VLA-Corrector

Lightweight Detect-and-Correct Inference for Adaptive Action Horizon

Yi Pan, Miao Pan, Qi Lu, Jiaming Huang, Man Zhang, Siteng Huang, Xin Li, Jie Zhang, Yongliang Shen, Xuhong Zhang, Wenqi Zhang

Code: github.com/ZJU-OmniAI/vla-corrector · arXiv: arxiv.org/abs/2607.01804

Open-loop VLA execution leaves blind spots.

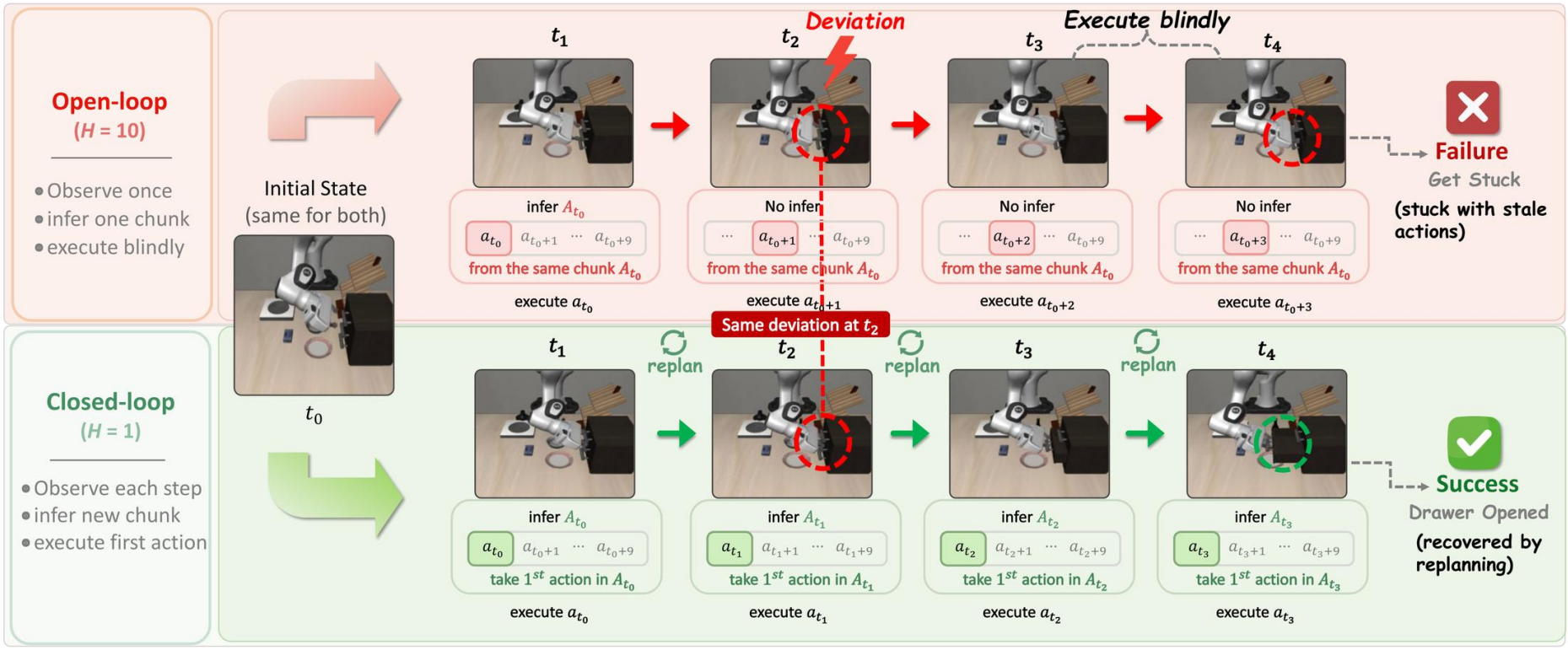
1. Action chunks make VLA execution efficient, but also open-loop.

