

WorldAgen: Unified State-Action Prediction with Test-Time World Model Training

Chi Wan^{*1}, Kangrui Wang^{*1}, Yuan Si¹, Pingyue Zhang¹, Manling Li¹

¹Northwestern University

Abstract

How can vision-language-action (VLA) models adapt to new environments where world dynamics shift? While recent research has combined world modeling and action prediction to improve VLA performance, existing methods largely rely on pretraining in static datasets, without mechanisms for active adaptation to new environments. As a result, these models often fail to generalize when deployed in unseen scenarios with novel object configurations or dynamics. We present **WorldAgen**, a unified framework that jointly learns world modeling and action prediction while enabling **Test-Time Training (TTT)** to adapt to new environments. WorldAgen employs a shared Transformer backbone with two heads: (1) a **world model** head that predicts future states from past state-action trajectories, and (2) an **agent model** head that predicts actions conditioned on task instructions. We design a **Mixed Unidirectional Attention Mask** to separate these two models. During test time, WorldAgen samples exploratory actions, collects ground-truth state transitions, and performs lightweight TTT updates to refine its world model. This adaptation improves the model’s understanding to the environments and leads to more accurate action predictions. Experiments on the CALVIN and LIBERO benchmarks demonstrate that our baseline model achieves comparable, and in some cases superior, performance to current state-of-the-art approaches. Moreover, with TTT on a small number of samples, our method surpasses existing state-of-the-art models, highlighting effectiveness of adapting world models at inference time.

Introduction

Vision-language-action (VLA) models have emerged as a powerful and popular paradigm for robotic manipulation (Ma et al. 2025; Din et al. 2025), enabling agents to follow natural language instructions and act directly from raw visual observations. Recent work has explored joint state-action prediction techniques (Tian et al. 2024; Guo et al. 2024; Liu et al. 2025; Xian and Gkanatsios 2023; Wolf et al. 2025), allowing models not only to predict the next actions but also to anticipate how their actions will transform the environment. This joint formulation has improved data effi-

ciency and scene understanding, leading to stronger performance across manipulation benchmarks.

However, we argue that current methods remain fundamentally limited by training on static datasets (Kim, Finn, and Liang 2025; Li et al. 2025; Kim et al. 2024). Once pre-trained, these models lack mechanisms to adapt their internal representations of world dynamics when deployed in novel environments. In realistic settings, distribution shifts such as new object layouts, lighting conditions, or physical properties are inevitable, and static world modeling fails to capture these variations (Wu et al. 2023b; Lanier et al. 2025). This leads to a key question: **How can we enable VLA models to actively adapt their understanding of the environment during test time?**

As shown in Figure 1, we address this question with WorldAgen, a unified framework that combines joint state-action prediction with a novel TTT strategy:

- **Joint State-Action Modeling.** WorldAgen uses a shared Transformer backbone with two heads. The world-model head predicts future states from historical state-action trajectories, while the agent-model head predicts actions conditioned on task instructions. By training these tasks jointly, WorldAgen aligns scene understanding with action prediction.
- **Test-Time Training for Scene Adaptation.** During test time, WorldAgen executes exploratory actions to collect ground-truth state transitions. It then performs lightweight, LoRA-based TTT updates to refine its world model. These updates improve the internal representation of the environment, indirectly boosting the agent’s ability to generate effective actions in novel scenarios, which is depicted in.

Our work reframes world modeling from a passive pretraining objective into an active test time adaptation mechanism, bridging the gap between offline training and real-world generalization.

In sum, we make the following contributions:

- **Unified joint state-action prediction architecture.** We present a unified Transformer-based structure that integrates world modeling and action prediction, enabling shared representations and work separately via a Mixed Unidirectional Attention Mask.

^{*}These authors contributed equally.

