lift the bowl and place it back on the plate 3 times	bowl lifted $\rightarrow$ bowl on plate $\rightarrow$ $\times$ 3	OM, OS
`T4`: robot to pick up the bottle and put it down the plate 3 times	bottle lifted $\rightarrow$ bottle on plate $\rightarrow$ $\times$ 3	OM, OS
`T4`: robot to lift the bowl and place it back on the plate 5 times	bowl lifted $\rightarrow$ bowl on plate $\rightarrow$ $\times$ 5	OM, OS
`T6`: robot to pick up the bowl and place it on the plate 7 times	bowl lifted $\rightarrow$ bowl on plate $\rightarrow$ $\times$ 7	OM, OS
`T7`: robot to swap the 2 bowls on their plates using the empty plate	bowl 1 on plate 3 $\rightarrow$ bowl 2 on plate 1 $\rightarrow$ bowl 1 on plate 2	OM, OR
`T8`: robot to rotate the 3 bowls on their plates from left to right using the empty plate	bowl 1 on plate 4 $\rightarrow$ bowl 2 on plate 1 $\rightarrow$ bowl 3 on plate 2 $\rightarrow$ bowl 1 on plate 3	OM, OR
`T9`: robot to put the cream cheese in the nearest basket and place that basket in the center	bowl 1 in basket 1 $\rightarrow$ basket 1 in center	OM, OO
`T10`: robot to put the cream cheese in the nearest basket and place the empty basket in the center	bowl 1 in basket 1 $\rightarrow$ basket 2 in center	OM, OO

Table 2: LIBERO-Mem tasks, subgoals, and memory.

Can VLAs be scaled temporally for memorization? OpenVLA encodes entire video sequences using 256 dense tokens, while our customized object-centric VLA design operates with only 16 slot tokens, but tokens scale linearly with slot and sequence dimension, leading to intractable memory and compute costs in long-horizon settings. Thus, it motivates us to design a more scalable strategy as Embodied-SlotSSM.

Embodied-SlotSSM

SlotSSM Formulation. An SSM (Gu and Dao 2023; Dao and Gu 2024) aims to approximate $\mathbf{H}_{1:t}$ as $\mathbf{h}_t$ through selective state-space modeling. In particular, the model defines a mapping from an input sequence $\mathbf{e}_{1:T} \in \mathbb{R}^{T\times D}$, encoded from $\mathbf{e}_t = v(\mathbf{o}_t)$ using a pretrained visual encoder $v(\cdot)$, to an output sequence $y_{1:T} \in \mathbb{R}^{T\times D}$ through the following input-dependent recurrence:

$$\mathbf{h}_t = \overline{A}(\mathbf{e}_t)\mathbf{h}_{t-1} + \overline{B}(\mathbf{e}_t)\mathbf{e}_t, \quad y_t = C(\mathbf{e}_t)\mathbf{h}_t \tag{4}$$

where $\mathbf{h}_t \in \mathbb{R}^H$ is the hidden state summarizing the history up to time t, and $\overline{A}(\mathbf{e}_t) \in \mathbb{R}^{H\times H}$, $\overline{B}(\mathbf{e}_t) \in \mathbb{R}^{H\times D}$, and $C(\mathbf{e}_t) \in \mathbb{R}^{D\times H}$ are input-conditioned matrices gen-

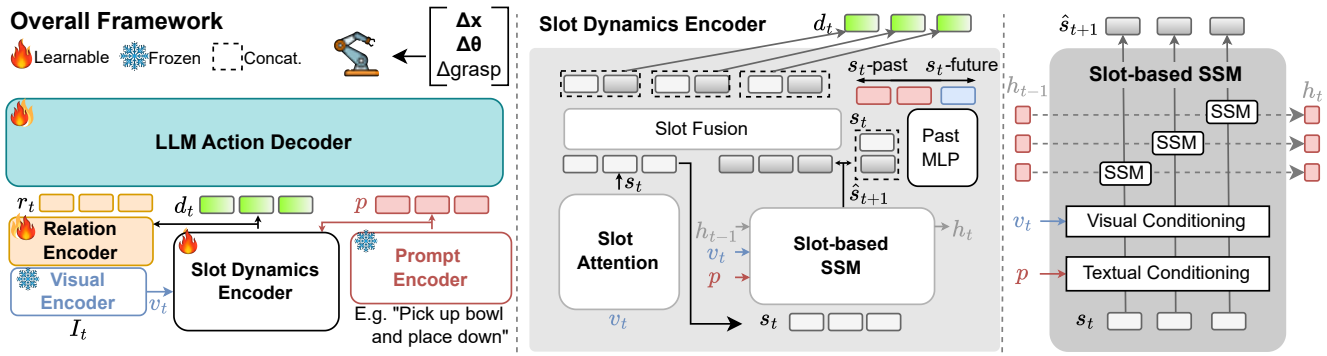

Figure 2: **Embodied-SlotSSM**: Our framework combining slot-based dynamics (Slot Attention, Slot Fusion, Slot-based SSM) with an LLM Action Decoder for object memory-aware action prediction based on textual prompts.

