PHYSICAL WORDS: A VOCABULARY FOR STRUCTURING LATENT REASONING IN ROBOT MANIPULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal reasoning provides robotic foundation models with predictive context needed for reliable action generation. However, language-based reasoning captures task semantics but struggles to express fine-grained physical dynamics, while latent reasoning models future state evolution but offers limited structure for organizing these dynamics into reusable reasoning primitives. To bridge this gap, we introduce **PhysWord**, a reasoning-before-acting foundation model built on **Physical Words**, which use atomic action verbs as shared semantic anchors for continuous representations of physical dynamics. Specifically, PhysWord predicts upcoming atomic action words and directly aligns their contextual hidden states with latent targets derived from the corresponding future observations. Building on these primitives, we construct a temporally structured PhysWord space with visual and optional tactile variants of each action word. Autoregressively predicting Physical Words across future timesteps captures transitions between execution stages, providing structured multimodal context for action generation. PhysWord achieves state-of-the-art success rates of 90.3% on visuotactile UniVTAC and 99.6% on vision-based LIBERO-Long, and outperforms competing methods on real-world tasks using bimanual grippers and dexterous hands. Further evaluations demonstrate that our structured latent reasoning framework generalizes to unseen scenarios and enables long-horizon manipulation.

1 INTRODUCTION

Vision-Language-Action (VLA) models have emerged as a promising foundation for general-purpose robotic manipulation (Kim et al., 2024; Liu et al., 2025a), integrating the semantic knowledge of pretrained vision-language models (VLMs) (Karamcheti et al., 2024; Bai et al., 2025; Beyer et al., 2024) with robot action learning. Reliable manipulation, however, requires anticipating how the physical environment evolves as the robot acts. This need has motivated reasoning-before-acting VLA approaches (Huang et al., 2025; Liu et al., 2026), which generate intermediate task plans or predictive context to guide action generation. World Action Models (WAMs) further integrate future world prediction with action generation, enabling policies to leverage dynamics knowledge learned from video and manipulation data. These advances highlight the value of multimodal reasoning in providing predictive context that aligns robot actions with evolving environments.

Existing reasoning approaches differ in how they represent predictive context. On the one hand, some VLA models adopt the reasoning paradigm of pretrained VLMs, using language to decompose tasks and express task-level action semantics (Physical Intelligence et al., 2025; Lin et al., 2025; Zawalski et al., 2025). However, linguistic descriptions remain physically coarse, as a verb such as "rotate" identifies an operation without fully conveying its fine-grained dynamics. On the other hand, WAMs (Ye et al., 2026b; Wang et al., 2026) and latent reasoning methods (Liu et al., 2026; Bi et al., 2025; Tian et al., 2025) model future state evolution, providing rich predictive representations of interaction dynamics. Yet these approaches do not necessarily organize their representations around shared action semantics. Since the same manipulation skill can be applied across different objects and scenes, a useful reasoning representation should preserve shared semantics while capturing context-specific physical dynamics. This raises a central question: Can atomic action semantics organize physical latent representations into reusable reasoning primitives?

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

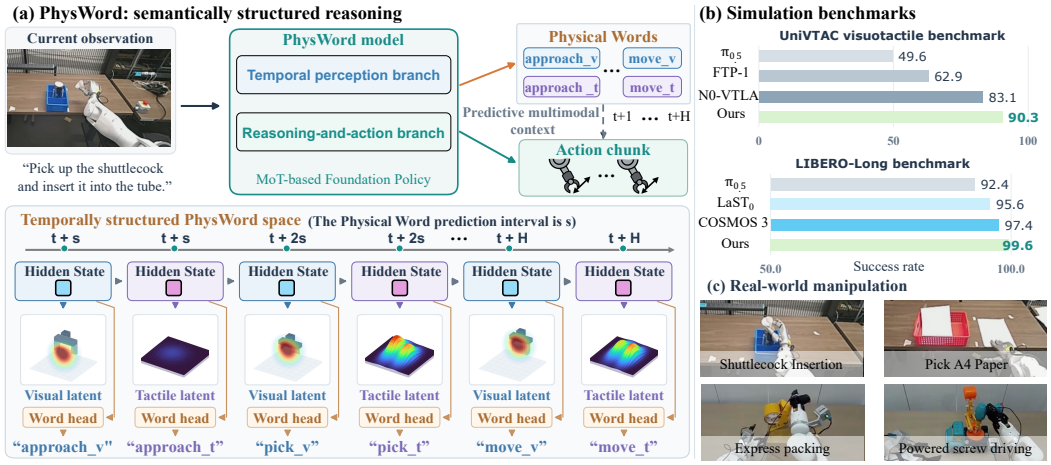


Figure 1: **Overview of PhysWord.** (a) PhysWord anchors continuous physical dynamics to atomic action words, with separate visual and tactile variants learned through direct latent supervision. Autoregressively generating Physical Words within a temporally structured space provides multimodal predictive context for action generation. (b) PhysWord achieves SOTA performance on both the visuotactile and vision-based simulation benchmarks. (c) Real-world experiments span bimanual gripper and dexterous-hand systems, both equipped with tactile sensing.

To this end, we introduce PhysWord, a reasoning-before-acting foundation model built on Physical Words, which use atomic action verbs as semantic anchors for continuous latent dynamics. We assemble a vocabulary of atomic action verbs, such as "pick," "rotate," and "push," based on the action taxonomy of prior work (Yuan et al., 2026b; Chen et al., 2025b). As shown in Figure 1(a), PhysWord predicts upcoming atomic action words from the current observation context and directly aligns their contextual hidden states with latent targets derived from the corresponding future observations. Joint semantic and latent supervision encourages these representations to capture shared action semantics while retaining action-related physical dynamics. To capture how these dynamics evolve during execution, we organize Physical Words at multiple future timesteps within the action-chunk horizon, forming a temporally structured PhysWord space. Within this space, we use separate visual and tactile variants of each action word to capture complementary motion and contact dynamics, including tactile variants only for tactile-enabled tasks. By autoregressively predicting these Physical Words, PhysWord captures changes in action semantics and physical dynamics across execution stages, providing structured multimodal predictive context for long-horizon manipulation. This token interface adapts to the available sensory modalities, bringing demonstrations with and without tactile observations into a unified training formulation. This flexibility enables us to jointly pretrain PhysWord on approximately 7,000 hours of real-robot and human-hand demonstrations. Across this corpus, we use a VLM (OpenAI, 2026) to automatically segment trajectories into atomic action stages and align the resulting atomic verbs with the corresponding observations.

As shown in Figure 1(b), PhysWord achieves success rates of 90.3% on visuotactile UniVTAC and 99.6% on vision-only LIBERO-Long, outperforming prior SOTA methods. Real-world experiments on bimanual gripper and dexterous-hand platforms with visual and tactile sensing further demonstrate its effectiveness in contact-rich, precise, and long-horizon manipulation. To assess reusability, we evaluate both unseen scene configurations and extended task sequences absent from training. PhysWord outperforms competing methods under unseen backgrounds, object instances, and object positions on *Powered Screw Driving* and *Pick A4 Paper*, and successfully executes extended manipulation sequences, supporting reuse across both scenarios and task stages. To assess the predictive value of its tactile representations, we replace measured tactile inputs with predicted latents at inference. Remarkably, the resulting policies retain strong performance with only modest degradation, suggesting that semantic structuring preserves contact information needed for control and enables deployment without online tactile sensing. **Our contributions are summarized as follows:**

- We introduce PhysWord, a reasoning-before-acting foundation model built on Physical Words, which use atomic action verbs as semantic anchors for continuous physical dynamics by directly supervising their contextual hidden states with future latent targets.
- We construct a temporally structured PhysWord space with visual and tactile variants of atomic action words, capturing evolving motion and contact dynamics for action generation.

- PhysWord achieves 90.3% on UniVTAC and 99.6% on LIBERO-Long, outperforming prior SOTA methods. Experiments on real-world tasks demonstrate superior performance, stronger generalization, and long-horizon manipulation through sequential reuse of Physical Words.

2 Related Work

Vision-Language-Action (VLA) Models. VLA models (Kim et al., 2024; Black et al., 2024; Li et al., 2023; Bjorck et al., 2025; Liu et al., 2025a; Wen et al., 2025) adapt pretrained VLMs to generate robot actions from visual observations and language instructions. Reasoning-before-acting VLAs introduce intermediate reasoning to anticipate task progression. Language-based approaches, including subtask prediction, embodied chain-of-thought (CoT), and motion descriptions (Physical Intelligence et al., 2025; Lin et al., 2025; Zawalski et al., 2025; Belkhale et al., 2024), convey task semantics and subtask plans but struggle to express fine-grained dynamics. Beyond language, recent methods predict future visual observations (Wang et al., 2025b; Zhao et al., 2025; Chen et al., 2025a; Gao et al., 2025; Bu et al., 2025; Zhang et al., 2025) or incorporate anticipated signals from multiple modalities (Liu et al., 2025b; Cen et al., 2025; Gu et al., 2025).

Latent Reasoning and World Action Models (WAMs). Latent CoT represents intermediate reasoning through continuous latent states in LLMs and VLMs (Hao et al., 2024; Deng et al., 2024; Yang et al., 2025; Wang et al., 2025a). ATLAS (Guo et al., 2026) further highlights the importance of LLMs' native language-based CoT for general reasoning tasks. In robotics, recent VLA models adopt latent reasoning to guide control using compact representations of future physical dynamics (Zhang et al., 2026; Huang et al., 2025; Tan et al., 2025). Related WAMs jointly model video and action sequences (Kim et al., 2026; Agarwal et al., 2026; Ye et al., 2026b; Wang et al., 2026). Several approaches condition action generation on latent representations of predicted future states (Hu et al., 2024; Yuan et al., 2026c; Luo et al., 2026). Although these approaches capture rich future dynamics, their latent representations are not structured around shared action semantics. PhysWord introduces this structure by anchoring future visual and tactile dynamics to atomic action verbs, forming reusable reasoning primitives that provide predictive context for action generation.

3 Methodology

In this section, we present PhysWord for structured latent reasoning in robotic manipulation, including its architecture (Section 3.2), Physical Word formulation (Section 3.3), temporally structured PhysWord space (Section 3.4), and training objectives (Section 3.5).

3.1 Preliminaries

Problem Formulation. We consider language-conditioned robotic manipulation with visual observations and optional tactile sensing. At time t, the robot receives a language instruction ℓ, multi-view visual observations $\mathscr{V}_t$, and, when available, tactile observations $\mathscr{T}_t$. We denote the current observations and their temporal history by $\mathscr{O}_t$. Conditioned on this context, the policy predicts an action chunk: $\mathbf{A}_t = (\mathbf{a}_{t+1}, \ldots, \mathbf{a}_{t+H}) \sim \pi_\theta(\cdot \mid \mathscr{O}_t, \ell)$, where H denotes the action prediction horizon and $\mathbf{a}_t$ is an embodiment-specific control vector. For dual-arm gripper platforms, each action is 14-DOF, comprising a 3-DOF relative translation, a 3-DOF Euler-angle rotation, and a binary gripper command for each arm. For dexterous-hand platforms, each action is 56-DOF, comprising 12 end-effector pose components across the two arms and 44 joint commands across the two Sharpa hands.

3.2 PhysWord Model Architecture

PhysWord adopts a mixture-of-transformers (MoT) architecture with a temporal perception branch initialized from Cosmos-Predict2.5-2B (Ali et al., 2025) and a reasoning-and-action branch initialized from Qwen3-VL-2B (Bai et al., 2025). As shown in Figure 2 (a), the temporal branch encodes visual history and anticipated scene evolution, while the reasoning-and-action branch generates Physical Words followed by robot actions.