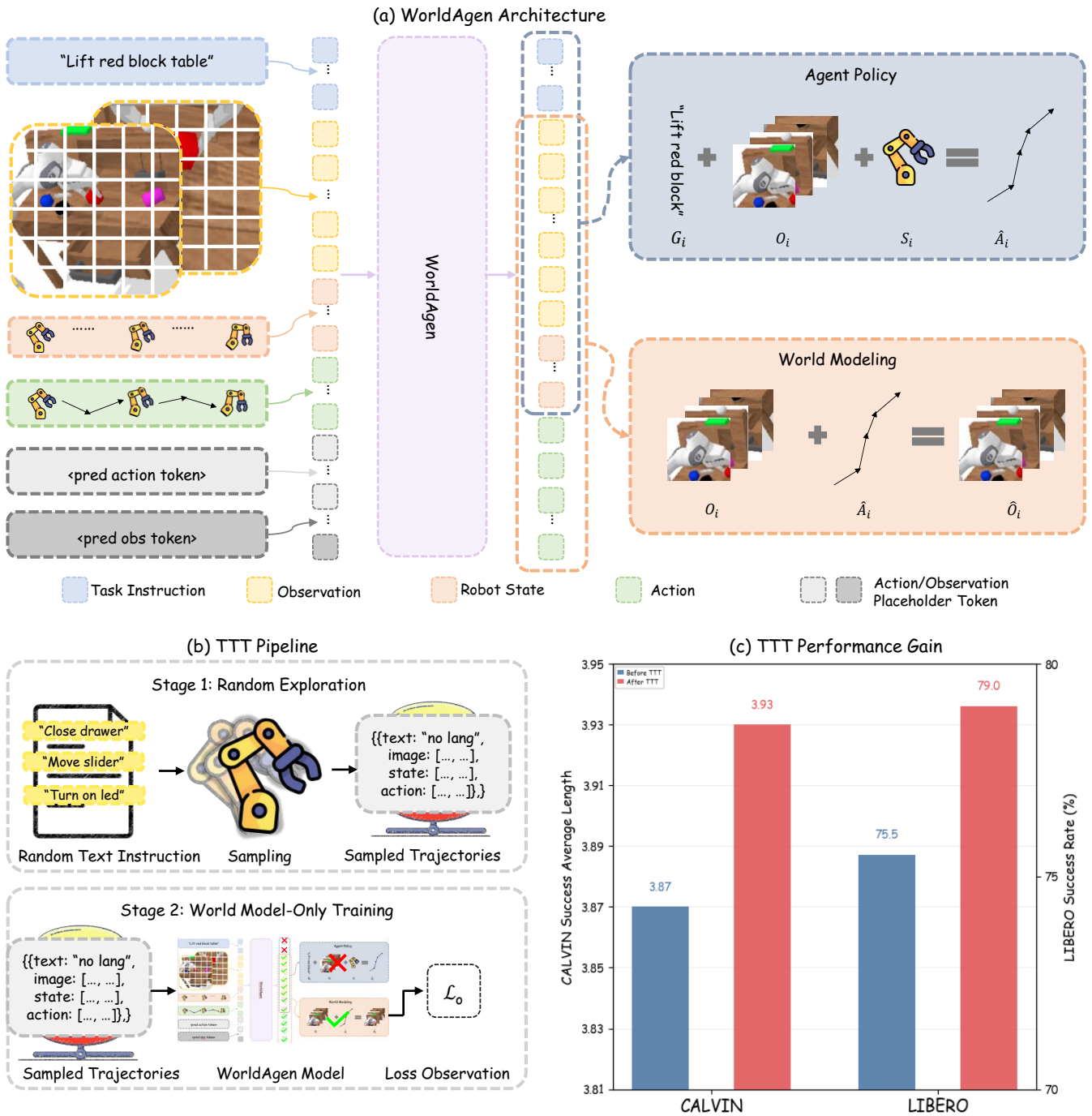


Figure 1: Overview of WorldAgen. (a) WorldAgen Architecture. WorldAgen unifies an agent policy head for task-conditioned action prediction and a world modeling head for task-agnostic state prediction within a single Transformer backbone. At trajectory unit i , the agent receives the task instruction G_i , the observation chunk O_i , and the robot state chunk S_i , the action placeholder A_p and observation placeholder O_p . Then it predicts the action chunk \hat{A}_i . After that, the world model predicts the future observation chunk \hat{O}_i to refine the shared representation of environment dynamics. (b) Two-stage TTT pipeline. Stage 1 performs random exploration in the new environment to collect unlabeled trajectories. Stage 2 adapts the world model using LoRA-based fine-tuning on the observation loss \mathcal{L}_o , improving environment modeling without modifying the agent policy. (c) TTT Performance We tested WorldAgen with TTT strategy on CALVIN and LIBERO benchmark, which show great performance gain.

- **Active test-time adaptation for VLA models.** We introduce a TTT paradigm that transforms world modeling from a static pretraining into an active, test time adaptation mechanism, allowing VLA models to refine their scene understanding on the fly.
- **Empirical validation across challenging benchmarks.** Our baseline achieves performance comparable to, or even better than, state-of-the-art methods. Furthermore, by fine-tuning the world modeling component with only a small number of samples at test time, our method attains state-of-the-art results on both the CALVIN and LIBERO benchmark.

We believe these results demonstrate that continuous adaptation during test time, rather than merely scaling model size or pretraining data, is a key ingredient for robust and generalizable robotic manipulation.

Method

Task Formulation

We frame our task as a Partially Observable Markov Decision Process (POMDP). Each rollout trajectory is divided into a sequence of trajectory units indexed by i . A trajectory unit is defined as $U_i = (G_i, O_i, S_i, A_i, A_p, O_p)$, where G_i denotes the task instruction, and O_i , S_i , and A_i denote the observation, robot state, and action chunks, respectively, accompanied by an action placeholder A_p and an observation placeholder O_p .

Each chunk (i.e., O_i , S_i , and A_i) consists of a sequence of time-indexed elements, including observations, states, or actions extracted from the raw trajectory τ . Each element (i.e., o_t , s_t , a_t) is further tokenized into a sequence of tokens that serves as the actual model input.

In unit i , the model first predicts an action chunk \hat{A}_i conditioned on the instruction G_i , the observation chunk O_i , and the state chunk S_i . It then predicts the next observation chunk conditioned on (O_i, S_i, A_i) during training or (O_i, S_i, \hat{A}_i) during inference¹. Executing the predicted actions advances the environment and produces the next unit U_{i+1} . A summary of the key notations used in the WorldAgen framework is provided in Table 12 in Appendix.

Joint State–Action Prediction

WorldAgen contains two predictive components: a task-conditioned agent model and a task-agnostic world model. Both models share a single Transformer backbone, enabling a unified representation across tasks, as illustrated in Figure 1(a).

We maintain two types of histories:

Agent history:

$$h_i^a = (G_i, O_x, S_x, \hat{A}_x)_{x=i-k}^{i-1},$$

which contains the task instruction G_i along with the observation, robot state, and predicted action chunks from the previous k segments. This history is used by the agent model for task-aware action prediction.

¹A history buffer h is also included in the conditioning; detailed formulation is provided in a later section.