erated by learnable functions applied to $\mathbf{e}_t$. By conditioning the dynamics on the input $\mathbf{e}_t$ at every step, the model flexibly adapts its internal transitions and output mappings based on the current context. As the underlying process of visual representations is inherently modular, SlotSSM (Jiang et al. 2024) is coupled with a slot encoder (Locatello et al. 2020) to decompose $\mathbf{e}_t$ into K individual slot representations $\mathbf{e}_t = [\mathbf{s}_t^1, ..., \mathbf{s}_t^K]$, thereby maintaining $\mathbf{h}_t, \mathbf{y}_t$ also modularly as $\mathbf{h}_t = \text{concat}\,[\mathbf{h}_t^1, ..., \mathbf{h}_t^K]$, and $\mathbf{y}_t = \text{concat}\,[\mathbf{y}_t^1, ..., \mathbf{y}_t^K]$. Thus, the matrices $\overline{A}_t$, $\overline{B}_t$, and C_t are designed to be block-diagonal, where each block is conditioned solely on the corresponding slot input. In our work, $K = 16$ by default.

$$\overline{A}_t = \text{diag}\left(\{\overline{A}(\mathbf{s}_t^k)\}_{k=1}^K\right), \quad \overline{B}_t = \text{diag}\left(\{\overline{B}(\mathbf{s}_t^k)\}_{k=1}^K\right), \quad C_t = \text{diag}\left(\{C(\mathbf{s}_t^k)\}_{k=1}^K\right) \tag{5}$$

Embodied-SlotSSM. Our proposed approach is a slot-based state-space model designed to enable structured, persistent memory for long-horizon visuomotor control. Thus, we make use of individual slot representations $[\mathbf{h}_t^1, ..., \mathbf{h}_t^K]$ for efficient action prediction $\hat{\mathbf{a}}_t \sim P_\theta(\mathbf{a}_t | \mathbf{h}_t^1, ..., \mathbf{h}_t^K, l)$ in a non-Markovian manner. To facilitate action prediction from modular representations and mitigate the memory recall challenges of SSMs (Waleffe et al. 2024; Arora et al. 2024), we propose to model both transient and persistent memory for object-centric representations that can be used for action decoding.

Transient Memory via Temporal Localization

To support short-term reasoning for motion encoding, we model *transient memory* by temporally localizing objects using a combination of Slot Attention and State-Space Modeling. This approach enables the agent to bind scene features to discrete object-centric representations and track their temporal evolution for recent interaction history.

Slot Attention for Object Localization. Manipulation requires disentangling and tracking discrete object entities, yet conventional visual encoders often produce entangled representations that conflate background and object-level signals. To address this, we apply Slot Attention (Locatello et al. 2020) as a tokenization function $g(\cdot)$ that transforms dense visual embeddings $\mathbf{v}_t \in \mathbb{R}^{K \times D_{\text{enc}}}$ into a set of modular object-centric tokens $\mathbf{s}_t = \{\mathbf{s}_t^1, \ldots, \mathbf{s}_t^N\}$, where $\mathbf{s}_t \in \mathbb{R}^{N \times D_{\text{slot}}}$. Slot Attention iteratively binds spatial feature patches to a fixed number of learnable object queries via attention and recurrent updates, yielding disentangled representations that reflect the semantics of $\mathbf{v}_t$ (Xu et al. 2022; Jia, Liu, and Huang 2023). The slot update at time t is computed via multi-head attention followed by a recurrent refinement:

$$\begin{aligned}
\mathbf{a}_{i,j} &= \frac{1}{\sqrt{D_{\text{enc}}}} \mathbf{q}_i \cdot \mathbf{k}_j^\top, \quad \tilde{\mathbf{a}}_{i,j} = \frac{e^{\mathbf{a}_{i,j}}}{\sum_{l=1}^N e^{\mathbf{a}_{l,j}}}, \\
\mathbf{w}_{i,j} &= \frac{\tilde{\mathbf{a}}_{i,j}}{\sum_{l=1}^K \tilde{\mathbf{a}}_{i,l}}, \quad \mathbf{u}_i = \sum_{j=1}^K \mathbf{w}_{i,j} \mathbf{v}_j, \\
\mathbf{s}_t^i &= \text{GRU}(\mathbf{u}_i, \mathbf{s}_t^i),
\end{aligned} \tag{6}$$

where $\mathbf{q}_i$, $\mathbf{k}_j$, and $\mathbf{v}_j$ denote the projected queries, keys, and values. Slot representations are refined by aggregating the input features $\mathbf{v}_t$ and updating via a GRU. To enable temporal coherence in object identity, the slots are initialized at t as:

$$\mathbf{s}_t^{(0)} = \begin{cases} \text{RandomInit}(), & \text{if } t = 0 \\ \mathbf{s}_{t-1}^{(T)}, & \text{if } t > 0, \end{cases} \tag{7}$$

where T denotes the number of recurrent refinement steps per frame. At the beginning of a sequence ($t = 0$), slots are randomly initialized. For all subsequent steps ($t > 0$), the slots are initialized using the final outputs from the previous timestep. This allows the model to propagate slot identity across time and facilitates tracking of persistent objects.