Multimodal inputs. The temporal perception branch receives five head-view observations from $t-4$ to t, each at a resolution of 256×256, and predicts the next H frames. The reasoning-and-action

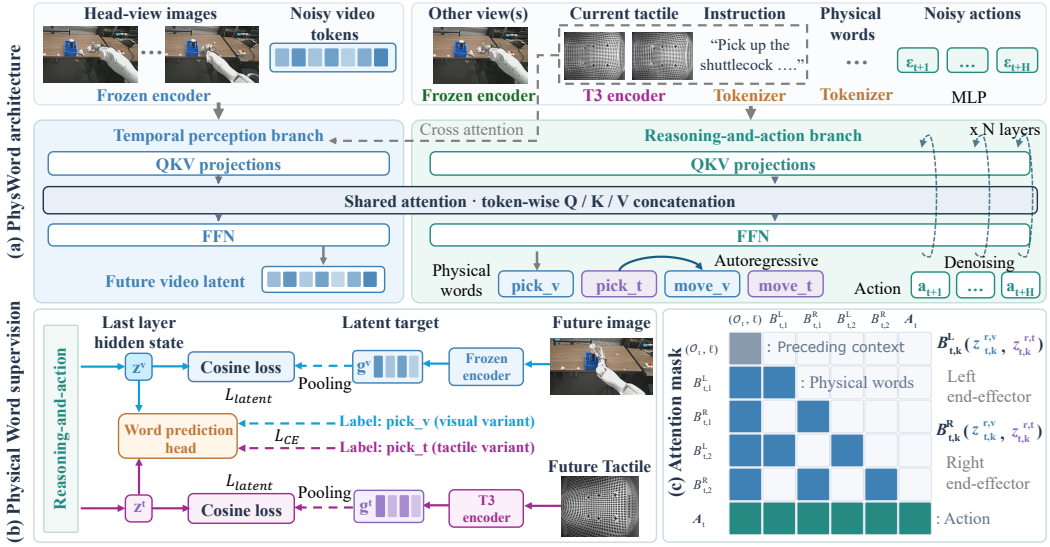


Figure 2: **PhysWord framework.** (a) PhysWord connects temporal perception and reasoning-and-action branches through shared attention in a MoT architecture, generating Physical Words autoregressively before denoising actions. (b) The hidden states of visual and tactile Physical Word variants receive joint action-label cross-entropy supervision and cosine alignment with future sensory latent targets. (c) Physical Words use a blockwise-causal attention mask, with full attention within each end-effector's timestep block, causal attention across its blocks, and no cross-effector attention.

branch processes the remaining camera views and the language instruction ℓ, retaining adaptive-resolution image processing. Both visual encoders are frozen, while all other components remain trainable. For tactile-enabled tasks, we use the T3 tactile encoder (Zhao et al., 2024), initialized from FTP-1 pretraining, to extract a 512-dimensional feature from each 3D visuotactile observation. We aggregate these features according to the end-effector configuration. For each two-finger gripper, we concatenate its two sensor features into a 1024-dimensional vector per timestep, then concatenate the vectors at t and $t-1$ to form a 2048-dimensional tactile token. For dexterous hands, we concatenate five fingertip features into a 2560-dimensional vector and apply channel-wise pooling to obtain the 2048-dimensional token. Both configurations yield one tactile token per end-effector, matching the LLM's word embedding dimension and facilitating joint pretraining across grippers and dexterous hands. Bimanual tasks use two tactile input tokens. Tactile features and language instructions condition the temporal perception branch through Cosmos's cross-attention and the reasoning-and-action branch through input tokens in the LLM's embedding space.

Shared attention and action generation. The two branches retain independent QKV projections and FFNs, along with their original pretrained positional encodings. After projection, their query, key, and value features are concatenated along the token dimension for shared attention. At inference, Cosmos performs a single denoising step with full attention to obtain predictive features, whose keys and values are cached for subsequent token generation. As shown in Figure 2, PhysWord then autoregressively generates Physical Words under a blockwise-causal mask. Noisy action tokens follow the Physical Word tokens within the reasoning-and-action branch. They can attend to all preceding multimodal context and are trained using the flow-matching objective $\mathscr{L}_{\text{act}}$. Following Black et al. (2024), a shared MLP maps action tokens to a unified action space with sufficient dimensions to accommodate the supported embodiments.

3.3 Physical Words

A Physical Word is a reasoning primitive that couples an atomic action word label with a contextual hidden state representing future visual or tactile dynamics.

Atomic action vocabulary. Large-scale language pretraining provides LLMs with rich semantic knowledge, while native next-token prediction offers a natural interface for expressing intermediate reasoning (Guo et al., 2026). In robotic manipulation, continuous latents compactly encode multimodal context, but representations of the same action across scenarios may also capture incidental visual variations. Therefore, we introduce Physical Words to organize latent reasoning around

atomic action semantics, preserving shared skill semantics while capturing context-dependent details. Based on prior action taxonomies (Yuan et al., 2026b; Chen et al., 2025b; Nasiriany et al., 2022), we define 16 action labels as semantic anchors for continuous latent representations. Additional action words can be incorporated by initializing their Physical Word variants from the corresponding pretrained LLM embeddings.

Atomic Action Vocabulary

Each label has a visual variant (`_v`) and a tactile variant (`_t`), e.g., `pick_v` and `pick_t`.

Movement: Approach, Move. **Manipulation:** Pick, Place, Flip, Rotate, Toggle, Open, Close, Push, Pull, Insert, Press, Click, Strike, Handover.

From action words to latent representations. Each Physical Word associates an action-word label w with a contextual hidden state $\mathbf{z}$ from which the label is predicted. Let $e(w)$ denote the trainable vocabulary embedding of w and $\mathbf{g}$ the corresponding future sensory latent target used to supervise $\mathbf{z}$. To capture future physical dynamics, we predict both visual dynamics and tactile contact states. Since visual and tactile targets occupy distinct latent spaces and may impose conflicting supervision on a single hidden state, we instantiate separate visual and tactile variants of each action word. Both are initialized from the same pretrained embedding (Bai et al., 2025) while maintaining independent trainable vocabulary entries. This preserves a common semantic initialization while allowing modality-specific hidden states to represent future dynamics. For vision-only tasks, we omit tactile inputs and generate only visual Physical Words, such as `pick_v`. As shown in Figure 2 (b), the word prediction head maps $\mathbf{z}$ to vocabulary logits, supervised by cross-entropy (L_{CE}) against w. The same $\mathbf{z}$ is directly aligned with $\mathbf{g}$ using cosine distance (L_{latent}). This joint supervision couples action semantics with future physical dynamics within a shared contextual representation. PhysWord is trained with blockwise teacher forcing and generates Physical Words autoregressively over future timestep blocks at inference. Compared with latent supervision alone, Physical Words yield lower intra-class distances among latent representations of the same action, and more focused attention on action-relevant regions, as shown in Section 4.2 and Appendix D. Appendix A.3 details our automatic atomic-action annotation procedure.

Visual and tactile latent targets. Given action-word labels paired with future sensory observations, we construct visual latent targets following prior latent VLA methods (Liu et al., 2026). At each target timestep, we encode the future head-view image using Qwen3-VL's visual encoder and pool its final-layer spatial tokens into a single latent target for the corresponding visual Physical Words. All tactile observations are derived from visuotactile signals. We construct future tactile targets separately for each end-effector using the same T3 encoder and feature aggregation as the current tactile inputs. These targets supervise tactile Physical Words to anticipate future contact information. Future observations serve only as training targets and are not used at inference.

3.4 Temporally Structured PhysWord Space

Manipulation unfolds continuously as action semantics and contact conditions evolve. A single future Physical Word may miss intermediate transitions within an action chunk, such as approaching an object before picking it up. We therefore organize Physical Words over multiple future timesteps to provide action generation with temporally structured predictive context.

Temporal organization. To reduce the inference latency of dense Physical Word prediction, we predict Physical Words every s steps over the horizon H, yielding $K = H/s$ future prediction times: $t+s, t+2s, \ldots, t+H$, where t is the current timestep. Let $\mathbf{z}_{t,k}^{r,m}$ denote the contextual output hidden state used to predict a Physical Word for future time $t+ks$, where $k \in \{1,\ldots,K\}$. Here, $r \in \{\mathrm{L},\mathrm{R}\}$ denotes the left or right end-effector, and $m \in \{\mathrm{v},\mathrm{t}\}$ denotes the visual or tactile Physical Word variant. We organize these representations as

$$\mathbf{Z}_t = \left[B_{t,1}^{\mathrm{L}} \quad B_{t,1}^{\mathrm{R}} \quad \cdots \quad B_{t,K}^{\mathrm{L}} \quad B_{t,K}^{\mathrm{R}}\right], \qquad B_{t,k}^{r} = \left(\mathbf{z}_{t,k}^{r,\mathrm{v}}, \mathbf{z}_{t,k}^{r,\mathrm{t}}\right). \tag{1}$$

Blocks are ordered by future timestep, with the left end-effector preceding the right. Each block $B_{t,k}^{r}$ contains the visual and tactile predictions for end-effector r at time $t+ks$.

Temporal blockwise generation. As shown in Figure 2(c), PhysWord generates Physical Words autoregressively over future timestep blocks, jointly predicting visual and tactile variants for each

end-effector. For $k = 1$, we initialize generation with a Physical Word start token appended to the task instruction. We apply a full-block input–target shift: for $k \geq 2$, each block receives word embeddings from $t + (k-1)s$, while its output hidden states receive action-label and sensory-latent supervision for $t + ks$. Attention is full within each block and causal across blocks of the same end-effector, with cross-effector attention blocked. The complete attention mask is provided in Figure 8.

3.5 Training Objectives

We train PhysWord in two stages. In Stage 1, we pretrain the model on approximately 7,000 hours of human-hand and robot demonstrations with and without tactile observations. Details of the pre-training data and configuration are provided in Appendix A. In Stage 2, we fine-tune the PhysWord on task-specific demonstrations for downstream manipulation. For each stage, we jointly optimize four objectives: the Cosmos flow-matching loss L_{video}, Physical Word cross-entropy loss L_{CE}, future latent prediction loss L_{latent}, and action loss L_{act}. The latent alignment loss is defined as:

$$L_{\text{latent}} = 1 - \frac{\mathbf{z}_t^\top \mathbf{g}_t}{\|\mathbf{z}_t\|_2 \|\mathbf{g}_t\|_2 + \varepsilon}, \tag{2}$$

where ε ensures numerical stability. For brevity, end-effector, future-offset, and modality indices are omitted in the loss notation. We average this term over all valid latent predictions to obtain $\mathscr{L}_{\text{latent}}$. The total objective combines all four losses: $L = L_{\text{video}} + L_{\text{CE}} + L_{\text{latent}} + L_{\text{act}}$.

4 Experiments

We compare PhysWord with baselines on two simulation benchmarks (Section 4.1), assess component contributions through ablations (Section 4.2), and evaluate real-world performance and Physical Word reuse across unseen configurations and long-horizon tasks (Section 4.3).

4.1 Simulation Experiments

Benchmarks and Protocol. We evaluate PhysWord on two benchmarks: UniVTAC for contact-rich manipulation with visuotactile sensing and LIBERO-Long for long-horizon manipulation. **UniVTAC** (Chen et al., 2026) comprises eight tasks spanning lifting, classification, and insertion. **LIBERO-Long** (Liu et al., 2023) comprises ten long-horizon tasks. For both benchmarks, we use 50 training demonstrations per task and follow the official success criteria. We evaluate PhysWord over three runs of 100 rollouts per task and compute each task's standard deviation across runs.