World history:

$$h_i^w = (O_x, S_x, \hat{A}_x)_{x=i-k}^{i-1},$$

which includes the observation, robot state, and predicted action chunks from the previous k segments, and is used by the world model to predict task-agnostic environment dynamics.

We define the two models at trajectory segment U_i as follows:

Agent Model. The agent model predicts the next action chunk \hat{A}_i conditioned on the current observation chunk O_i , robot state chunk S_i , task instruction G_i , and agent history h_i^a :

$$p_a(\hat{A}_i | G_i, O_i, S_i, h_i^a).$$

World Model. The world model predicts the next observation chunk \hat{O}_i conditioned on the current observation chunk O_i , robot state chunk S_i , predicted action chunk \hat{A}_i , and world history h_i^w :

$$p_w(\hat{O}_i | O_i, S_i, \hat{A}_i, h_i^w).$$

During training, we adopt teacher forcing and replace all predicted actions \hat{A}_i with the ground-truth actions A_i .

Trajectory Splitting and Action Chunking

Predicting multiple actions at once has been shown to improve the execution efficiency of real-world robots (Intelligence et al. 2025). However, consecutive observation frames often contain highly redundant visual information, and predicting long observation sequences slows down both training and deployment. To balance these factors, WorldAgen supports different action chunk lengths m and observations chunk length n . We further propose a robust trajectory splitting and action chunking scheme that flexibly adapts to different experimental configurations.

Concretely, given a rollout $\tau = \{(o_t, s_t, a_t)\}_{t=0}^{T-1}$ with T timesteps, we transform it into a sequence of trajectory units $\{U_i\}_{i=0}^{I-1}$ with I units. The initial unit U_0 aggregates the first n timesteps of the rollout and is paired with the first action chunk $A_0 = A_{0 \rightarrow n-1} = [a_0, a_1, \dots, a_{n-1}]$. For units U_i with $i > 0$, we uniformly sample n timesteps to construct the current observation chunk O_i , state chunk S_i , and associated action chunk A_i :

$$\begin{aligned} O_i &= O_{n+(i-1)m \rightarrow n+im-1} \\ &= [o_{n+im}, o_{n+im+\lfloor \frac{1}{n}m \rfloor}, \dots, o_{n+im+\lfloor \frac{n-1}{n}m \rfloor}], \\ S_i &= S_{n+(i-1)m \rightarrow n+im-1} \\ &= [s_{n+(i-1)m}, s_{n+(i-1)m+\lfloor \frac{1}{n}m \rfloor}, \\ &\quad \dots, s_{n+(i-1)m+\lfloor \frac{n-1}{n}m \rfloor}], \\ A_i &= A_{n+(i-1)m \rightarrow n+im-1} \\ &= [a_{n+(i-1)m}, a_{n+(i-1)m+1}, \dots, a_{n+im-1}]. \end{aligned}$$

This procedure keeps each observation chunk fixed to length n , reducing redundant visual information and improving runtime efficiency.

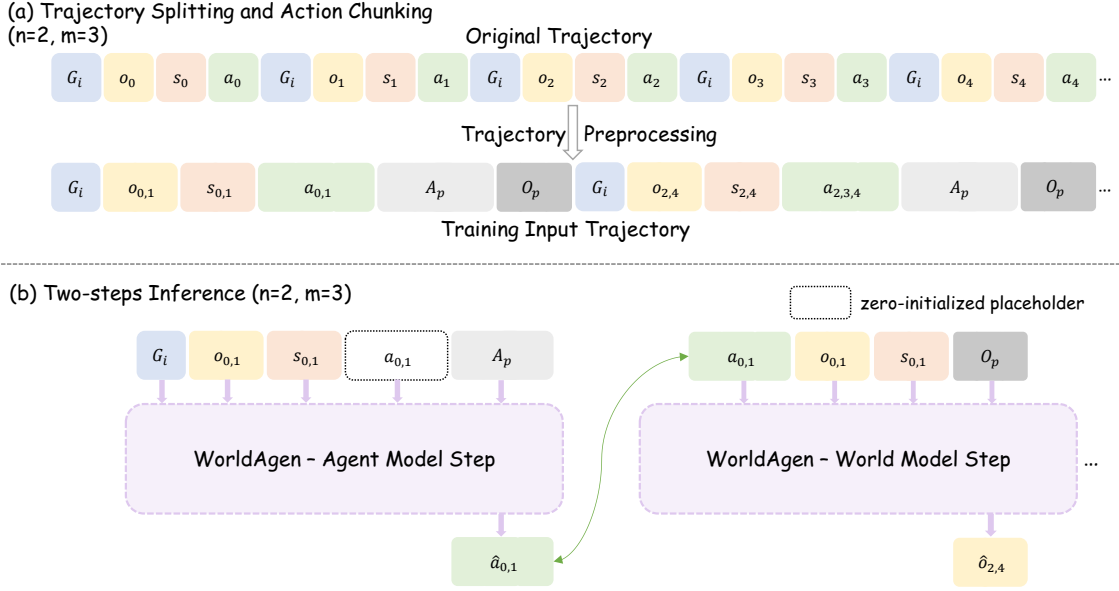


Figure 2: **Trajectory Splitting, Chunking, and Two-Step Inference.** Example with observation chunk length $n = 2$ and action chunk length $m = 3$. (a) *Trajectory Splitting and Chunking*: a raw trajectory is divided into trajectory units U_i , where each unit contains an observation chunk O_i and a robot state chunk S_i obtained via uniform subsampling from m consecutive observations and states, and an action chunk A_i obtained by grouping m actions. (b) *Two-Step Inference*: the agent model first initializes a zero action placeholder A_p and then replaces it with the predicted action chunk \hat{A}_i . The world model subsequently predicts the next observation chunk \hat{O}_i , conditioned on the updated trajectory unit.