2. When unexpected drift occurs, the robot may keep executing stale actions.

3. This blind spot can turn small errors into task failure.

Task failed because of stale action

Action horizon brings performance-efficiency trade-off



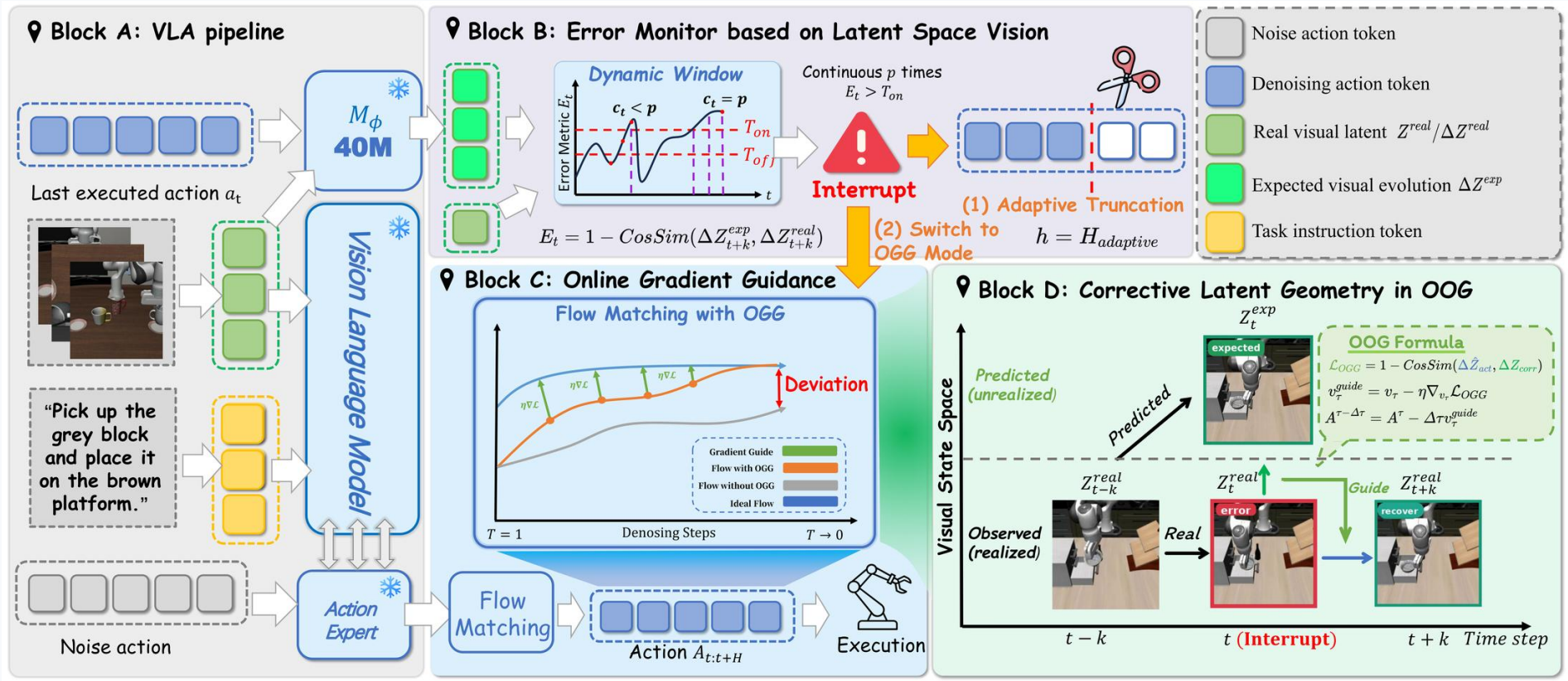
Long action horizon

Efficient VLA calls, but queued actions can become stale after drift.

Per-step replanning

More reactive, but expensive for large VLA policies.

Detect drift, truncate stale chunks, guide recovery.



01 Frozen VLA backbone with an external latent dynamics corrector.

02 Latent-space Vision Monitor detects persistent visual dynamics mismatch.

03 Online Gradient Guidance is applied only to the recovery query.

03 CORRECTOR

A small external model instead of full VLA retraining.

Trainable component

~40M

Residual MLP corrector at roughly 38-42M parameters.

Policy backbone

Frozen

The base VLA is not fully retrained.

Action horizon

Adaptive

Stable chunks continue; stale chunks are interrupted.

- VLA-Corrector shows that robust VLA execution does not require heavier backbones or constant replanning.
- By adding a **lightweight detect-and-correct layer** at inference time, it monitors latent visual dynamics, interrupts **stale action chunks**, and guides recovery only when execution drifts.
- This provides a practical path to overcome **open-loop blind spots** with **low cost**, turning fixed horizons into **adaptive, self-correcting execution**.