Baselines and Implementation details. On UniVTAC, we compare PhysWord with two categories of baselines: (1) *vision-based models*, including $\pi_{0.5}$ (Physical Intelligence et al., 2025), StarVLA-α (Ye et al., 2026a), and InternVLA-A1 (Cai et al., 2026), and (2) *tactile-enabled policies*, including VITaL (George et al., 2025), ACT^{Tac} with the UniVTAC Encoder, FTP-1 (Yuan et al., 2026a), N0-VTLA (Team et al., 2026), and T-Rex (Niu et al., 2026). On LIBERO-Long, we compare PhysWord with two categories of baselines: (1) *VLA models*, including $\pi_{0.5}$, OpenVLA-OFT (Kim et al., 2025), GR00T-N1.7 (Bjorck et al., 2025), and LaST$_0$ (Liu et al., 2026), and (2) *WAMs*, including Cosmos Policy (Kim et al., 2026), Motus (Bi et al., 2025), Fast-WAM (Yuan et al., 2026c), and Cosmos 3 (Agarwal et al., 2026). Across both benchmarks, we report official baseline results or reproduce them through full fine-tuning following each method's official setup. On UniVTAC, we train a separate policy for each task, following FTP-1. On LIBERO-Long, we jointly train a single policy on all tasks. Detailed implementation settings for PhysWord and the baselines are provided in Appendix F.

Results on UniVTAC. PhysWord achieves a mean success rate of 90.3% (average per-task standard deviation (SD): 1.5 percentage points), surpassing the strongest tactile baseline, N0-VTLA, by 7.2 percentage points and the strongest vision-based VLA baseline, InternVLA-A1, by 23.2 percentage points. Gains are particularly pronounced on *Put Bottle in Shelf* and *Lift Can*, suggesting benefits for sustained motion coordination and contact-sensitive manipulation. We further evaluate **PhysWord-feedback**, which replaces measured tactile inputs with predicted tactile latents. It achieves 89% mean success, only 1.3 percentage points below the full model, suggesting that the predicted tactile latents faithfully capture contact information needed for control. These results indicate that atomic action semantics can anchor future dynamics while preserving the fidelity of tactile predictions. Per-

Table 1: Comparison on UniVTAC (left) and LIBERO-Long (right) in terms of success rates (%). Bold indicates the best performance for each task.

UniVTAC (Visuotactile simulator)										LIBERO-Long	
Method	**Lift Bottle**	**Insert Tube**	**Insert Hole**	**Grasp Classify**	**Pull Out Key**	**Put Bottle**	**Insert HDMI**	**Lift Can**	**Mean S.R. ↑**	**Method**	**S.R. ↑**
ACT^{Tac}	71	56	24	99	46	31	28	29	48.0	$\pi_{0.5}$	92.4
VITaL	72	34	25	**100**	47	32	6	8	40.5	OpenVLA-OFT	94.5
$\pi_{0.5}$	**100**	74	25	49	35	34	8	72	49.6	GR00T-N1.7	94.4
StarVLA-α	32	69	52	68	51	88	24	65	56.1	LaST$_0$	95.6
InternVLA-A1	58	95	93	86	66	37	12	90	67.1	Cosmos Policy	97.6
FTP-1	97	79	64	99	48	47	4	65	62.9	Motus	97.6
N0-VTLA	99	99	95	**100**	99	60	25	88	83.1	Fast-WAM	95.2
T-Rex	99	95	**100**	95	99	79	8	75	81.3	Cosmos 3	97.4
PhysWord	**100**	**100**	**100**	**100**	**100**	**93**	**34**	**95**	**90.3**	**PhysWord**	**99.6**

task standard deviations for PhysWord are reported in Appendix C.1, while detailed experiments on predictive tactile feedback in simulation and the real world are detailed in Appendix C.4.

Results on LIBERO-Long. We also evaluate PhysWord on the vision-only benchmark, where it predicts only visual Physical Words. As shown in Table 1, PhysWord achieves a mean success rate of 99.6%, with an average per-task SD of 0.3 percentage points. It outperforms $\pi_{0.5}$ (92.4%), LaST$_0$ (95.6%), and Cosmos 3 (97.4%) by 7.2, 4.0, and 2.2 percentage points, respectively. These gains over VLA, latent reasoning, and WAM baselines suggest that temporally structured Physical Words provide effective predictive context for long-horizon action generation. By organizing future visual dynamics around atomic action semantics, they capture transitions between task stages.

Model inference speed. With BF16 precision, five action denoising steps, and a Physical Word prediction interval of $s = 4$, PhysWord runs at 4.9 Hz on an NVIDIA H100 GPU. An optimized inference runtime increases throughput to 10.6 Hz through kernel fusion, caching of reusable metadata, and parallel preprocessing and encoding. Details of these strategies are provided in Appendix F.3.

4.2 Ablation Studies

We conduct ablation studies on LIBERO-Long and two UniVTAC tasks (*Lift Can* and *Put Bottle*) to examine Physical Word structure and temporal organization. Appendix C.2 presents further ablations on tactile inputs and prediction, visual target selection, and Physical Word embedding design. Appendix C.1 shows additional results for PhysWord without the temporal perception branch.

4.2.1 Effectiveness of Physical Word Latent Reasoning

We compare multimodal structures and supervision objectives to assess their contributions to Physical Word latent reasoning. Figure 3 (left) summarizes three groups of ablations. Our PhysWord achieves the best overall performance on both visuotactile UniVTAC and vision-only LIBERO-Long. All variants with temporal prediction use the best-performing sampling interval of $s = 4$.

Group 1: Individual supervision signals. We first evaluate visual latent supervision alone (O), tactile latent supervision alone (T), and atomic-word prediction alone (W). For the latent-only variants, we follow LaST$_0$ and autoregressively predict latent tokens using the same temporal organization as PhysWord. These variants underperform joint semantic and latent supervision, suggesting that action semantics and future dynamics provide complementary guidance for latent reasoning.

Group 2: Joint semantic and latent supervision. Combining atomic-word prediction with visual (WO) or tactile (WT) latent supervision improves performance over the corresponding single-objective variants in Group 1. These gains suggest that shared action labels serve as semantic anchors for latent learning, helping the model capture common action semantics while retaining context-specific physical details. We further evaluate WO_{MLP} (82% on Put Bottle and 78% on Lift Can) and WT_{MLP} (80% on Put Bottle and 70% on Lift Can), both of which apply a learnable MLP to the contextual hidden states before aligning them with future latent targets. Both variants perform worse than direct hidden-state supervision. These results suggest that directly supervising the word hidden states more effectively couples future physical dynamics with the action semantics.

Group	Variant	UniVTAC Bottle	UniVTAC Can	LIBERO Avg.
1	O	77	83	97.0
	T	73	65	–
	W	77	67	97.4
2	WO	87	89	99.6
	WT	88	75	–
3	$WT \rightarrow WO$	90	90	–
	$WO \rightarrow WT$	93	95	–

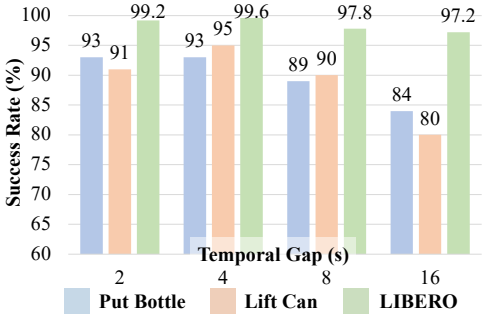


Figure 3: **Ablation study.** Left: Physical Word structure ablations on two UniVTAC tasks and LIBERO-Long. Right: effect of the prediction interval in the temporally structured PhysWord space.

Group 3: Multimodal reasoning structure. With direct latent supervision, we compare two within-block token orders: tactile before visual ($WT \rightarrow WO$) and visual before tactile ($WO \rightarrow WT$). Both orders outperform the unimodal variants, suggesting that visual and tactile Physical Words provide complementary predictive context. We further compare pretrained VLM word embeddings with random initialization for both orderings. On *Put Bottle* and *Lift Can*, random initialization reduces success rates from 90%/90% to 84%/86% for $WT \rightarrow WO$, and from 93%/95% to 89%/90% for $WO \rightarrow WT$, respectively. These results highlight the value of inheriting word embeddings from large-scale VLM pretraining for learning semantically structured physical latent representations.

Representation analysis. To examine the effect of atomic-word supervision beyond policy performance, we analyze intra-class latent distances using 50 examples for each of the four most frequent atomic actions in our downstream real-world data: *pick*, *place*, *approach*, and *move*. We compare latent-only supervision (O, T) with Physical Words (WO, WT) using identical samples within each modality. We compute the mean pairwise cosine distance between contextual hidden states within each action class. Averaged across the four actions, word supervision reduces visual distances from 5.13×10^{-3} (O) to 3.03×10^{-3} (WO) and tactile distances from 2.50×10^{-4} (T) to 1.25×10^{-4} (WT), corresponding to reductions of approximately 41% and 50%, respectively. Meanwhile, Figure 4(b) compares latent-to-image attention maps for *O*/*W*/*WO* and *T*/*WT*. As highlighted by the red boxes, joint word and latent supervision concentrates attention on manipulated objects and task-relevant regions. Together, these findings suggest that semantic anchoring promotes consistent action-related representations across contexts while reducing sensitivity to incidental scene variations. Appendix D provides detailed intra-class distance results and t-SNE visualizations (Figure 6).

4.2.2 Temporal Prediction Interval

We evaluate Physical Word prediction intervals of 2, 4, 8, and 16 with a fixed action horizon of $H = 16$. For example, an interval of 2 yields predictions at $t+2, t+4, \ldots, t+16$. As shown in Figure 3, shorter intervals generally improve performance, suggesting that denser predictions capture intermediate action transitions and changes in physical dynamics that coarser sampling may miss. We select an interval of 4 to balance manipulation success and inference efficiency, preserving informative intermediate predictions while limiting reasoning overhead.

4.3 Real-World Experiments

Real-world experimental setup. We evaluate PhysWord on two robotic platforms equipped with dexterous hands and parallel grippers, respectively. For dexterous manipulation, we use Tianji arms equipped with Sharpa Wave hands featuring tactile sensing on all fingers. As shown in Figure 4 (a), we evaluate contact-rich *Pick Paper* and long-horizon *Shuttlecock Insertion*. For gripper-based manipulation, we use an i7 robot with Xense tactile sensors mounted on its gripper fingertips, evaluating long-horizon *Express Packing* and precision-demanding *Powered Screw Driving*. Additional details of the robotic platforms and data collection are provided in Appendix B.

Baselines and implementation details. We compare PhysWord with $\pi_{0.5}$, a representative VLA model, and FTP-1, a recent tactile foundation policy. All methods receive the same head- and wrist-view RGB observations. FTP-1 and PhysWord additionally receive tactile observations. For each task, we collect 100 training demonstrations and evaluate each method over three runs of 30 trials each. Additional details of the evaluation protocol are provided in Appendix G.

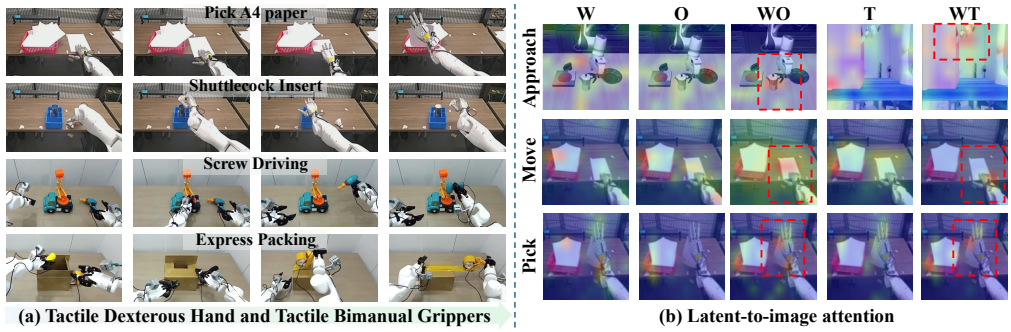


Figure 4: **Visualizations.** (a) PhysWord real-world rollouts. (b) Latent-to-image attention maps.