Figure 2 (a) illustrates an example with $n = 2$ and $m = 3$. Algorithm of trajectory splitting and action chunking for variable length of action, observation and robot state chunk is shown in Algorithm 1 in Appendix , where $\text{UNIFORMSUBSAMPLE}(a, b, n)$ uniformly samples n distinct indices from the integer interval $\{a, \dots, b\}$ without replacement and returns them in ascending order, which we then use to construct the observation and state chunks.

Mixed Unidirectional Attention Mask

To ensure that the world modeling and agent policy tasks are trained jointly without information leakage, we introduce a mixed unidirectional attention masking mechanism as shown in Figure 3. This mechanism integrates two complementary components:

Local mask: ensures that the current observation chunk O_i , the robot proprioception chunk S_i , the task instruction G_i , and the action placeholder A_p cannot access the action chunk A_i . This prevents information leakage within a single timestep.

Global mask: guarantees that the world modeling head remains invisible to the task instruction G_i across all time steps. Additionally, the global mask enforces mutual invisibility among the task instruction and observations to avoid cross-modal leakage.

Meanwhile, the general structure conforms to strictly causal prediction, such that both the world modeling and policy heads cannot access any future tokens.

Pretraining

We pretrain WorldAgen using large-scale trajectory data collected from robot demonstrations. We split trajectory using our proposed robust splitting method. During pretraining, we jointly optimize the agent model and world model using teacher forcing:

$$\mathcal{L} = \mathcal{L}_a + \lambda \mathcal{L}_o,$$

where \mathcal{L}_a is the cross-entropy loss for predicting future action chunks, \mathcal{L}_o is the reconstruction loss for predicting future observation chunks, and λ is a weighting factor balancing the two objectives.

It is worth noting that during training, we feed the ground-truth action chunk (teacher forcing), whereas during testing, the model performs autoregressive prediction by feeding the previously predicted action chunk at each timestep.

Two-Step Inference

At test time, WorldAgen runs in a two-step inference loop over trajectory units as shown in Figure 2 (b). For unit i , we first perform the agent model step. We construct the input sequence with a zero-initialized action placeholder A_p and feed it to the agent model, which predicts the current action chunk \hat{A}_i . The predicted actions then overwrite the placeholder A_p .

Next, we perform the world model step. Conditioned on the task instruction G_i , the current observation chunk O_i and the updated action chunk A_i (same as \hat{A}_i), the world model

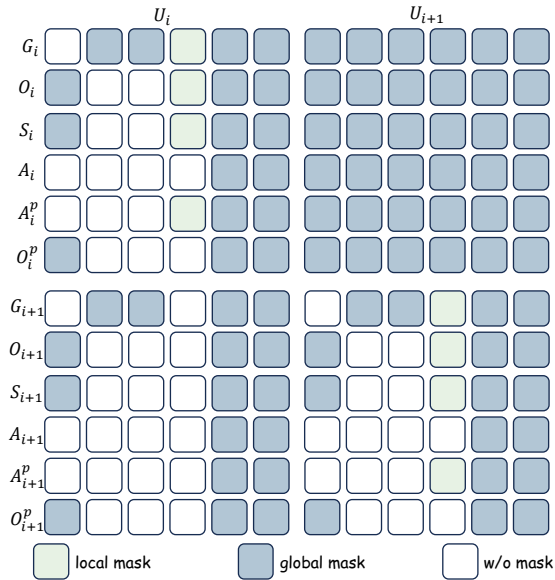


Figure 3: Proposed Mixed Unidirectional Attention Mask: combines local and global masking. For each trajectory unit, WorldAgen applies a local mask that prevents the action chunk A_i from attending to the corresponding action placeholder A_p . A global mask is further used to construct a task-agnostic world-modeling head, so that the observation placeholder O_p will not attend to the task instruction G_i . General structure conforms to strictly causal prediction.

predicts the next observation chunk \hat{O}_i . During testing, these two steps are applied alternately to roll out the whole trajectory .

Test-Time Training

While joint pretraining improves world understanding, distribution shifts in novel environments can still degrade performance. To address this, we introduce a two-stage Test-Time Training strategy that adapts the world model online, refining its understanding of environment dynamics and indirectly improving downstream action prediction. The whole pipeline is given in Figure 1 (b).

Stage 1: Random Exploration and Data Collection. At deployment, the agent first performs free exploratory roll-outs in the target environment. During this phase, it records trajectories of (O_i, S_i, A_i) . To ensure that adaptation focuses on environment dynamics rather than task-specific instructions, all collected trajectories are relabeled with a generic “no-lang” token in place of language instructions.

Stage 2: World Model Adaptation. Using the collected trajectories, we adapt only the shared backbone with LoRA-based parameter-efficient fine-tuning, while keeping the agent policy head frozen. The update rule is:

$$\theta'_w \leftarrow \theta_w - \eta \nabla_{\theta_w} \mathcal{L}_o,$$

where θ_w are the parameters of the world model head and η is the learning rate.