Temporal Contrastive Loss. To further enforce temporal consistency, we employ a contrastive objective over a fixed temporal horizon. Given an anchor slot $\mathbf{s}_t^i$, we treat its corresponding slot in a nearby frame $\mathbf{s}_{t+\delta}^i$ (within the same sequence) as a *positive* and contrast it against slots from different videos or different locations as *negatives*. The contrastive loss is given by:

$$\mathcal{L}_{\text{contrast}} = -\sum_{(i,t)} \log \frac{\sum_{(i',t') \in \mathcal{P}(i,t)} \exp\left(\frac{\text{sim}(\mathbf{s}_t^i, \mathbf{s}_{t'}^{i'})}{\tau}\right)}{\sum_{(j,t'') \in \mathcal{P}(i,t) \cup \mathcal{N}(i,t)} \exp\left(\frac{\text{sim}(\mathbf{s}_t^i, \mathbf{s}_{t''}^{j})}{\tau}\right)}, \tag{8}$$

where $\text{sim}(\cdot, \cdot)$ is cosine similarity with $\tau = 1$.

Slot Dynamics Encoder with SlotSSM

Proposition 1 (Object history enables individuation under visual ambiguity)**.** *Let $\mathcal{O}_t = \{o_t^{(1)}, \ldots, o_t^{(k)}\}$ be k objects at time t with latents $z_t^{(j)} = f_{vis}(v_t^{(j)})$. If $v_t^{(i)} \approx v_t^{(j)}$ for all $i \neq j$, then object identities cannot be distinguished from the current frame and $z_t^{(i)} \approx z_t^{(j)}$. Disambiguation therefore requires a policy $\pi(a_t \mid \mathcal{H}_t)$ that conditions on object-specific histories $\mathcal{H}_t = \{h_t^{(1)}, \ldots, h_t^{(k)}\}$, where $h_t^{(j)} = f_{hist}(z_{1:t}^{(j)})$.*

With f_{vis} as the slot-attention module, we can define f_{hist} via a SlotSSM formulation. Rather than predicting a single next-step embedding, SlotSSM predicts a window of $P = p + q$ object latents, spanning p past and q future representations centered at the current timestep. This reflects the observation that meaningful object behavior often unfolds over short temporal segments rather than single-frame transitions. By modeling a localized temporal window, SlotSSM captures motion continuity, smooths noisy transitions, and anticipates short-term dynamics. The windowed prediction provides rich supervision during training by requiring the model to reconstruct both past and future latent states, such that SlotSSM not only learns forward dynamics (e.g., anticipating object motion) but also enforces temporal consistency via backward reconstruction, thereby improving representation stability and generalization under non-Markovian conditions. Formally, for each slot j, SlotSSM predicts a window of static representations around timestep t as:

$$\left\{\mathbf{z}_{t+\delta}^{(j)}\right\}_{\delta=-p}^{q} = W_{\text{pred}}\left(\hat{\mathbf{s}}_{t+1} || \mathbf{s}_t^{(j)}\right), \tag{9}$$

where $\mathbf{z}_{t+\delta}^{(j)}$ is the set of predicted slots representing the $j-th$ object around the time step t, which is obtained from a shared decoding MLP function, called *Past MLP*, with weights W_{pred} applied independently to each slot. This formulation allows SlotSSM to function as a transient memory module, retaining localized temporal context per object slot to support robust tracking and downstream action decoding.

Action Control through Slot-Conditioned Decoding

In practice, long-horizon observation chunking prevents P from spanning the full sequence due to compute limits; thus, in a naïve implemnetation of Embodied-SlotSSM (which we denote as Naïve E-SlotSSM), we additionally provide an oracle text-embedded subgoal $\mathbf{g}_t^{(j)}$ (e.g., "bowl 1 on plate 3" for swapping tasks such as `T7`) for each relevant object. Given a slot-conditioned decoder then predicts actions from these object-centric features, SlotSSM maintains a latent state $\mathbf{d}_t^{(j)}$ via a *Slot Fusion* module applied to $\mathbf{s}_t^{(j)}$, the predicted next-slot $\hat{\mathbf{s}}_{t+1}^{(j)}$ and oracle subgoal $\mathbf{g}_t^{(j)}$, capturing both current dynamics and temporal context at object level.