Table 2: **Real-world evaluation.** Success rates (%) on four real-world manipulation tasks and on Pick Paper and Screw Drive with unseen backgrounds (Bac), objects (Obj), and positions (Pos). Drop denotes the relative change from the in-domain mean for the same two tasks.

(a) Manipulation performance

Method	Pick Paper	Shuttlecock Insert	Express Packing	Screw Drive	Mean ± Avg. SD
$\pi_{0.5}$	53.3	43.3	46.7	40.0	45.8±3.6
FTP-1	66.7	56.7	50.0	46.7	55.0±4.6
PhysWord	93.3	60.0	73.3	60.0	71.7±2.5

(b) Unseen configurations

Method	Pick Paper			Screw Drive			Mean	Drop
Unseen:	Bac	Obj	Pos	Bac	Obj	Pos	± Avg. SD.	
$\pi_{0.5}$	46.7	40.0	13.3	26.7	20.0	26.7	28.9 ± 3.8	-38%
FTP-1	60.0	46.7	26.7	23.3	23.3	30.0	35.0 ± 6.2	-38%
PhysWord	90.0	83.3	80.0	43.3	36.7	46.7	63.3 ± 3.3	-17%

Real-world results. Table 2(a) shows that PhysWord outperforms both baselines on all four tasks, achieving a mean success rate of 71.7%, compared with 45.8% for $\pi_{0.5}$ and 55.0% for FTP-1. Improvements on *Pick Paper* and *Shuttlecock Insertion* demonstrate its effectiveness for dexterous manipulation requiring fine contact control. Gains on *Express Packing* and *Powered Screw Driving* further support its applicability to long-horizon manipulation and contact-sensitive tool use.

Unseen configurations. We directly evaluate all methods on unseen backgrounds, objects, and positions, ensuring that none of the test configurations appear in the training data. As shown in Table 2(b), PhysWord achieves 63.3% mean success, outperforming $\pi_{0.5}$ and FTP-1 by 34.4 and 28.3 percentage points, respectively. We evaluate Physical Word predictions against GPT annotations on 2,000 frames sampled across all rollouts and configurations, covering *approach*, *move*, *pick*, and *place*, and obtain 90.5% frame-level accuracy. Together, these results suggest that Physical Words preserve action semantics across contexts, providing reusable semantic anchors for predicting physical dynamics and supporting robust manipulation.

Long-horizon extension. We further evaluate extended versions of *Pick A4 Paper* and *Express Packing* involving task sequences absent from training, requiring repeated pick-and-place cycles and revisiting packing stages. PhysWord outperforms both baselines by 70.0 percentage points on *Pick A4 Paper*, and exceeds $\pi_{0.5}$ and FTP-1 by 40.0 and 36.7 percentage points, respectively, on *Express Packing*. During repeated paper manipulation, the predicted Physical Word switches from `place` back to `approach` when a new sheet is introduced, providing context for the next cycle. These results suggest that Physical Words support the reuse of learned action primitives as task conditions evolve, enabling execution beyond the sequences observed during training. Appendix C.3.1 provides detailed experimental setups and per-task results, along with additional experiments examining the contribution of Physical Words to executing extended long-horizon sequences.

5 Conclusion and Limitation

In this paper, we presented PhysWord, a reasoning-before-acting foundation model that uses atomic action semantics to structure latent representations of future physical dynamics. Its temporally organized visual and tactile Physical Words capture evolving motion and contact dynamics, providing predictive context for action generation. This temporal structure also supports the reuse of action primitives across task stages. Experiments on simulation benchmarks and real-world platforms demonstrate strong manipulation performance, generalization to unseen configurations, and effective execution of long-horizon tasks. Our current action vocabulary does not cover whole-body control, such as climbing stairs, which we leave for future work.

AI USE STATEMENT

We use GPT (OpenAI, 2026) to segment demonstrations into atomic action stages and annotate action labels separately for the left and right end-effectors, as detailed in Appendix A. We also use GPT to improve the writing of the manuscript. The authors reviewed AI-assisted content, verified references, code, and results, and take responsibility for the final paper.

REFERENCES

Niket Agarwal, Arslan Ali, Jon Allen, Martin Antolini, Adeline Aubame, Alisson Azzolini, Junjie Bai, Maciej Bala, Yogesh Balaji, Josh Bapst, et al. Cosmos 3: Omnimodal world models for physical ai. *arXiv preprint arXiv:2606.02800*, 2026.

Arslan Ali, Junjie Bai, Maciej Bala, Yogesh Balaji, Aaron Blakeman, Tiffany Cai, Jiaxin Cao, Tianshi Cao, Elizabeth Cha, Yu-Wei Chao, et al. World simulation with video foundation models for physical ai. *arXiv preprint arXiv:2511.00062*, 2025.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Suneel Belkhale, Tianli Ding, Ted Xiao, Pierre Sermanet, Quon Vuong, Jonathan Tompson, Yevgen Chebotar, Debidatta Dwibedi, and Dorsa Sadigh. Rt-h: Action hierarchies using language, 2024. URL https://arxiv.org/abs/2403.01823.

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

Hongzhe Bi, Hengkai Tan, Shenghao Xie, Zeyuan Wang, Shuhe Huang, Haitian Liu, Ruowen Zhao, Yao Feng, Chendong Xiang, Yinze Rong, et al. Motus: A unified latent action world model. *arXiv preprint arXiv:2512.13030*, 2025.

Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. pi0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Kevin Black, Manuel Y. Galliker, and Sergey Levine. Real-time execution of action chunking flow policies, 2025. URL https://arxiv.org/abs/2506.07339.

Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Shenyuan Gao, Xindong He, Xuan Hu, Xu Huang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv:2503.06669*, 2025.

Junhao Cai, Zetao Cai, Jiafei Cao, Yilun Chen, Zeyu He, Lei Jiang, Hang Li, Hengjie Li, Yang Li, Yufei Liu, et al. Internvla-a1: Unifying understanding, generation and action for robotic manipulation. *arXiv preprint arXiv:2601.02456*, 2026.

Jun Cen, Chaohui Yu, Hangjie Yuan, Yuming Jiang, Siteng Huang, Jiayan Guo, Xin Li, Yibing Song, Hao Luo, Fan Wang, et al. Worldvla: Towards autoregressive action world model. *arXiv preprint arXiv:2506.21539*, 2025.

Baijun Chen, Weijie Wan, Tianxing Chen, Xianda Guo, Congsheng Xu, Yuanyang Qi, Haojie Zhang, Longyan Wu, Tianling Xu, Zixuan Li, et al. Univtac: A unified simulation platform for visuo-tactile manipulation data generation, learning, and benchmarking. *arXiv preprint arXiv:2602.10093*, 2026.

Jiayi Chen, Wenxuan Song, Pengxiang Ding, Ziyang Zhou, Han Zhao, Feilong Tang, Donglin Wang, and Haoang Li. Unified diffusion vla: Vision-language-action model via joint discrete denoising diffusion process, 2025a. URL https://arxiv.org/abs/2511.01718.

Zeren Chen, Zhelun Shi, Xiaoya Lu, Lehan He, Sucheng Qian, Enshen Zhou, Zhenfei Yin, Wanli Ouyang, Jing Shao, Yu Qiao, et al. Rh20t-p: A primitive-level robotic manipulation dataset towards composable generalization agents in real-world scenarios. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 20532–20539. IEEE, 2025b.

Yuntian Deng, Yejin Choi, and Stuart Shieber. From explicit cot to implicit cot: Learning to internalize cot step by step. *arXiv preprint arXiv:2405.14838*, 2024.

Shenyuan Gao, Siyuan Zhou, Yilun Du, Jun Zhang, and Chuang Gan. Adaworld: Learning adaptable world models with latent actions. *arXiv preprint arXiv:2503.18938*, 2025.

Abraham George, Selam Gano, Pranav Katragadda, and Amir Barati Farimani. Vital pretraining: Visuo-tactile pretraining for tactile and non-tactile manipulation policies. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 258–264. IEEE, 2025.

Chenyang Gu, Jiaming Liu, Hao Chen, Runzhong Huang, Qingpo Wuwu, Zhuoyang Liu, Xiaoqi Li, Ying Li, Renrui Zhang, Peng Jia, et al. Manualvla: A unified vla model for chain-of-thought manual generation and robotic manipulation. *arXiv preprint arXiv:2512.02013*, 2025.

Ziyu Guo, Rain Liu, Xinyan Chen, and Pheng-Ann Heng. Atlas: Agentic or latent visual reasoning? one word is enough for both. *arXiv preprint arXiv:2605.15198*, 2026.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*, 2024.

Yucheng Hu, Yanjiang Guo, Pengchao Wang, Xiaoyu Chen, Yen-Jen Wang, Jianke Zhang, Koushil Sreenath, Chaochao Lu, and Jianyu Chen. Video prediction policy: A generalist robot policy with predictive visual representations. *arXiv preprint arXiv:2412.14803*, 2024.

Chi-Pin Huang, Yueh-Hua Wu, Min-Hung Chen, Yu-Chiang Frank Wang, and Fu-En Yang. Thinkact: Vision-language-action reasoning via reinforced visual latent planning. *arXiv preprint arXiv:2507.16815*, 2025.

Siddharth Karamcheti, Suraj Nair, Ashwin Balakrishna, Percy Liang, Thomas Kollar, and Dorsa Sadigh. Prismatic vlms: Investigating the design space of visually-conditioned language models. In *Forty-first International Conference on Machine Learning*, 2024.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success, 2025. URL https://arxiv.org/abs/2502.19645.

Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, et al. Cosmos policy: Fine-tuning video models for visuomotor control and planning. *arXiv preprint arXiv:2601.16163*, 2026.

Xiaoqi Li, Mingxu Zhang, Yiran Geng, Haoran Geng, Yuxing Long, Yan Shen, Renrui Zhang, Jiaming Liu, and Hao Dong. Manipllm: Embodied multimodal large language model for object-centric robotic manipulation, 2023. URL https://arxiv.org/abs/2312.16217.

Fanqi Lin, Ruiqian Nai, Yingdong Hu, Jiacheng You, Junming Zhao, and Yang Gao. Onetwovla: A unified vision-language-action model with adaptive reasoning. *arXiv preprint arXiv:2505.11917*, 2025.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning, 2023. URL https://arxiv.org/abs/2306.03310.

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

Zhuoyang Liu, Jiaming Liu, Jiadong Xu, Nuowei Han, Chenyang Gu, Hao Chen, Kaichen Zhou, Renrui Zhang, Kai Chin Hsieh, Kun Wu, et al. Mla: A multisensory language-action model for multimodal understanding and forecasting in robotic manipulation. *arXiv preprint arXiv:2509.26642*, 2025b.

Zhuoyang Liu, Jiaming Liu, Hao Chen, Jiale Yu, Ziyu Guo, Chengkai Hou, Chenyang Gu, Xiangju Mi, Renrui Zhang, Kun Wu, Zhengping Che, Jian Tang, Pheng-Ann Heng, and Shanghang Zhang. Last_0: Latent spatio-temporal chain-of-thought for robotic vision-language-action model, 2026. URL https://arxiv.org/abs/2601.05248.

Hao Luo, Wanpeng Zhang, Yicheng Feng, Sipeng Zheng, Haiweng Xu, Chaoyi Xu, Ziheng Xi, Yuhui Fu, and Zongqing Lu. Being-h0. 7: A latent world-action model from egocentric videos. *arXiv preprint arXiv:2605.00078*, 2026.

Soroush Nasiriany, Huihan Liu, and Yuke Zhu. Augmenting reinforcement learning with behavior primitives for diverse manipulation tasks. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 7477–7484. IEEE, 2022.

NeoteAI Team and Fudan TEAI Team. N0-Foundation: Towards the age of tactile intelligence. *Technical Report*, 2026. URL https://huggingface.co/datasets/NeoteAIEmbodied/OpenNeoData.