Method	T1	T2	T3	T4	T5	Avg. Len. \uparrow
RoboFlamingo	82.4	61.9	46.6	33.1	23.5	2.47
SuSIE	87.0	69.0	49.0	38.0	26.0	2.69
GR-1	85.4	71.2	59.6	49.7	40.1	3.06
3D Diffusor Actor	92.2	78.7	63.9	51.2	41.2	3.27
CLOVER	96.0	83.5	70.8	57.5	45.4	3.53
Seer	93.0	82.4	72.3	62.6	53.3	3.64
Seer-Large	92.7	84.6	76.1	68.9	60.3	3.83
WorldAgen	96.3	87.7	76.8	67.3	59.1	3.87
WorldAgen-TTT	96.6	88.5	78.5	68.7	60.5	3.93

Table 1: Performance comparison on the CALVIN benchmark. We report the success rate (%) for completing 5 consecutive tasks and the average sequence length (ASL).

This targeted update improves world understanding ability and enhances the shared representations, which are in turn leveraged by the agent model for action prediction.

By decoupling adaptation from the task goal and restricting it to the world model, **our TTT procedure enables robust scene adaptation without requiring additional task-specific annotations, paving the way for scalable test-time adaptation in VLA models.**

Experiments

Datasets

CALVIN. CALVIN (Mees et al. 2022) is an open-source simulated benchmark designed for learning long-horizon language-conditioned robot manipulation tasks. The benchmark requires agents to solve complex manipulation tasks by understanding a series of unconstrained language instructions in sequence.

LIBERO. LIBERO (Liu et al. 2023a) is a comprehensive benchmark for lifelong learning in robot manipulation that emphasizes knowledge transfer across diverse tasks.

Implementation Details

Test-Time Training TTT represents a crucial component of WorldAgen that enables adaptive performance improvements during inference. In the TTT phase, we apply LoRA fine-tuning to the backbone of Qwen3. Specifically, we apply LoRA to the attention projection layers (q-proj, k-proj, v-proj, o-proj) and the MLP projection layers (gate-proj, up-proj, down-proj) of Qwen3.

The test-time training procedure, including sampling and LoRA adaptation, can be executed on an RTX 4090 GPU and require approximately 8 minutes per task for CALVIN and 2 minutes per scene. More details can be found in Appendix.

Results

Our experimental results demonstrate the effectiveness of WorldAgen across both CALVIN and LIBERO benchmarks. **CALVIN** As shown in Table 1, we compare WorldAgen with recent VLA baselines, including RoboFlamingo (Li et al. 2023), SuSIE (Black et al. 2023), GR-1 (Wu et al.

Method	Avg. Success \uparrow	Put soup and box in basket	Put box and butter in basket	Turn on stove and put pot	Put bowl in drawer and close it
MT-ACT	41.0	30.0	50.0	75.0	85.0
MVP	68.2	83.3	90.0	80.0	88.3
MPL	77.3	66.6	86.6	96.6	95.0
OpenVLA	54.0	35.0	95.0	65.0	45.0
Seer	78.7	80.0	90.0	91.7	81.7
WorldAgen	75.5	70.0	75.0	95.0	100
WorldAgen-TTT	79.0	85.0	75.0	95.0	100

Put mugs on left and right plates	Pick book and place it in back	Put mug on plate and put pudding to right	Put soup and sauce in basket	Put both pots on stove	Put mug in microwave and close it
20.0	75.0	0.0	0.0	10.0	65.0
46.7	63.3	45.0	78.3	60.0	46.7
83.3	83.3	56.6	86.6	40.0	78.3
40.0	80.0	60.0	45.0	20.0	55.0
85.0	65.0	86.7	88.3	51.7	66.7
85.0	90.0	60.0	100	45.0	35.0
90.0	90.0	50.0	100	45.0	60.0

Table 2: Performance comparison on LIBERO benchmark. We report the success rate (%) across different manipulation tasks.

2023a), 3D Diffusor Actor (Ke, Gkanatsios, and Fragkiadaki 2024), and CLOVER (Bu et al. 2024). WorldAgen follows a GR-1-style GPT video architecture, but introduces the Mixed Unidirectional Attention Mask to separate task instruction from world modeling and employs a lightweight TTT strategy to enhance world-modeling capability. Our method achieves consistent performance improvements across five consecutive tasks, indicating that *strengthening world modeling via TTT improves environment understanding and boosts long-horizon manipulation performance*.

LIBERO As shown in Table 2, we further evaluate WorldAgen on the LIBERO benchmark against multi-task VLA baselines such as MT-ACT (Bharadhwaj et al. 2024), MVP (Xiao et al. 2022), MPI (Zeng et al. 2024), OpenVLA (Kim et al. 2024) and Seer (Tian et al. 2024). WorldAgen attains the best or on-par performance across most LIBERO subsets, further supporting that test-time adaptation of the world model leads to stronger environment understanding and improved robustness to distribution shifts.

We observe improvements or maintained performance across almost every task compared to the baseline. These consistent gains across diverse manipulation scenarios mirror our findings on CALVIN, demonstrating *robustness and generalization ability* of our TTT approach across different benchmarks and scenario dynamics.

Ablation Study

To better understand the key components contributing to performance, we conduct ablation studies on (1) the contribution of world modeling to policy learning, (2) the effect of

LoRA parameterization during TTT, (3) the impact of the amount of randomly sampled data used for TTT, and (4) transfer from simulator to the real robot. We further analyze the impact of trajectory splitting and preprocessing, image and action chunk lengths, and different Transformer backbones, and compare LoRA-based TTT with full fine-tuning while visualizing performance before and after TTT. All ablation and additional experimental results are provided in the Appendix.

Dataset	World Modeling	Avg. Success \uparrow
CALVIN	×	2.96
	✓	3.87
LIBERO	×	46.5%
	✓	78.0%

Table 3: Ablation study on world modeling. We compare models with and without image prediction to evaluate the contribution of world modeling to agent policy learning.