Then, to condition action generation on both the structured memory and the current scene, we introduce a lightweight *Relation Encoder* that produces 16 relational tokens (for $K = 16$ slots). This module performs cross-attention between the slot latents $\{\mathbf{d}_t^{(j)}\}_{j=1}^{K}$ and raw visual features $\mathbf{v}_t$ to produce L relation tokens $\{\mathbf{r}_t^{(j)}\}_{j=1}^{L}$, enabling context-aware reasoning over object states and interactions. The final action $\hat{\mathbf{a}}_t$ is decoded through a VLA head conditioned on: (1) the relation tokens $\{\mathbf{r}_t^{(j)}\}_{j=1}^{L}$, (2) the slot dynamics $\{\mathbf{d}_t^{(j)}\}_{j=1}^{K}$, and (3) the current task query embedding l:

$$\hat{\mathbf{a}}_t \sim P_\theta\left(\mathbf{a}_t \mid \{\mathbf{r}_t^{(j)}\}_{j=1}^{L}, \{\mathbf{d}_t^{(j)}\}_{j=1}^{K}, l\right). \tag{10}$$

By grounding actions in object-centric memory and task semantics, the policy gains the ability to reason over identity, relational context, and temporal dependencies, thereby supporting robust performance in partially observable, long-horizon manipulation tasks.

Experiments

Experimental Setup

We evaluate our model on object-centric manipulation tasks using the LIBERO-Goal and the LIBERO-Mem benchmark to respectively evaluate general and object-level POMDP task performance. All experiments are conducted in simulation with fixed scene layouts and randomly initialized object poses. Each task is repeated for N multiple seeds, where action accuracy is computed as the ratio of successful completions over total attempts (success rate $= \frac{\text{n}_0 \text{ success}}{N}$) in Table 3, or subgoal completion $= \frac{\text{subgoals completed}}{\text{total subgoals}}$ over N seeds in Table 4. We evaluate comparatively against Naive E-SlotSSM using OpenVLA (Kim et al. 2024) with slot attention (Locatello et al. 2020) (as object-centric version of SlotVLA (Hanyu et al. 2025)), π_0 (Black et al. 2024), each with horizon h denoting number of input frames, over $N = 20$.

General Task Performance: Our empirical Naive E-SlotSSM achieves the highest success rates on LIBERO-Goal, outperforming slot-based baselines. The results in Table 3 highlight the effectiveness of structured object-centric memory in non-Markovian manipulation tasks. While SlotVLA ($h = 8$) achieves moderate performance (75.5%) by extending temporal context length, it still struggles on tasks involving multi-object interactions and occlusions (e.g., *middle drawer open*, *top drawer open → bowl in*). In contrast, Naive E-SlotSSM consistently outperforms others ($\mathbf{83.0}\%$ avg), demonstrating robustness across both simple and temporally entangled tasks. Notably, it handles action sequences involving relational reasoning (e.g., *bottle in rack*) and spatial displacements (e.g., *bowl in plate*) reliably, attributed to its ability to integrate persistent slot tracking memory. These results affirm the usefulness of memory design for generalizing over object-centric tasks in general manipulation.

Object POMDP Task Performance with LIBERO-Mem: The empirical Naive E-SlotSSM also achieves the highest subgoal success rates on LIBERO-Goal, outperforming the baselines. Table 4 reveals the difficulty of fine-grained subgoal tracking in non-Markovian object manipulation. Both dense-token baseline (π_0) and SlotVLA models ($h = 1$, $h = 8$) achieve minimal subgoal completion (avg. 5.0%), failing to reason over sequences such as repeated placements or multi-step swaps. In contrast, Embodied-SlotSSM yields a significantly higher average subgoal completion of 14.8%, with partial successes in long-horizon repetitive actions (e.g.,

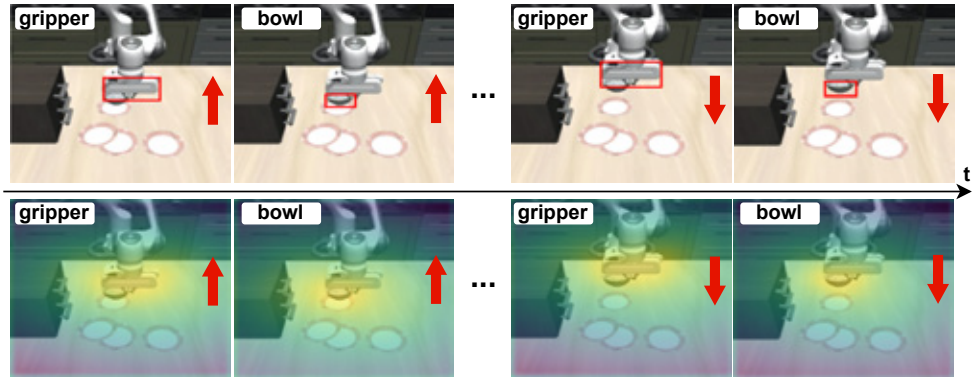

Figure 3: Slot visualization in task `T1`: gripper and bowl slots (bbox, attention) as robot lifts and places the bowl down.