Dantong Niu, Zhuoyang Liu, Zekai Wang, Boning Shao, Zhao-Heng Yin, Anirudh Pai, Yuvan Sharma, Stefano Saravalle, Ruijie Zheng, Jing Wang, et al. T-rex: Tactile-reactive dexterous manipulation. *arXiv preprint arXiv:2606.17055*, 2026.

OpenAI. GPT-5.6 System Card, 2026. URL https://deploymentsafety.openai.com/gpt-5-6.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization, 2025. URL https://arxiv.org/abs/2504.16054.

Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Perceiver-actor: A multi-task transformer for robotic manipulation. In *Conference on Robot Learning*, pp. 785–799. PMLR, 2023.

Shuhan Tan, Kashyap Chitta, Yuxiao Chen, Ran Tian, Yurong You, Yan Wang, Wenjie Luo, Yulong Cao, Philipp Krahenbuhl, Marco Pavone, et al. Latent chain-of-thought world modeling for end-to-end driving. *arXiv preprint arXiv:2512.10226*, 2025.

NeoteAI Team, Fudan TEAI Team, et al. n_0-vtla: Scaling vision-tactile-language-action model with latent tactile tokens. *arXiv preprint arXiv:2607.23782*, 2026.

Yang Tian, Yuyin Yang, Yiman Xie, Zetao Cai, Xu Shi, Ning Gao, Hangxu Liu, Xuekun Jiang, Zherui Qiu, Feng Yuan, et al. Interndata-a1: Pioneering high-fidelity synthetic data for pre-training generalist policy. *arXiv preprint arXiv:2511.16651*, 2025.

Qixun Wang, Yang Shi, Yifei Wang, Yuanxing Zhang, Pengfei Wan, Kun Gai, Xianghua Ying, and Yisen Wang. Monet: Reasoning in latent visual space beyond images and language. *arXiv preprint arXiv:2511.21395*, 2025a.

Yuqi Wang, Xinghang Li, Wenxuan Wang, Junbo Zhang, Yingyan Li, Yuntao Chen, Xinlong Wang, and Zhaoxiang Zhang. Unified vision-language-action model, 2025b. URL https://arxiv.org/abs/2506.19850.

Yuran Wang, Siqiao Huang, Mingleyang Li, Chenhao Zhang, Jiaqi Liang, Weiyang Jin, Yue Chen, Xuemin Chi, Donghao Zhou, Qize Yu, et al. Openwam: An open, modular exploration towards systematic world-action model pretraining. *arXiv preprint arXiv:2609.07398*, 2026.

Junjie Wen, Yichen Zhu, Jinming Li, Zhibin Tang, Chaomin Shen, and Feifei Feng. Dexvla: Vision-language model with plug-in diffusion expert for general robot control. *arXiv preprint arXiv:2502.05855*, 2025.

Zeyuan Yang, Xueyang Yu, Delin Chen, Maohao Shen, and Chuang Gan. Machine mental imagery: Empower multimodal reasoning with latent visual tokens. *arXiv preprint arXiv:2506.17218*, 2025.

Jinhui Ye, Ning Gao, Senqiao Yang, Jinliang Zheng, Zixuan Wang, Yuxin Chen, Pengguang Chen, Yilun Chen, Shu Liu, and Jiaya Jia. StarVLA-α: Reducing complexity in vision-language-action systems. *arXiv preprint arXiv:2604.11757*, 2026a.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, et al. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026b.

Chengbo Yuan, Zicheng Zhang, Mingjie Zhou, Wendi Chen, Yi Wang, Zhuoyang Liu, Dantong Niu, Shuo Wang, Hui Zhang, Wenkang Zhang, et al. Ftp-1: A generalist foundation tactile policy across tactile sensors for contact-rich manipulation. *arXiv preprint arXiv:2606.13102*, 2026a.

Haoqi Yuan, Zhixuan Liang, Anzhe Chen, Ye Wang, Haoyang Li, Pei Lin, Yiyang Huang, Zixing Lei, Tong Zhang, Jiazhao Zhang, et al. Qwen-robotmanip technical report: Alignment unlocks scale for robotic manipulation foundation models. *arXiv preprint arXiv:2606.17846*, 2026b.

Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-wam: Do world action models need test-time future imagination?, 2026c. URL https://arxiv.org/abs/2603.16666.

Michał Zawalski, William Chen, Karl Pertsch, Oier Mees, Chelsea Finn, and Sergey Levine. Robotic control via embodied chain-of-thought reasoning, 2025. URL https://arxiv.org/abs/2407.08693.

Likui Zhang, Tao Tang, Zhihao Zhan, Xiuwei Chen, Zisheng Chen, Jianhua Han, Jiangtong Zhu, Pei Xu, Hang Xu, Hefeng Wu, et al. Atomicvla: Unlocking the potential of atomic skill learning in robots. *arXiv preprint arXiv:2603.07648*, 2026.

Wenyao Zhang, Hongsi Liu, Zekun Qi, Yunnan Wang, Xinqiang Yu, Jiazhao Zhang, Runpei Dong, Jiawei He, He Wang, Zhizheng Zhang, et al. Dreamvla: a vision-language-action model dreamed with comprehensive world knowledge. *arXiv preprint arXiv:2507.04447*, 2025.

Jialiang Zhao, Yuxiang Ma, Lirui Wang, and Edward H Adelson. Transferable tactile transformers for representation learning across diverse sensors and tasks. *arXiv preprint arXiv:2406.13640*, 2024.

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

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

Ruijie Zheng, Dantong Niu, Yuqi Xie, Jing Wang, Mengda Xu, Yunfan Jiang, Fernando Castañeda, Fengyuan Hu, You Liang Tan, Letian Fu, Trevor Darrell, Furong Huang, Yuke Zhu, Danfei Xu, and Linxi Fan. EgoScale: Scaling dexterous manipulation with diverse egocentric human data, 2026. URL https://arxiv.org/abs/2602.16710.

- Section A: Pretraining Data and Details
- Section B: Real-World Experimental Setup
- Section C: Additional Quantitative Results
- Section D: Additional Representation Analysis
- Section E: Failure Case Analysis
- Section F: Additional Implementation Details
- Section G: Real-World Evaluation Protocol
- Section H: Ethics Statement
- Section I: Reproducibility Statement

A Pretraining Data and Details

A.1 Data Sources and Composition

We construct the pretraining corpus from 2,000 hours of human demonstrations collected in our lab, the full public releases of DROID (Khazatsky et al., 2024) and OpenNeoData (NeoteAI Team & Fudan TEAI Team, 2026), and real-world Sharpa demonstrations from T-Rex (Niu et al., 2026) and our own collection. These sources cover human-hand manipulation and robot manipulation with and without tactile observations across different end-effectors. Table 3 summarizes the data composition. Each training example pairs observation context with action targets, atomic action labels, and future sensory targets according to the available modalities. Future observations serve only as training targets and are not provided as policy inputs at inference. Note that our pretraining focuses on establishing Physical Word reasoning capabilities. We deliberately use a smaller manipulation corpus than those used by our main baselines, e.g., $\pi_{0.5}$, N0-VTLA, and T-Rex, to evaluate our approach without an advantage in pretraining data volume.

Table 3: Pretraining data composition and volume.

Dataset	**Duration (h)**	**Tactile**
Human hand data	2,000+	No
DROID	350	No
OpenNeoData	5,000+	Yes
Sharpa data	100+	Yes
Total	7,400+	–

A.2 Data Preprocessing

Visual and action quality filtering. Visual filtering identifies black or near-uniform frames, severe underexposure or overexposure, motion blur, and prolonged periods when the manipulating hand or gripper and relevant objects are not visible in any available view. For action quality, we discard records with invalid values or malformed entries and detect abnormal translation or rotation jumps, unusually short or long trajectories, frozen motion, and excessive proportions of stationary steps. Filtering thresholds are calibrated using training data only. We group trajectories by embodiment, action representation, and control frequency, use the 1st and 99th percentiles (Q_{01} and Q_{99}) as initial references, and manually adjust the thresholds for each filtering criterion.

Trajectory diversity filtering. To reduce redundant demonstrations, we compare trajectories within each dataset that share the same task and scene. We extract keyframes following the keyframe-discovery procedure of PerAct (Shridhar et al., 2023) and compare wrist positions in the world coordinate frame at corresponding keyframes. Trajectories whose corresponding wrist positions differ by less than 2 cm are treated as near-duplicates. We retain at most two demonstrations per group and discard the remainder to reduce motion redundancy that may limit spatial generalization.

Action representations and retargeting. Robot demonstrations retain their original action spaces. For human-hand data, we follow EgoScale's processing protocol (Zheng et al., 2026), retaining wrist

poses and retargeting finger motions to a 22-DoF dexterous hand. This retains the demonstrated wrist motion while providing hand-action targets compatible with the target hand kinematics.

A.3 Atomic Action Annotation

Annotation method. We use GPT (OpenAI, 2026) to segment processed demonstrations into atomic action stages and assign labels from our action vocabulary. For bimanual demonstrations, the left and right end-effectors are annotated independently, allowing simultaneous actions to receive distinct labels. We record each stage's temporal boundaries and align its label with the corresponding observations, providing cross-entropy targets paired with future sensory supervision. Visual and tactile Physical Words use modality-specific variants of the same action label, with tactile variants and their latent supervision included only when tactile observations are available.

Annotation quality assessment. We randomly sample 500 examples from the pretraining corpus and manually annotate their atomic actions. GPT-generated labels achieve 95% accuracy against these human annotations. Errors primarily involve ambiguous or incidental movements. For example, finger tapping on a table during pauses in manipulation may be mislabeled as *click*, despite the absence of a button press. These cases highlight the difficulty of distinguishing intentional manipulation from incidental motion. Nevertheless, VLM-based video annotation provides a practical, scalable pipeline for fine-grained atomic-action labeling. PhysWord maintains stable task execution despite these occasional ambiguities, suggesting robustness to limited annotation noise.

A.4 Pretraining Configuration

We pretrain PhysWord for two epochs using the same optimization configuration as downstream fine-tuning (Appendix F). We use AdamW with zero weight decay and a per-GPU batch size of 6. The base learning rate is 10^{-4}, with separate rates of 2×10^{-6} for the Cosmos backbone and 5×10^{-5} for the T3 tactile encoder. Training uses BF16 mixed precision and linear warmup over the first 5% of optimizer steps, followed by cosine learning-rate decay to zero.

B Real-World Experimental Setup

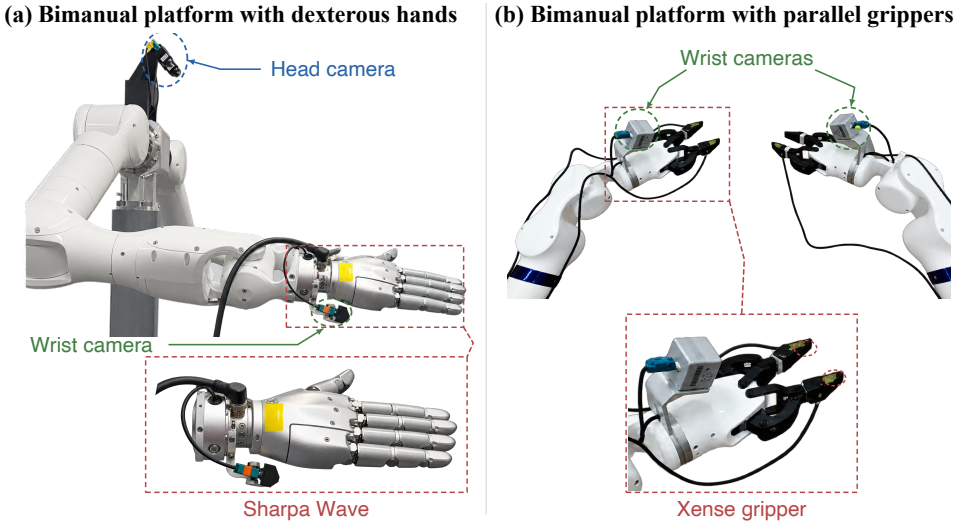


Figure 5: **Real-world robotic platforms and sensing.** Representative hardware views of (a) the bimanual platform with Sharpa Wave dexterous hands and (b) the bimanual platform with Xense tactile grippers. Blue and green annotations identify head and wrist cameras, respectively; red insets highlight the end effectors.