World Modeling Ability To validate the contribution of world modeling to policy learning, we compare models with and without image prediction capability by including or removing image-prediction tokens in the input as shown in Table 3. World modeling consistently improves performance: on CALVIN, the average success length increases from 2.961 to 3.87 (+30.7%), and on LIBERO, the success rate rises from 46.5% to 78.0% (+67.7%). These results indicate that *world modeling yields richer representations of environment dynamics, leading to more effective action*

prediction and policy learning.

LoRA Rank	Avg. Len. \uparrow
16	3.928
32	3.918
64	3.918
128	3.930
256	3.923

Table 4: Ablation study on LoRA rank during TTT in CALVIN dataset. All other parameters are kept constant.

LoRA Configuration We study the effect of LoRA rank on TTT performance while fixing the learning rate and training data size as shown in Table 4. Performance only varies within 0.02 across ranks, indicating that *under fixed hyper-parameters, TTT performance is largely insensitive to the LoRA rank*. We therefore use a LoRA rank of 128 in our main experiments, which slightly outperforms other settings while remaining computationally efficient.

Test Time Training Data	Avg. Lens. \uparrow
6	3.871
90	3.922
204	3.928
340	3.917

Table 5: Ablation study on TTT data sampling volume. We vary the amount of test-time training data, defined as the product of number of samples and repeat times.

TTT Data Sampling Volume We study how the amount of test-time training data affects performance as shown in Table 5. As the number of TTT samples increases from 6 to 204, the average success score improves from 3.871 to 3.928, indicating that more diverse exploration data benefits world-model adaptation. However, further increasing the data to 340 slightly reduces performance to 3.917, suggesting diminishing returns and mild overfitting to image generation rather than action prediction. These results indicate that *moderately increasing TTT data improves performance, but excessive data can be counterproductive, highlighting the need for an appropriate TTT data budget*.

	w/o TTT	w/o Noise	w/ Noise
CALVIN	3.87	3.93	3.90
LIBERO	75.5	79.0	78.0

Table 6: Performance comparison under Gaussian noise perturbation.

Simulator to Real World We add Gaussian noise (std = 0.1) to the observations to simulate real-world conditions, such as camera distortion and sensor noise, as shown in Table 6.

This experiment indicates that *even under noisy conditions, our method consistently outperforms the baselines, demonstrating its robustness and strong adaptability to real-world uncertainty*.

Related Work

Vision-Language-Action Models. Vision-Language-Action (VLA) models unify perception, language, and control for robotic manipulation (Sapkota et al. 2025; Din et al. 2025). Early systems such as RT-1 and RT-2 established the effectiveness of large-scale transformer policies, while PaLM-E and LLaVA further incorporated multimodal grounding (Brohan et al. 2023; Zitkovich et al. 2023; Driess et al. 2023; Liu et al. 2023b). However, most VLA models directly predict actions without explicitly modeling world dynamics, limiting generalization to novel environments (Zhang et al. 2025).

World Models. World models learn predictive dynamics to support planning and decision-making (Sutton 1990; Hafner et al. 2025). Neural approaches such as World Models and Dreamer improve sample efficiency and long-horizon reasoning (Ha and Schmidhuber 2018; Hafner et al. 2022, 2025). In robotics, however, these models are typically used for state prediction or model-based planning and remain largely decoupled from language-grounded action generation (Sakagami et al. 2023).

Test-Time Training. Test-time training adapts models to distribution shifts during inference via self-supervised objectives (Sun et al. 2020; Ma 2024). While TTT has been explored in both vision and NLP (Ye et al. 2023), robotic adaptation has mainly focused on perception modules or low-level policies (Zhao, Queralta, and Westerlund 2020). Our work instead incorporates TTT into a unified VLA framework by adapting the world model itself during deployment.

Conclusion

We introduced **WorldAgen**, a unified vision–language–action framework that jointly learns world modeling and action prediction through a shared Transformer backbone. WorldAgen integrates a world-model head that predicts future states and an agent head that predicts task-conditioned actions. At test time, the model performs lightweight Test-Time Training (TTT) using short exploratory rollouts, turning world modeling into an online adaptation signal. This enables the agent to quickly adjust to new environments and leads to consistent performance gains on CALVIN and LIBERO, demonstrating that adaptive VLA systems can benefit substantially from continual test-time interaction.

Limitations and Future Work

Our study evaluates only a single model size and architecture, and focuses test-time training exclusively on the world-model component. Future work includes exploring larger and more diverse model architectures, as well as developing joint test-time-training methods that improve both action prediction and world modeling.