Task (token n_0)	SlotVLA (h=1) (16)	SlotVLA (h=8) (128)	Naive. SlotSSM (32)
bowl in stove	45%	95%	100%
bowl in cabinet	75%	90%	90%
plate on front-of-stove	70%	80%	95%
stove turned on	70%	95%	100%
bottle on cabinet	20%	90%	90%
bowl on plate	0%	90%	90%
mid drawer open	5%	25%	45%
cheese in bowl	25%	55%	75%
bottle on rack	10%	70%	75%
top drawer open → bowl in	0%	65%	70%
Average	32.0%	75.5%	**83.0%**

Table 3: Task success rates on LIBERO-Goal.

Task (token n_0)	π_0 (h=1) (256)	SlotVLA (h=1) (16)	SlotVLA (h=8) (128)	Naive. SlotSSM (32)
`T1`	50.0%	0%	50.0%	50.0%
`T2`	0%	0%	0%	0%
`T3`	0%	0%	0%	33.3%
`T4`	0%	0%	0%	0%
`T5`	0%	0%	0%	14.3%
`T6`	0%	0%	0%	0%
`T7`	0%	0%	0%	0%
`T8`	0%	0%	0%	0%
`T9`	0%	0%	0%	30%
`T10`	0%	0%	0%	20%
Average	5.0%	0%	5.0%	**14.8%**

Table 4: Subgoal completion percentages on LIBERO-Mem.

50% on `T1` of pick and place bowl (1x), and 33.3% on the repeated (3x) version. These gains suggest that slot-based temporal memory, especially when aligned with object and subgoal states, provides a strong inductive bias for step-wise reasoning under partial observability. Nonetheless, as persistent memory modeling remains modest by relying on an object-level subgoal monitor to track progress and monitor progress performance, indicating that subgoal grounding is a key open challenge in real-world non-Markovian settings.

Qualitative Analyses: Like SlotVLA (Hanyu et al. 2025), we visualize slot attention across time to observe whether the model consistently attends to the same object instance, especially under occlusion or motion. Visualizations (Figure 3) reveal that our model maintains consistent attention to target objects over time. This indicates the emergence of robust object permanence and tracking, which can be critical for long-horizon reasoning. More qualitative analyses of the trajectories will be shown in the extended version.

Conclusion

The complexity of real-world environments requires agents to reason over past interactions, especially when handling multiple similar objects. Such settings demand persistent object-specific memory and introduce non-Markovian dependencies beyond reactive control. To study this challenge, we introduced LIBERO-Mem, a benchmark for memory-intensive robotic manipulation featuring long-horizon, temporally entangled, and repetitive subgoals that test robust memory rather than short-term perception. We further proposed Embodied-SlotSSM, a structured memory model maintaining spatio-temporally consistent slot representations that track object identity and state, thereby enabling temporally grounded, context-aware action prediction. Experiments on LIBERO-Mem show that the proposed method surpasses memory-less baselines, thereby advancing object-centric decision-making in non-Markovian environments and encouraging scalable memory-based VLAs grounded in structured representations.

Limitations: LIBERO-Mem is a simulated setting for future physical extension. The empirical version of Embodied-SlotSSM (Naive E-SlotSSM) currently serves as a weak baseline that leverages oracle subgoal representations rather than inferring it autonomously, leaving open the challenge of self-discovered subgoal reasoning and POMDP at object level.

Broader Impact: LIBERO-Mem and Embodied-SlotSSM advance memory-aware, object-centric visuomotor systems for real-world settings, helping robots track task structure and avoid redundant actions in daily and industrial settings.

References

Arora, S.; Eyuboglu, S.; Zhang, M.; Timalsina, A.; Alberti, S.; Zou, J.; Rudra, A.; and Ré, C. 2024. Simple linear attention language models balance the recall-throughput tradeoff. In *ICML*.

Belkhale, S.; Ding, T.; Xiao, T.; Sermanet, P.; Vuong, Q.; Tompson, J.; Chebotar, Y.; Dwibedi, D.; and Sadigh, D. 2024. RT-H: Action Hierarchies Using Language. *CoRR*, abs/2403.01823.

Bharadhwaj, H.; Vakil, J.; Sharma, M.; Gupta, A.; Tulsiani, S.; and Kumar, V. 2024. RoboAgent: Generalization and Efficiency in Robot Manipulation via Semantic Augmentations and Action Chunking. In *ICRA*.

Black, K.; Brown, N.; Driess, D.; Esmail, A.; Equi, M.; Finn, C.; Fusai, N.; Groom, L.; Hausman, K.; Ichter, B.; Jakubczak, S.; Jones, T.; Ke, L.; Levine, S.; Li-Bell, A.; Mothukuri, M.; Nair, S.; Pertsch, K.; Shi, L. X.; Tanner, J.; Vuong, Q.; Walling, A.; Wang, H.; and Zhilinsky, U. 2024. π_0: A Vision-Language-Action Flow Model for General Robot Control. *CoRR*, abs/2410.24164.

Brohan, A.; Brown, N.; Carbajal, J.; Chebotar, Y.; et al. 2023. RT-1: Robotics Transformer for Real-World Control at Scale. In *RSS*.