Table 4: Arm joint position limits in radians, excluding end-effector joints.

Joint	i7 Pro left	i7 Pro right	Tianji
J1	$[-3.14, 3.14]$	$[-3.14, 3.14]$	$[-3.1067, 3.1067]$
J2	$[-0.6977, 3.8377]$	$[-3.8377, 0.6977]$	$[-2.0944, 2.0944]$
J3	$[-3.14, 3.14]$	$[-3.14, 3.14]$	$[-3.1067, 3.1067]$
J4	$[-2.4248, 2.4248]$	$[-2.4248, 2.4248]$	$[-2.5307, 1.0472]$
J5	$[-3.14, 3.14]$	$[-3.14, 3.14]$	$[-3.1067, 3.1067]$
J6	$[-1.0467, 1.0467]$	$[-1.0467, 1.0467]$	$[-1.0472, 1.0472]$
J7	$[-1.6572, 1.8317]$	$[-1.6572, 1.8317]$	$[-1.5708, 1.5708]$

B.1 Robotic Platforms and Sensing

Figure 5 presents the two bimanual platforms used in our experiments. The dexterous-hand platform comprises Tianji Marvin arms equipped with two Sharpa Wave hands, each featuring 22 active degrees of freedom and five-finger tactile sensing. The gripper platform uses an i7 Pro robot with Xense tactile sensors mounted on its parallel-gripper fingertips. Each arm has seven revolute joints, denoted J1–J7. Table 4 reports their joint angle limits, excluding end-effector joints. Together, these platforms support evaluation across gripper-based and high-dimensional dexterous manipulation. Both platforms provide three RGB views: one head view capturing the overall scene and two wrist views providing close-range observations of manipulation. Visuotactile observations are acquired from the gripper fingertips or dexterous-hand fingers and supplied to the policy alongside RGB observations and language instructions.

B.2 Downstream Task Data Collection

We collect demonstrations through human teleoperation on both platforms. The bimanual gripper platform uses a PICO-based VR teleoperation interface. The dexterous-hand platform uses the same interface, augmented with Manus gloves to capture finger motions for dexterous-hand control. During teleoperation, we record multi-view RGB observations, tactile signals, robot proprioception, and action commands. We collect 100 demonstrations per task for downstream fine-tuning.

C Additional Quantitative Results

We provide additional simulation results, controlled ablations, and real-world evaluations to further examine the contributions of Physical Words, temporal composition, and predictive tactile feedback.

C.1 Additional Simulation Experiments

Table 5 reports per-task results on all eight UniVTAC tasks, extending the main-paper comparison with PhysWord-Lite.

PhysWord-Lite. To examine whether Physical Words remain effective without the temporal perception branch, we remove Cosmos and supply all camera views to the reasoning-and-action branch. Physical Word generation and its semantic and latent supervision are retained. PhysWord-Lite maintains strong performance on contact-rich manipulation tasks, indicating that effective contact-sensitive control does not depend solely on the temporal perception branch. These results highlight the contribution of the physical latents encoded by Physical Words, which provide action generation with semantically structured predictions of future visual dynamics and contact states. The temporal branch further strengthens perception, while Physical Words retain their effectiveness even without this additional support.

C.2 Additional Ablation Studies

We conduct three ablations on *Put Bottle in Shelf* and *Lift Can*. Unless explicitly varied, we retain the full model's training data, optimization settings, action objective, and evaluation protocol, with a Physical Word prediction interval of $s = 4$.

Table 5: Additional results on UniVTAC, reporting task success rates (%). PhysWord-Lite removes the temporal perception branch. PhysWord-feedback replaces online tactile measurements with predicted tactile latents during evaluation. The SD row is the average of the per-task SDs.

Method	Lift Bottle	Insert Tube	Insert Hole	Grasp Classify	Pull Out Key	Put Bottle	Insert HDMI	Lift Can	Mean S.R. ↑
ACT^{Tac}	71	56	24	99	46	31	28	29	48.0
VITaL	72	34	25	100	47	32	6	8	40.5
$\pi_{0.5}$	100	74	25	49	35	34	8	72	49.6
StarVLA-α	32	69	52	68	51	88	24	65	56.1
InternVLA-A1	58	95	93	86	66	37	12	90	67.1
FTP-1	97	79	64	99	48	47	4	65	62.9
N0-VTLA	99	99	95	100	99	60	25	88	83.1
T-Rex	99	95	100	95	99	79	8	75	81.3
PhysWord	100	100	100	100	100	93	34	95	90.3
(SD)	±0	±0.5	±0	±0	±0.5	±2	±7	±2	±1.5
PhysWord-Lite	99	99	100	98	100	83	29	83	86.4
PhysWord-feedback	99	99	100	98	93	92	37	94	89

Table 6: Ablation of tactile input and prediction, reporting task success rates (%).

Variant	Put Bottle	Lift Can
Full PhysWord	93	95
w/o tactile input	88	87
w/o tactile prediction	87	89
w/o both	82	81

Current tactile input and future tactile prediction. We distinguish the contribution of measured contact context from that of predictive tactile reasoning. The *w/o tactile input* variant removes measured tactile features from both branches while retaining tactile Physical Words and their supervision. The *w/o tactile prediction* variant removes tactile Physical Words and their associated supervision while retaining measured tactile inputs. The *w/o both* variant removes both components, yielding a vision-only configuration. As shown in Table 6, all three variants underperform the full PhysWord model. Current tactile observations inform the policy about the existing contact state, whereas tactile Physical Words anticipate how contact will evolve during execution.

View selection for visual latent targets. We compare visual latent supervision from head-view images, wrist-view images, and both views, while keeping the policy's observation inputs unchanged. When using both views, we generate an additional visual Physical Word at each sampled future timestep, with the two visual tokens supervised by their respective view-specific targets. Table 7 shows that head-view targets yield the best performance. The head view captures object motion and spatial relations across the workspace, providing broader context for predicting future dynamics. Wrist views provide local detail but may become heavily occluded during contact with objects, limiting their predictive information. These results support our use of head-view features as visual latent targets.

Shared versus modality-specific Physical Word embeddings. We compare repeating the same action token twice, e.g., `pick`, `pick`, with using separate modality-specific tokens, e.g., `pick_v`, `pick_t`. Both designs produce two contextual hidden states, supervised by visual and tactile latent targets, respectively. The repeated-token design shares one vocabulary embedding, whereas the modality-specific design uses independently trainable embeddings initialized from the same pretrained action word. Modality-specific tokens achieve better performance (Table 8). Although both modalities describe the same action, their targets occupy distinct representational spaces. This result suggests that separate vocabulary embeddings facilitate modality-specific adaptation while preserv-

Table 7: Effect of visual latent target views on success rates (%), with policy inputs unchanged.

Target view	Put Bottle	Lift Can
Head view	93	95
Wrist views	85	83
Head and wrist views	84	89

Table 8: Shared versus modality-specific Physical Words. Entries report task success rates (%).

Physical Word design	Put Bottle	Lift Can
Shared: `pick`, `pick`	90	89
Separate: `pick_v`, `pick_t`	93	95

ing a common semantic initialization, helping the corresponding hidden states capture complementary motion and contact information.

C.3 Additional Real-World Experiments

C.3.1 Long-Horizon Execution beyond Training Trajectories

We investigate long-horizon generalization on extended versions of two real-world tasks, *Pick A4 Paper* and *Express Packing*. We also compare Physical Words with latent-only reasoning to assess how semantic anchoring supports the reuse of learned behaviors in task sequences absent from training.

Experimental setup. We evaluate long-horizon generalization using extended task sequences absent from the training demonstrations, testing whether learned physical words can be reused as task conditions evolve. For *Pick A4 Paper*, training demonstrations contain a single pick-and-place cycle. During evaluation, a new sheet is introduced after each completed placement, requiring three consecutive cycles within one episode. For *Express Packing*, training demonstrations involve packing one object. At test time, the robot must pack two objects and repeat the closure stage after the box flaps are manually reopened, before sealing the box. Success requires completing the entire extended sequence within a single episode, with every required stage satisfying the task-specific completion criteria in Appendix G. To complement the ablations in Section 4.2.1, we further examine the contribution of semantic anchoring on these extended tasks. For PhysWord, *Extended* uses both visual and tactile Physical Words. The *w/o Physical Words* variant retains visual and tactile latent supervision but removes atomic-word supervision, directly feeding predicted latents back as subsequent reasoning inputs instead of word embeddings.

Results. As shown in the *Extended* columns of Table 9, both baselines fail to complete extended *Pick A4 Paper*, whereas PhysWord achieves 70.0% success, demonstrating its ability to extend a learned single-cycle behavior into repeated execution. On *Express Packing*, whose training demonstrations already contain multiple stages, all methods exhibit some ability to handle the extension. PhysWord nevertheless achieves the highest success rate of 66.7%, compared with 26.7% for $\pi_{0.5}$ and 30.0% for FTP-1. Replacing Physical Words with latent-only reasoning reduces success to 36.7% on extended *Pick A4 Paper* and 50.0% on extended *Express Packing*, corresponding to decreases of 33.3 and 16.7 percentage points. The decline is particularly pronounced on *Pick A4 Paper*, where training demonstrations contain only a single pick-and-place cycle. Qualitative observations reveal that many failures occur at newly introduced stage transitions. After placing a sheet in the basket, PhysWord switches from `place` to `approach` when another sheet is introduced, providing explicit semantic context for the next cycle. The latent-only variant may instead remain near the previous placement location, exhibiting vertical oscillations without initiating the next reach. In these cases, another cycle is triggered only when the new sheet is placed at a specific location, suggesting greater dependence on spatial cues to resume execution. These results suggest that semantic anchoring provides action-relevant guidance at stage transitions, supporting both repeated execution of single-cycle behaviors and adaptation of multi-stage behaviors to sequences absent from training.

Table 9: Success rates (%) on the original tasks and their extended long-horizon variants. For PhysWord, the Original and Extended columns both report results for the full model.

Method	Pick A4 Paper			Express Packing		
	Original	Extended	Extended (w/o PW)	Original	Extended	Extended (w/o PW)
$\pi_{0.5}$	53.3	0.0	–	46.7	26.7	–
FTP-1	66.7	0.0	–	50.0	30.0	–
PhysWord	93.3	70.0	36.7	73.3	66.7	50.0

Table 10: Predictive tactile feedback on real-world dexterous tasks. Entries report success rates (%). The two PhysWord variants use the same trained checkpoint.

Method	Online tactile sensing	Pick A4 Paper	Shuttlecock Insertion
$\pi_{0.5}$	No	53.3	43.3
FTP-1	Yes	66.7	56.7
PhysWord	Yes	93.3	60.0
PhysWord w/ tactile feedback	No	90.0	56.7

C.4 Predictive Tactile Feedback

Simulation evaluation. We evaluate whether a policy trained with tactile observations can operate without online tactile sensing. Using the same trained PhysWord checkpoint, we replace measured tactile inputs with previously predicted tactile latents. For initial timesteps not yet covered by predictions, we initialize tactile context using observations from the opening frames of training demonstrations for the corresponding task. In our simulation setup, only the first action of each predicted chunk is executed before replanning, yielding overlapping predictions for future timesteps. We apply temporal ensembling (Zhao et al., 2023) to aggregate tactile latent estimates associated with the same target timestep and feed the resulting estimate back as tactile context when that timestep is reached. As shown in Table 5, PhysWord-feedback achieves performance comparable to PhysWord with measured tactile inputs. These results suggest that semantically anchored tactile latents retain the contact information needed for control, supporting deployment when tactile sensors are unavailable or fail.