References

- Bharadhwaj, H.; Vakil, J.; Sharma, M.; Gupta, A.; Tulsiani, S.; and Kumar, V. 2024. Roboagent: Generalization and efficiency in robot manipulation via semantic augmentations and action chunking. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 4788–4795. IEEE.
- Black, K.; Nakamoto, M.; Atreya, P.; Walke, H.; Finn, C.; Kumar, A.; and Levine, S. 2023. Zero-shot robotic manipulation with pretrained image-editing diffusion models. *arXiv preprint arXiv:2310.10639*.
- Brohan, A.; Brown, N.; Carbajal, J.; Chebotar, Y.; Dabis, J.; Finn, C.; Gopalakrishnan, K.; Hausman, K.; Herzog, A.; Hsu, J.; Ibarz, J.; Ichter, B.; Irpan, A.; Jackson, T.; Jesmonth, S.; Joshi, N. J.; Julian, R.; Kalashnikov, D.; Kuang, Y.; Leal, I.; Lee, K.-H.; Levine, S.; Lu, Y.; Malla, U.; Manjunath, D.; Mordatch, I.; Nachum, O.; Parada, C.; Peralta, J.; Perez, E.; Pertsch, K.; Quiambao, J.; Rao, K.; Ryoo, M.; Salazar, G.; Sanketi, P.; Sayed, K.; Singh, J.; Sontakke, S.; Stone, A.; Tan, C.; Tran, H.; Vanhoucke, V.; Vega, S.; Vuong, Q.; Xia, F.; Xiao, T.; Xu, P.; Xu, S.; Yu, T.; and Zitkovich, B. 2023. RT-1: Robotics Transformer for Real-World Control at Scale. *arXiv:2212.06817*.
- Bu, Q.; Zeng, J.; Chen, L.; Yang, Y.; Zhou, G.; Yan, J.; Luo, P.; Cui, H.; Ma, Y.; and Li, H. 2024. Closed-loop visuomotor control with generative expectation for robotic manipulation. *Advances in Neural Information Processing Systems*, 37: 139002–139029.
- Din, M. U.; Akram, W.; Saoud, L. S.; Rosell, J.; and Hus-sain, I. 2025. Vision Language Action Models in Robotic Manipulation: A Systematic Review. *arXiv:2507.10672*.
- Driess, D.; Xia, F.; Sajjadi, M. S.; Lynch, C.; Chowdhery, A.; Wahid, A.; Tompson, J.; Vuong, Q.; Yu, T.; Huang, W.; et al. 2023. Palm-e: An embodied multimodal language model.
- Guo, Y.; Hu, Y.; Zhang, J.; Wang, Y.-J.; Chen, X.; Lu, C.; and Chen, J. 2024. Prediction with action: Visual policy learning via joint denoising process. *Advances in Neural Information Processing Systems*, 37: 112386–112410.
- Ha, D.; and Schmidhuber, J. 2018. World Models. *CoRR*.
- Hafner, D.; Lillicrap, T.; Norouzi, M.; and Ba, J. 2022. Mastering Atari with Discrete World Models. *arXiv:2010.02193*.
- Hafner, D.; Pasukonis, J.; Ba, J.; and Lillicrap, T. 2025. Mastering diverse control tasks through world models. *Nature*, 1–7.
- He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 16000–16009.
- Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W.; et al. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2): 3.
- Intelligence, P.; Black, K.; Brown, N.; Darpinian, J.; Dhabalia, K.; Driess, D.; Esmail, A.; Equi, M.; Finn, C.; Fusai, N.; et al. 2025. $\pi_{0.5}$: a Vision-Language-Action Model with Open-World Generalization. *arXiv preprint arXiv:2504.16054*.
- Jaegle, A.; Gimeno, F.; Brock, A.; Vinyals, O.; Zisserman, A.; and Carreira, J. 2021. Perceiver: General perception with iterative attention. In *International conference on machine learning*, 4651–4664. PMLR.
- Ke, T.-W.; Gkanatsios, N.; and Fragkiadaki, K. 2024. 3d diffuser actor: Policy diffusion with 3d scene representations. *arXiv preprint arXiv:2402.10885*.
- Kim, M. J.; Finn, C.; and Liang, P. 2025. Fine-Tuning Vision-Language-Action Models: Optimizing Speed and Success. *arXiv:2502.19645*.
- Kim, M. J.; Pertsch, K.; Karamcheti, S.; Xiao, T.; Balakrishna, A.; Nair, S.; Rafailov, R.; Foster, E.; Lam, G.; Sanketi, P.; Vuong, Q.; Kollar, T.; Burchfiel, B.; Tedrake, R.; Sadigh, D.; Levine, S.; Liang, P.; and Finn, C. 2024. Open-VLA: An Open-Source Vision-Language-Action Model. *arXiv:2406.09246*.
- Kojima, Y.; Xu, J.; Zou, X.; and Wang, X. 2025. LoRA-TTT: Low-Rank Test-Time Training for Vision-Language Models. *arXiv:2502.02069*.
- Lanier, J.; Kim, K.; Karamzade, A.; Liu, Y.; Sinha, A.; He, K.; Corsi, D.; and Fox, R. 2025. Adapting World Models with Latent-State Dynamics Residuals. *arXiv:2504.02252*.
- Li, P.; Wu, Y.; Xi, Z.; Li, W.; Huang, Y.; Zhang, Z.; Chen, Y.; Wang, J.; Zhu, S.-C.; Liu, T.; and Huang, S. 2025. ControlVLA: Few-shot Object-centric Adaptation for Pre-trained Vision-Language-Action Models. *arXiv:2506.16211*.
- Li, X.; Liu, M.; Zhang, H.; Yu, C.; Xu, J.; Wu, H.; Cheang, C.; Jing, Y.; Zhang, W.; Liu, H.; et al. 2023. Vision-language foundation models as effective robot imitators. *arXiv preprint arXiv:2311.01378*.
- Liu, B.; Zhu, Y.; Gao, C.; Feng, Y.; Liu, Q.; Zhu, Y.; and Stone, P. 2023a. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36: 44776–44791.
- Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023b. Visual instruction tuning. *Advances in neural information processing systems*, 36: 34892–34916.
- Liu, S.; Wu, L.; Li, B.; Tan, H.; Chen, H.; Wang, Z.; Xu, K.; Su, H.; and Zhu, J. 2025. RDT-1B: a Diffusion Foundation Model for Bimanual Manipulation. *arXiv:2410.07864*.
- Ma, J. 2024. Improved self-training for test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 23701–23710.
- Ma, Y.; Song, Z.; Zhuang, Y.; Hao, J.; and King, I. 2025. A Survey on Vision-Language-Action Models for Embodied AI. *arXiv:2405.14093*.
- Mees, O.; Hermann, L.; Rosete-Beas, E.; and Burgard, W. 2022. CALVIN: A Benchmark for Language-Conditioned Policy Learning for Long-Horizon Robot Manipulation Tasks. *IEEE Robotics and Automation Letters (RA-L)*, 7(3): 7327–7334.
- Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.