Cherepanov, E.; Kachaev, N.; Kovalev, A. K.; and Panov, A. I. 2025. Memory, Benchmark & Robots: A Benchmark for Solving Complex Tasks with Reinforcement Learning. *CoRR*, abs/2502.10550.

Chevalier-Boisvert, M.; Dai, B.; Towers, M.; Perez-Vicente, R.; Willems, L.; Lahlou, S.; Pal, S.; Castro, P. S.; and Terry, J. 2023. Minigrid & miniworld: Modular & customizable reinforcement learning environments for goal-oriented tasks. *NeurIPS*, 36: 73383–73394.

Dao, T.; and Gu, A. 2024. Transformers are SSMs: Generalized Models and Efficient Algorithms Through Structured State Space Duality. In *ICML*.

Devin, C.; Abbeel, P.; Darrell, T.; and Levine, S. 2018. Deep object-centric representations for generalizable robot learning. In *ICRA*, 7111–7118. IEEE.

Elsayed, G.; Mahendran, A.; van Steenkiste, S.; Greff, K.; Mozer, M. C.; and Kipf, T. 2022. SAVi++: Towards End-to-End Object-Centric Learning from Real-World Videos. In *NeurIPS*.

Fan, K.; Bai, Z.; Xiao, T.; Zietlow, D.; Horn, M.; Zhao, Z.; Simon-Gabriel, C.; Shou, M. Z.; Locatello, F.; Schiele, B.; Brox, T.; Zhang, Z.; Fu, Y.; and He, T. 2023. Unsupervised Open-Vocabulary Object Localization in Videos. In *ICCV*.

Fang, H.; Grotz, M.; Pumacay, W.; Wang, Y. R.; Fox, D.; Krishna, R.; and Duan, J. 2025. SAM2Act: Integrating Visual Foundation Model with A Memory Architecture for Robotic Manipulation. *CoRR*, abs/2501.18564.

Fang, H.-S.; Fang, H.; Tang, Z.; Liu, J.; Wang, J.; Zhu, H.; and Lu, C. 2023. RH20T: A Robotic Dataset for Learning Diverse Skills in One-Shot. In *RSS 2023 Workshop on Learning for Task and Motion Planning*.

Goyal, A.; Lamb, A.; Hoffmann, J.; Sodhani, S.; Levine, S.; Bengio, Y.; and Schölkopf, B. 2021. Recurrent Independent Mechanisms. In *ICLR*.

Gu, A.; and Dao, T. 2023. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. *arXiv preprint arXiv:2312.00752*.

Gu, J.; Xiang, F.; Li, X.; Ling, Z.; Liu, X.; Mu, T.; Tang, Y.; Tao, S.; Wei, X.; Yao, Y.; et al. 2023. Maniskill2: A unified benchmark for generalizable manipulation skills. *arXiv preprint arXiv:2302.04659*.

Gu, Q.; Kuwajerwala, A.; Morin, S.; Jatavallabhula, K. M.; Sen, B.; Agarwal, A.; Rivera, C.; Paul, W.; Ellis, K.; Chellappa, R.; et al. 2024. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. In *ICRA*.

Hanyu, T.; Chung, N.; Le, H.; Nguyen, T.; Ikebe, Y.; Gunderman, A.; Minh, D. N. H.; Vo, K.; Kieu, T.; Yamazaki, K.; Rainwater, C.; Nguyen, A.; and Le, N. 2025. SlotVLA: Towards Modeling of Object-Relation Representations in Robotic Manipulation. *arXiv preprint arXiv:2511.06754*.

Heravi, N.; Wahid, A.; Lynch, C.; Florence, P.; Armstrong, T.; Tompson, J.; Sermanet, P.; Bohg, J.; and Dwibedi, D. 2022. Visuomotor control in multi-object scenes using object-aware representations. *arXiv preprint arXiv:2205.06333*.

Hirose, N.; Glossop, C.; Sridhar, A.; Mees, O.; and Levine, S. 2024. LeLaN: Learning A Language-Conditioned Navigation Policy from In-the-Wild Video. In *CoRL*.

James, S.; Ma, Z.; Arrojo, D. R.; and Davison, A. J. 2020. Rlbench: The robot learning benchmark & learning environment. *IEEE Robotics and Automation Letters*.

Jatavallabhula, K. M.; Kuwajerwala, A.; Gu, Q.; Omama, M.; Chen, T.; Maalouf, A.; Li, S.; Iyer, G.; Saryazdi, S.; Keetha, N.; et al. 2023. Conceptfusion: Open-set multimodal 3d mapping. *arXiv preprint arXiv:2302.07241*.

Jia, B.; Liu, Y.; and Huang, S. 2023. Improving Object-centric Learning with Query Optimization. In *ICLR*.

Jiang, J.; Deng, F.; Singh, G.; and Ahn, S. 2023a. Object-Centric Slot Diffusion. In *NeurIPS*.

Jiang, J.; Deng, F.; Singh, G.; Lee, M.; and Ahn, S. 2024. Slot State Space Models. In *NeurIPS*.