Real-world evaluation. We evaluate predictive tactile feedback on *Pick A4 Paper* and *Shuttlecock Insertion*, both of which require precise contact control. Paper manipulation relies on sustained frictional contact before grasping, whereas shuttlecock insertion requires a stable grasp and controlled contact during insertion and pressing. Using the same trained checkpoint, we replace measured tactile inputs with predicted latents, following the initialization and temporal-ensembling procedure used in simulation. On real-world platforms, we execute the first eight actions of each predicted chunk, consistent with the main real-world experiments. As shown in Table 10, PhysWord achieves success rates of 90.0% and 56.7% on Pick A4 Paper and Shuttlecock Insertion without online tactile sensing, compared with 93.3% and 60.0% using measured tactile inputs. The modest decrease of 3.3 percentage points on each task suggests that predicted latents preserve contact information needed for dexterous control. Together with the simulation results, these findings support deploying tactile-trained policies through predictive feedback when tactile measurements are unavailable. We further quantify prediction accuracy by comparing all stored tactile latent predictions with their corresponding future ground-truth latents. These targets are extracted after each rollout from recorded tactile observations and matched to predictions by target timestep. The tactile recordings are used only for offline evaluation and are not supplied to the policy during these feedback rollouts. Across all predictions, the mean cosine similarity is 0.9873 and the mean squared error is 0.0630.

D Additional Representation Analysis

We further examine whether Physical Words learn more semantically structured representations than latent supervision alone. Complementing the attention visualizations and average intra-class

Table 11: Per-action intra-class distances (↓) for visual and tactile representations. Physical Words yield lower distances than latent-only supervision across all evaluated actions.

Action	Visual		Tactile	
	Latent only (O)	Physical Words (WO)	Latent only (T)	Physical Words (WT)
Pick	0.0049	**0.0030**	0.0003	**0.0001**
Place	0.0050	**0.0024**	0.0002	**0.0001**
Approach	0.0044	**0.0026**	0.0003	**0.0002**
Move	0.0062	**0.0041**	0.0002	**0.0001**

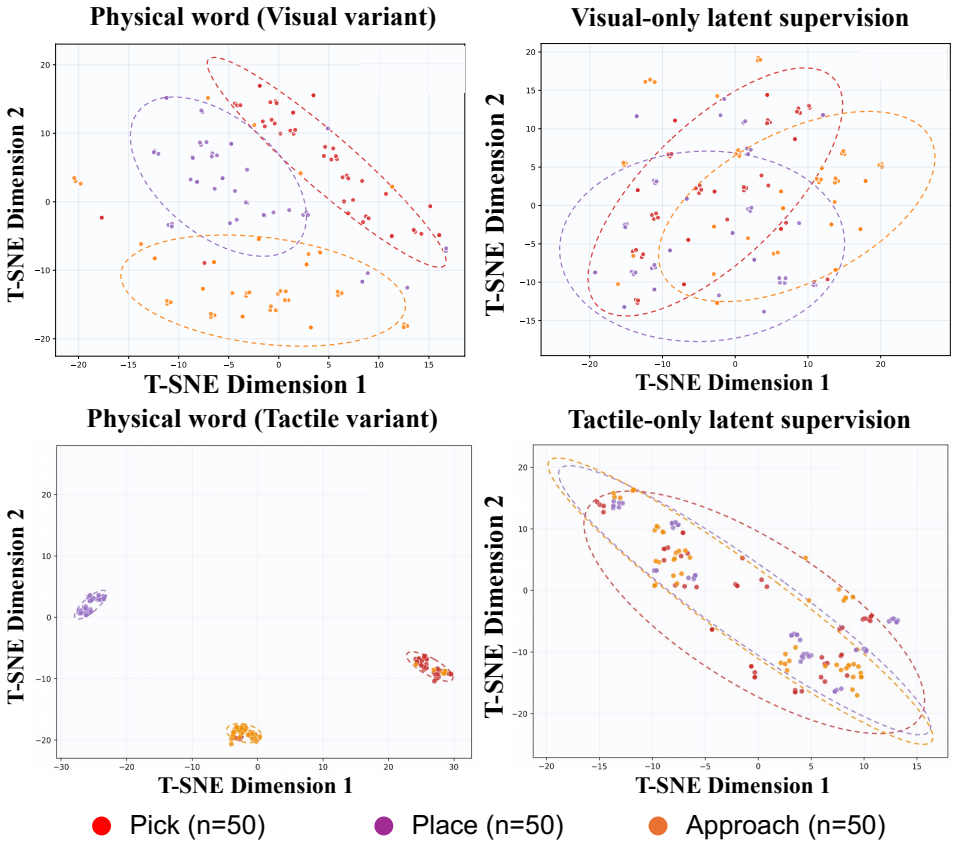


Figure 6: **t-SNE visualization of contextual hidden states.** We compare Physical Words with latent-only supervision for visual (top) and tactile (bottom) representations. Colors denote atomic actions: *pick*, *place*, and *approach*, with 50 samples per action.

distances in the main paper, we report per-action intra-class distances, comparing Physical Word representations with those learned through latent supervision alone. We also use t-SNE to visualize whether representations of the same action cluster across different scenes.

D.1 Detailed Intra-Class Distance Analysis

Experimental setup. We evaluate *pick*, *place*, *approach*, and *move*, the four most frequent atomic actions in our downstream real-world data. For each action, we randomly sample 50 examples for both visual and tactile analyses. We compare visual Physical Words (WO) with visual-only latent supervision (O), and tactile Physical Words (WT) with tactile-only latent supervision (T). Each comparison uses identical samples. We extract the corresponding contextual hidden states and compute the mean pairwise cosine distance within each action class to measure representational consistency across scenarios.

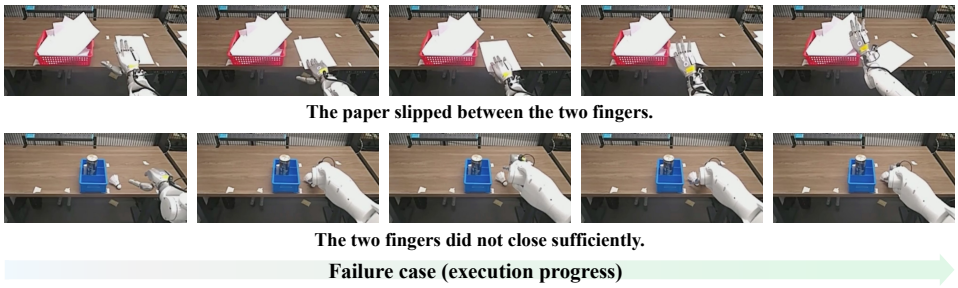


Figure 7: **Failure case.** Representative execution failures during contact-rich manipulation.

Results and analysis. Table 11 shows that atomic-word supervision reduces intra-class distances for all four actions in both modalities. Averaged across actions, visual and tactile distances decrease by approximately 41% and 50%, respectively. These consistent reductions suggest that shared action labels encourage representations of the same action to remain semantically coherent across varying contexts. Together with future-latent supervision, semantic anchoring helps organize context-dependent physical information around reusable action semantics.

D.2 t-SNE Visualization of Physical Word Representations

We compare Physical Words with latent-only supervision through separate t-SNE analyses of visual (WO versus O) and tactile (WT versus T) representations. Using samples from Section D.1, we project each variant's contextual hidden states into two dimensions and color the points by atomic action. As shown in Figure 6, Physical Words form more compact clusters with clearer separation between atomic action categories than latent-only supervision in both modalities. Consistent with the intra-class distance results, these patterns suggest that semantic anchoring promotes shared action semantics across scenes. Moreover, the tactile feedback results in Section C.4 show that predicted tactile latents can replace sensor-derived inputs with only modest performance degradation. Together, these findings suggest that Physical Words improve semantic organization while retaining the contact information needed for tactile-conditioned control.

E Failure Case Analysis

Figure 7 illustrates two failures in dexterous manipulation: paper slippage and insufficient finger closure.

Paper slippage. Because the paper is thin, grasping too close to its edge provides limited contact area, allowing it to slip between the fingers during lifting. Future data collection could emphasize more stable grasp locations farther from the paper edge.

Insufficient finger closure. During shuttlecock grasping, the thumb contacts the object before the index finger achieves sufficient closure. The resulting tactile feedback appears to trigger lifting prematurely, before a stable grasp is established. Despite the initial failure, PhysWord detects that the shuttlecock remains unlifted, predicts the Physical Word `approach` again, and initiates another grasping attempt. This behavior demonstrates its ability to reuse action primitives for recovery based on execution feedback.

F Additional Implementation Details

F.1 Baseline Implementation Details

For UniVTAC, we use baseline results reported in FTP-1 (Yuan et al., 2026a) and N0-VTLA (Team et al., 2026) where available and reproduce missing results using official implementations and pre-trained checkpoints. All reproduced baselines use the same official training split as PhysWord, with 50 demonstrations per task. We match their image input resolutions to those of PhysWord, as well

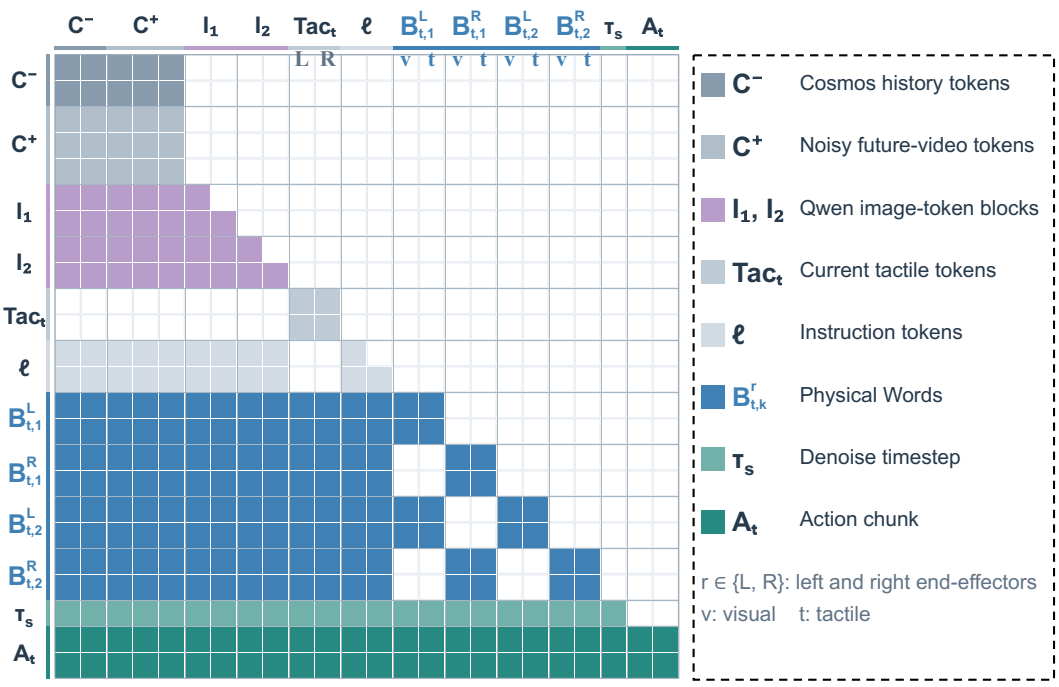


Figure 8: Attention mask for the reasoning-and-action branch. Rows denote queries and columns denote keys. Colored cells indicate allowed attention, while white cells indicate masked connections.

as tactile input resolutions for tactile-enabled methods, while retaining each method's official fine-tuning configuration. Since FTP-1 uses task-dependent camera configurations on UniVTAC, we use both head and wrist views when reproducing its results on previously unreported tasks. For action execution, we follow FTP-1 by executing only the first action of each predicted chunk before replanning.