- Sakagami, R.; Lay, F. S.; Dömel, A.; Schuster, M. J.; Albu-Schäffer, A.; and Stulp, F. 2023. Robotic world models—conceptualization, review, and engineering best practices. *Frontiers in Robotics and AI*, 10: 1253049.
- Sapkota, R.; Cao, Y.; Roumeliotis, K. I.; and Karkee, M. 2025. Vision-Language-Action Models: Concepts, Progress, Applications and Challenges. [arXiv:2505.04769](https://arxiv.org/abs/2505.04769).
- Sun, Y.; Wang, X.; Liu, Z.; Miller, J.; Efros, A. A.; and Hardt, M. 2020. Test-Time Training with Self-Supervision for Generalization under Distribution Shifts. [arXiv:1909.13231](https://arxiv.org/abs/1909.13231).
- Sutton, R. S. 1990. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *Machine learning proceedings 1990*, 216–224. Elsevier.
- Tian, Y.; Yang, S.; Zeng, J.; Wang, P.; Lin, D.; Dong, H.; and Pang, J. 2024. Predictive Inverse Dynamics Models are Scalable Learners for Robotic Manipulation. [arXiv:2412.15109](https://arxiv.org/abs/2412.15109).
- Wolf, R.; Shi, Y.; Liu, S.; and Rayyes, R. 2025. Diffusion Models for Robotic Manipulation: A Survey. [arXiv:2504.08438](https://arxiv.org/abs/2504.08438).
- Wu, H.; Jing, Y.; Cheang, C.; Chen, G.; Xu, J.; Li, X.; Liu, M.; Li, H.; and Kong, T. 2023a. Unleashing large-scale video generative pre-training for visual robot manipulation. *arXiv preprint arXiv:2312.13139*.
- Wu, P.; Escontrela, A.; Hafner, D.; Abbeel, P.; and Goldberg, K. 2023b. Daydreamer: World models for physical robot learning. In *Conference on robot learning*, 2226–2240. PMLR.
- Xian, Z.; and Gkanatsios, N. 2023. Chaineddiffuser: Unifying trajectory diffusion and keypose prediction for robotic manipulation. In *Conference on Robot Learning/Proceedings of Machine Learning Research*. Proceedings of Machine Learning Research.
- Xiao, T.; Radosavovic, I.; Darrell, T.; and Malik, J. 2022. Masked visual pre-training for motor control. *arXiv preprint arXiv:2203.06173*.
- Yang, A.; Li, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Gao, C.; Huang, C.; Lv, C.; Zheng, C.; Liu, D.; Zhou, F.; Huang, F.; Hu, F.; Ge, H.; Wei, H.; Lin, H.; Tang, J.; Yang, J.; Tu, J.; Zhang, J.; Yang, J.; Yang, J.; Zhou, J.; Zhou, J.; Lin, J.; Dang, K.; Bao, K.; Yang, K.; Yu, L.; Deng, L.; Li, M.; Xue, M.; Li, M.; Zhang, P.; Wang, P.; Zhu, Q.; Men, R.; Gao, R.; Liu, S.; Luo, S.; Li, T.; Tang, T.; Yin, W.; Ren, X.; Wang, X.; Zhang, X.; Ren, X.; Fan, Y.; Su, Y.; Zhang, Y.; Zhang, Y.; Wan, Y.; Liu, Y.; Wang, Z.; Cui, Z.; Zhang, Z.; Zhou, Z.; and Qiu, Z. 2025. Qwen3 Technical Report. [arXiv:2505.09388](https://arxiv.org/abs/2505.09388).
- Ye, H.; Ding, Y.; Li, J.; and Ng, H. T. 2023. Robust Question Answering against Distribution Shifts with Test-Time Adaptation: An Empirical Study. [arXiv:2302.04618](https://arxiv.org/abs/2302.04618).
- Zeng, J.; Bu, Q.; Wang, B.; Xia, W.; Chen, L.; Dong, H.; Song, H.; Wang, D.; Hu, D.; Luo, P.; et al. 2024. Learning manipulation by predicting interaction. *arXiv preprint arXiv:2406.00439*.
- Zhang, J.; Wu, S.; Luo, X.; Wu, H.; Gao, L.; Shen, H. T.; and Song, J. 2025. InSpire: Vision-Language-Action Models with Intrinsic Spatial Reasoning. [arXiv:2505.13888](https://arxiv.org/abs/2505.13888).
- Zhao, W.; Queralta, J. P.; and Westerlund, T. 2020. Sim-to-real transfer in deep reinforcement learning for robotics: a survey. In *2020 IEEE symposium series on computational intelligence (SSCI)*, 737–744. IEEE.
- Zitkovich, B.; Yu, T.; Xu, S.; Xu, P.; Xiao, T.; Xia, F.; Wu, J.; Wohlhart, P.; Welker, S.; Wahid, A.; et al. 2023. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, 2165–2183. PMLR.