Jiang, Y.; Gupta, A.; Zhang, Z.; Wang, G.; Dou, Y.; Chen, Y.; Fei-Fei, L.; Anandkumar, A.; Zhu, Y.; and Fan, L. 2023b. VIMA: General Robot Manipulation with Multimodal Prompts. In *ICML*.

Khazatsky, A.; Pertsch, K.; Nair, S.; Balakrishna, A.; Dasari, S.; et al. 2024. DROID: A large-scale in-the-wild robot manipulation dataset. In *RSS*.

Kim, M. J.; Pertsch, K.; Karamcheti, S.; Xiao, T.; Balakrishna, A.; Nair, S.; Rafailov, R.; Foster, E. P.; Sanketi, P. R.; Vuong, Q.; Kollar, T.; Burchfiel, B.; Tedrake, R.; Sadigh, D.; Levine, S.; Liang, P.; and Finn, C. 2024. OpenVLA: An Open-Source Vision-Language-Action Model. In *CoRL*.

Kumar, V.; Shah, R.; Zhou, G.; Moens, V.; Caggiano, V.; Gupta, A.; and Rajeswaran, A. 2023. RoboHive: A Unified Framework for Robot Learning. In *NeurIPS*, volume 36.

Lam, K. C.; Pereira, F.; Vaziri-Pashkam, M.; Woodard, K.; and McMahon, E. 2020. Mental representations of objects reflect the ways in which we interact with them. In *Annual Meeting of the Cognitive Science Society*.

Le, H.; Chung, N.; Kieu, T.; Yang, J.; and Le, N. 2025. UNO: Unifying One-stage Video Scene Graph Generation via Object-Centric Visual Representation Learning. *CoRR*, abs/2509.06165.

Li, B.; Wang, Y.; Mao, J.; Ivanovic, B.; Veer, S.; Leung, K.; and Pavone, M. 2024a. Driving Everywhere with Large Language Model Policy Adaptation. In *CVPR*.

Li, X.; Hsu, K.; Gu, J.; Mees, O.; Pertsch, K.; Walke, H. R.; Fu, C.; Lunawat, I.; Sieh, I.; Kirmani, S.; Levine, S.; Wu, J.; Finn, C.; Su, H.; Vuong, Q.; and Xiao, T. 2024b. Evaluating Real-World Robot Manipulation Policies in Simulation. In *CoRL*.

Li, X.; Liu, M.; Zhang, H.; Yu, C.; Xu, J.; Wu, H.; Cheang, C.; Jing, Y.; Zhang, W.; Liu, H.; Li, H.; and Kong, T. 2024c. Vision-Language Foundation Models as Effective Robot Imitators. In *ICLR*.

Liu, B.; Zhu, Y.; Gao, C.; Feng, Y.; Liu, Q.; Zhu, Y.; and Stone, P. 2023. LIBERO: Benchmarking Knowledge Transfer for Lifelong Robot Learning. In *NeurIPS*.

Locatello, F.; Weissenborn, D.; Unterthiner, T.; Mahendran, A.; Heigold, G.; Uszkoreit, J.; Dosovitskiy, A.; and Kipf, T. 2020. Object-Centric Learning with Slot Attention. In *NeurIPS*.

Migimatsu, T.; and Bohg, J. 2020. Object-centric task and motion planning in dynamic environments. *RA-L*.

Mondal, S. S.; Cohen, J. D.; and Webb, T. W. 2024. Slot Abstractors: Toward Scalable Abstract Visual Reasoning. In *ICML*.

Morad, S.; Kortvelesy, R.; Bettini, M.; Liwicki, S.; and Prorok, A. 2023. Popgym: Benchmarking partially observable reinforcement learning. *arXiv preprint arXiv:2303.01859*.

Mu, T.; Ling, Z.; Xiang, F.; Yang, D.; Li, X.; Tao, S.; Huang, Z.; Jia, Z.; and Su, H. 2021. Maniskill: Generalizable manipulation skill benchmark with large-scale demonstrations. *arXiv preprint arXiv:2107.14483*.

Nasiriany, S.; Maddukuri, A.; Zhang, L.; Parikh, A.; Lo, A.; Joshi, A.; Mandlekar, A.; and Zhu, Y. 2024. RoboCasa: Large-Scale Simulation of Everyday Tasks for Generalist Robots. In *RSS*.

Octo Model Team; Ghosh, D.; Walke, H.; Pertsch, K.; Black, K.; Mees, O.; Dasari, S.; Hejna, J.; Xu, C.; Luo, J.; Kreiman, T.; Tan, Y.; Sanketi, P.; Vuong, Q.; Xiao, T.; Sadigh, D.; Finn, C.; and Levine, S. 2024. Octo: An Open-Source Generalist Robot Policy. In *RSS*. Delft, Netherlands.

O'Neill, A.; Rehman, A.; Maddukuri, A.; et al. 2024. Open X-Embodiment: Robotic Learning Datasets and RT-X Models : Open X-Embodiment Collaboration. In *ICRA*.