For real-world evaluation, we compare against $\pi_{0.5}$ and FTP-1. $\pi_{0.5}$ is a PaliGemma-based (Beyer et al., 2024) VLA model that uses flow matching for action generation. We initialize from its official pretrained checkpoint and fine-tune the entire model on downstream demonstrations. Following the official implementation, all RGB images are resized to 224×224. We retain this resolution to preserve compatibility with the pretrained visual encoder. FTP-1 extends the VLA model with sensor-specific tactile encoders and a shared tactile expert that conditions the action expert. We initialize from its released pretrained checkpoint and follow its downstream fine-tuning protocol. Tactile inputs are resized to 224×224 and processed by a sensor-specific Transformer followed by a shared T3 Transformer module. We retain its normalization scheme for proprioceptive states and actions, as well as its optimizer and associated hyperparameters. For a controlled comparison, all methods use the same task demonstrations, camera views, and total number of training iterations. We evaluate the final checkpoint of each method.

F.2 PhysWord Implementation Details

Training configuration. For supervised fine-tuning, we train PhysWord for 100 epochs using AdamW with zero weight decay and a global batch size of 48 on NVIDIA H100 GPUs. The base learning rate is 10^{-4}, with reduced rates of 2×10^{-6} for the Cosmos backbone and 5×10^{-5} for the T3 tactile encoder. We apply linear warmup over the first 5% of optimizer steps, followed by cosine decay to zero, using BF16 mixed precision. To stabilize tactile latent alignment, we apply stop-gradient to the future tactile targets, blocking gradients from the tactile cosine distance loss to T3 through these targets. T3 remains trainable through the current tactile input pathway, while FTP-1 initialization and the reduced learning rate help preserve its pretrained representations and limit target drift. Each action dimension is normalized using the task-specific training-set 1st and 99th percentiles and clipped to $[-1, 1]$. Following FTP-1, tactile RGB inputs are resized to 224×224.

Input processing and attention. Head-view images supplied to the temporal perception branch are resized to 256×256, while the remaining views use Qwen3-VL's native adaptive-resolution processing. Attention patterns for image and text inputs follow the respective pretrained configurations of Cosmos and Qwen3-VL. Figure 8 supplements the main-text description with the complete attention mask, including the preceding observation context and action tokens. Token counts in the schematic are reduced for readability.

Inference and real-time chunking. For real-world deployment, we adopt real-time chunking (RTC) (Black et al., 2025) to overlap policy inference with action execution. Each predicted chunk contains $H = 16$ actions, eight of which are executed before switching to the next chunk. During execution, the policy asynchronously predicts the next chunk using newly acquired observations, allowing the robot to continue executing the current action sequence without pausing for inference.

F.3 Inference Efficiency and Runtime Optimization

All measurements follow the inference configuration in Section 4.1 and run on a single NVIDIA H100 GPU with batch size one. Starting from a baseline implementation running at 4.9 Hz, we apply the following runtime optimizations. None of them modifies model weights, attention masks, or the inference configuration, and all observation-dependent features are recomputed for every request. Table 12 reports the cumulative latency improvements in the order in which the optimizations are applied.

Full CUDA Graph capture. The baseline captures only the Physical Word and action-denoising steps as CUDA graphs. We extend graph capture to all major forward passes, including the RGB and tactile encoders, the history VAE, the Cosmos denoising step, and the context prefill. For each request, new inputs are copied into static buffers before graph replay, reducing Python dispatch and kernel-launch overhead. As shown in Row 1 of Table 12, full CUDA Graph capture reduces median latency from 205.4 to 163.2 ms, a reduction of 42.2 ms.

Mask-preserving FlashAttention. Several attention operations in the reasoning-and-action branch originally relied on explicit dense masks, which precluded efficient attention kernels. We verify the structure of each mask and dispatch it to an equivalent FlashAttention path. Causal regions over the context and Physical Words use bottom-right-aligned causal attention. Tactile queries attend bidirectionally to all keys within the tactile-token block. The timestep and action queries of each denoising step are computed separately according to their respective visibility. Masks that do not match a verified pattern fall back to the original implementation, preserving the causal and tactile-isolation constraints. Row 2 shows that mask-preserving FlashAttention further reduces median latency from 163.2 to 145.5 ms, saving 17.7 ms.

GPU RGB/tactile preprocessing. Raw RGB images are transferred asynchronously through pinned memory. A Triton kernel then performs normalization and Qwen3-VL patch packing directly on the GPU, producing inputs that match those of the original processor. This reduces median latency by 13.8 ms (Row 3). Furthermore, tactile images from all sensors and history frames are resized and normalized in a single batched GPU operation, which is also captured as a CUDA graph. This provides an additional 3.9 ms reduction when introduced later in the optimization sequence (Row 8). Together, these two steps account for 17.7 ms of the total latency reduction.

Fused RMSNorm, packed projections, and fused RoPE. We implement custom Triton kernels for RMSNorm and rotary position embedding (RoPE), fusing their elementwise and reduction operations to reduce intermediate tensors and memory traffic. The fused RMSNorm replaces the RMSNorm layers in both the Qwen and Cosmos backbones, retaining FP32 reduction and the original casting order while reducing median latency by 15.2 ms (Row 4) and 2.9 ms (Row 6), respectively. In the reasoning-and-action branch, the query, key, and value projections, as well as the gate and up projections of each MLP, are packed into single GEMMs; together with fused RoPE, this reduces median latency by another 8.2 ms (Row 5). These three steps collectively contribute a latency reduction of 26.3 ms.

Session/Positional Metadata caching. For a fixed instruction and input shape, instruction tokens, attention masks, visual grid information, and positional and rotary metadata remain unchanged across requests. We cache these after the first call and reuse per-layer KV workspaces within each request, while RGB, tactile, and head-view history inputs are updated at every step. Specifically,

Table 12: End-to-end inference latency on a single NVIDIA H100 GPU as runtime optimizations are applied cumulatively. Frequency is computed from median latency.

Row	Implementation	Median (ms)	P95 (ms)	Freq. (Hz)
	Baseline	205.4	216.5	4.9
1	+ Full CUDA Graph capture	163.2	164.0	6.1
2	+ Mask-preserving FlashAttention	145.5	146.3	6.9
3	+ GPU RGB preprocessing	131.7	132.0	7.6
4	+ Fused RMSNorm (Qwen)	116.5	119.0	8.6
5	+ Packed projections and fused RoPE	108.3	108.7	9.2
6	+ Fused RMSNorm (Cosmos)	105.4	105.8	9.5
7	+ Session metadata caching	101.7	102.7	9.8
8	+ GPU tactile preprocessing	97.8	99.4	10.2
9	+ Positional metadata caching	96.3	97.9	10.4
10	+ VAE/RGB stream overlap	**94.6**	**95.1**	**10.6**

session metadata caching and positional metadata caching reduce median latency by 3.7 ms (Row 7) and 1.5 ms (Row 9), respectively, contributing a combined reduction of 5.2 ms.

VAE/RGB stream overlap. Encoding the head-view history with the video VAE is independent of encoding the current frames. We therefore run the VAE on an auxiliary CUDA stream concurrently with the RGB and tactile encoders. A stream dependency ensures that the Cosmos step starts only after the history latents are ready. This reduces median latency from 96.3 to 94.6 ms (Row 10), saving another 1.7 ms. With all optimizations enabled, inference runs at 10.6 Hz, corresponding to a 2.2$\times$ speedup over the baseline.

G Real-World Evaluation Protocol

G.1 Task Definitions and Success Criteria

We mark 30 distinct positions on the table and evaluate each task with the manipulated object's center placed at each position in turn. Target receptacles or fixtures, including the basket, shuttlecock tube, express box, and toy-car base, are also randomly initialized within a task-specific region. All methods are evaluated under matched initial configurations, with the environment reset after each trial. Success is assessed manually using the task-specific criteria below. Representative task executions are shown in Figure 4 of the main paper and the supplementary videos.

Pick A4 Paper. An A4 sheet is initially placed entirely on the tabletop, with no edge overhanging. The dexterous hand first establishes contact with its five fingers and uses friction to gradually slide the sheet toward the table edge. Once part of the sheet extends beyond the edge, the hand grasps and lifts it, then transfers it into a basket. A trial succeeds only if the entire sheet is placed inside the basket.

Shuttlecock Insertion. One to three shuttlecocks are randomly placed on the table. The robot must grasp each shuttlecock by its feathers without damaging it, insert it into a shuttlecock tube, and press it down with the palm. Success requires all shuttlecocks to be fully inserted and remain intact. Partial insertion does not count as success.

Express Packing. The robot places an object from the tabletop into an express box and closes all four flaps. It then picks up tape with the right gripper and coordinates both grippers to apply the tape along the lid seam and tear it off. Success requires completion of the entire sequence and a fully sealed box. Tape that fails to secure the lid seam is counted as a failure.

Powered Screw Driving. The left gripper grasps a screw and positions it in the designated hole on a toy-car base, while the right gripper operates a powered screwdriver to drive it into place. A trial succeeds only when the screw is fully driven into the designated hole.

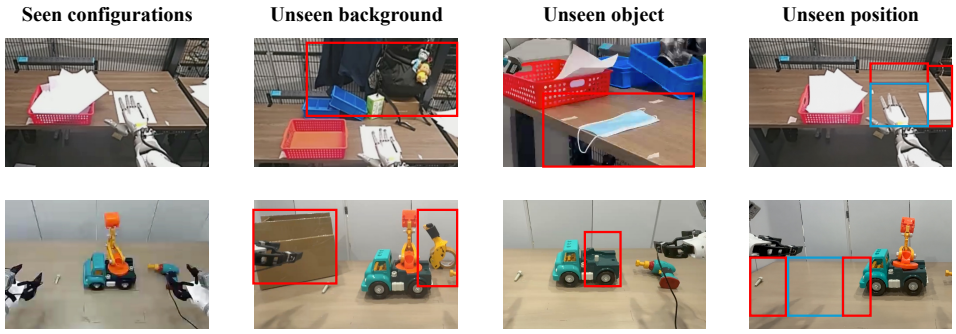


Figure 9: **Real-world generalization.** Examples of seen configurations and unseen variations in backgrounds, objects, and positions. Blue borders indicate configurations within the training distribution, while red borders indicate unseen configurations.

G.2 Evaluation under Unseen Configurations

We evaluate generalization on *Pick A4 Paper* and *Powered Screw Driving* under three types of distribution shift: unseen backgrounds, object instances, and object positions. Each factor is varied separately while the remaining conditions are controlled. All methods are tested on the same configurations using the success criteria defined above. Figure 9 illustrates these settings.

Unseen backgrounds. We introduce background distractors absent from the training demonstrations in both tasks, while retaining the task objects and their placement distribution.

Unseen objects. For *Pick A4 Paper*, we replace the A4 sheet with napkins, masks, notebooks, tennis balls, and sheets of different colors and sizes. The objective remains to pick up the object and place it entirely inside the basket, allowing the grasping strategy to vary with the object. For *Powered Screw Driving*, we use toy-car bases with unseen structural configurations, such as variants without wheels or the front section, while retaining the screw-fastening objective.

Unseen positions. We place the manipulated objects at workspace positions not represented in the training demonstrations, while retaining the training backgrounds and object instances.

H Ethics Statement

The public datasets used in this work are available under licenses permitting research use. Our in-lab data collection does not disclose personal information.

I Reproducibility Statement

We describe the model architecture in Section 3.2 and the simulation setup and evaluation protocol in Section 4.1. Appendix F provides additional implementation details for PhysWord and the baselines, covering both simulation and real-world experiments to support reproducibility.