Oquab, M.; Darcet, T.; Moutakanni, T.; Vo, H. V.; Szafraniec, M.; Khalidov, V.; Fernandez, P.; HAZIZA, D.; Massa, F.; El-Nouby, A.; Assran, M.; Ballas, N.; Galuba, W.; Howes, R.; Huang, P.-Y.; Li, S.-W.; Misra, I.; Rabbat, M.; Sharma, V.; Synnaeve, G.; Xu, H.; Jegou, H.; Mairal, J.; Labatut, P.; Joulin, A.; and Bojanowski, P. 2024. DINOv2: Learning Robust Visual Features without Supervision. *TMLR*.

Pleines, M.; Pallasch, M.; Zimmer, F.; and Preuss, M. 2025. Memory Gym: Towards Endless Tasks to Benchmark Memory Capabilities of Agents. *JMLR*, 26: 1–40.

Shao, H.; Hu, Y.; Wang, L.; Song, G.; Waslander, S. L.; Liu, Y.; and Li, H. 2024. LMDrive: Closed-Loop End-to-End Driving with Large Language Models. In *CVPR*.

Sridhar, A. K.; Shah, D.; Glossop, C.; and Levine, S. 2023. NoMaD: Goal Masked Diffusion Policies for Navigation and Exploration. *ICRA*.

Tao, S.; Xiang, F.; Shukla, A.; Qin, Y.; Hinrichsen, X.; Yuan, X.; Bao, C.; Lin, X.; Liu, Y.; Chan, T.; Gao, Y.; Li, X.; Mu, T.; Xiao, N.; Gurha, A.; Huang, Z.; Calandra, R.; Chen, R.; Luo, S.; and Su, H. 2024. ManiSkill3: GPU Parallelized Robotics Simulation and Rendering for Generalizable Embodied AI. *CoRR*, abs/2410.00425.

Tian, T.; Li, B.; Weng, X.; Chen, Y.; Schmerling, E.; Wang, Y.; Ivanovic, B.; and Pavone, M. 2024. Tokenize the World into Object-level Knowledge to Address Long-tail Events in Autonomous Driving. In *CoRL*.

Tobin, J.; Fong, R.; Ray, A.; Schneider, J.; Zaremba, W.; and Abbeel, P. 2017. Domain randomization for transferring deep neural networks from simulation to the real world. In *IROS*.

Tyree, S.; Tremblay, J.; To, T.; Cheng, J.; Mosier, T.; Smith, J.; and Birchfield, S. 2022. 6-dof pose estimation of household objects for robotic manipulation: An accessible dataset and benchmark. In *IROS*, 13081–13088. IEEE.

Waleffe, R.; Byeon, W.; Riach, D.; Norick, B.; Korthikanti, V.; Dao, T.; Gu, A.; Hatamizadeh, A.; Singh, S.; Narayanan, D.; Kulshreshtha, G.; Singh, V.; Casper, J.; Kautz, J.; Shoeybi, M.; and Catanzaro, B. 2024. An Empirical Study of Mamba-based Language Models. *CoRR*, abs/2406.07887.

Walke, H. R.; Black, K.; Zhao, T. Z.; Vuong, Q.; Zheng, C.; Hansen-Estruch, P.; He, A. W.; Myers, V.; Kim, M. J.; Du, M.; Lee, A.; Fang, K.; Finn, C.; and Levine, S. 2023. BridgeData V2: A Dataset for Robot Learning at Scale. In *CoRL*.

Wang, L.; Chen, X.; Zhao, J.; and He, K. 2024. Scaling Proprioceptive-Visual Learning with Heterogeneous Pre-trained Transformers. In *NeurIPS*.

Xu, J.; De Mello, S.; Liu, S.; Byeon, W.; Breuel, T.; Kautz, J.; and Wang, X. 2022. Groupvit: Semantic segmentation emerges from text supervision. In *CVPR*.

Yamazaki, K.; Hanyu, T.; Vo, K.; Pham, T.; Tran, M.; Doretto, G.; Nguyen, A.; and Le, N. 2024. Open-fusion: Real-time open-vocabulary 3d mapping and queryable scene representation. In *ICRA*, 9411–9417. IEEE.

Yang, J. H.; Glossop, C.; Bhorkar, A.; Shah, D.; Vuong, Q.; Finn, C.; Sadigh, D.; and Levine, S. 2024. Pushing the Limits of Cross-Embodiment Learning for Manipulation and Navigation. In *RSS*.

Zawalski, M.; Chen, W.; Pertsch, K.; Mees, O.; Finn, C.; and Levine, S. 2024. Robotic Control via Embodied Chain-of-Thought Reasoning. In *CoRL*.

Zhai, X.; Mustafa, B.; Kolesnikov, A.; and Beyer, L. 2023. Sigmoid Loss for Language Image Pre-Training. In *ICCV*.

Zitkovich, B.; Yu, T.; Xu, S.; et al. 2023. RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control. In *CoRL*.