04 RESULTS

Reported gains across simulation and real robots.

MetaWorld / PI0.5

+15.65 pts

48.70% → 64.35% avg. success

LIBERO / PI0.5

+3.80 pts

94.00% → 97.80% few-shot success

AgileX PiPER

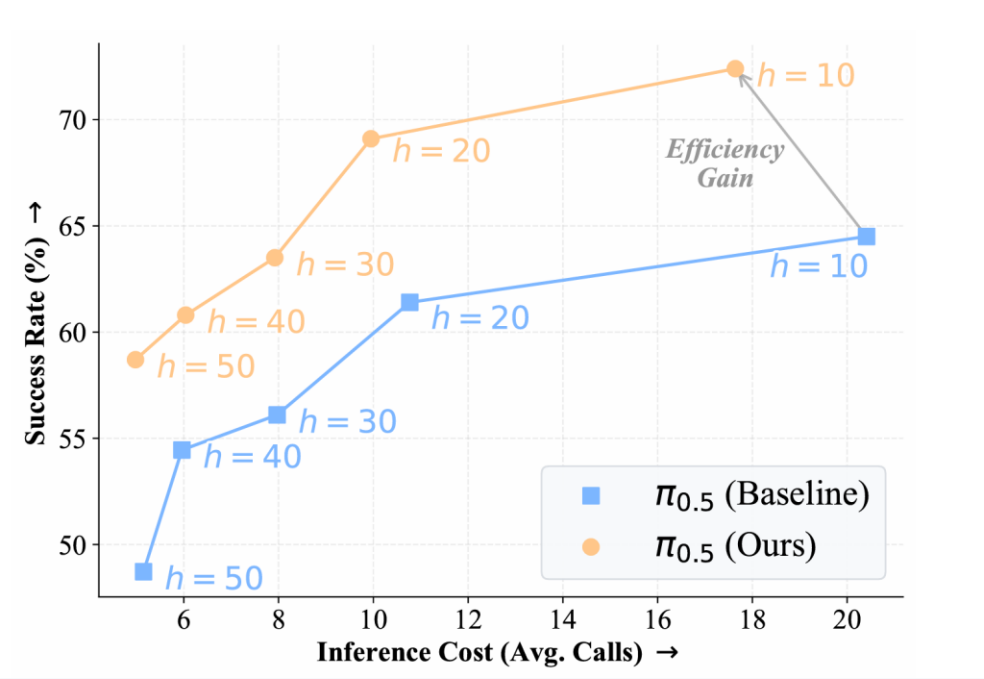
+17.7 pts

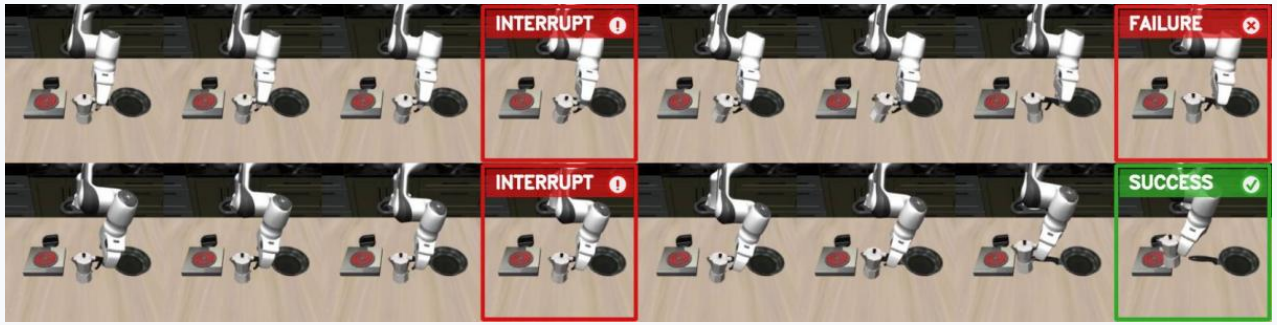
55.6% → 73.3% real-world success

Critical phases

83.7%

Truncations in manually labeled critical phases





Qualitative recovery: truncate stale chunk, replan with OGG, complete the task.

05 REAL ROBOTS

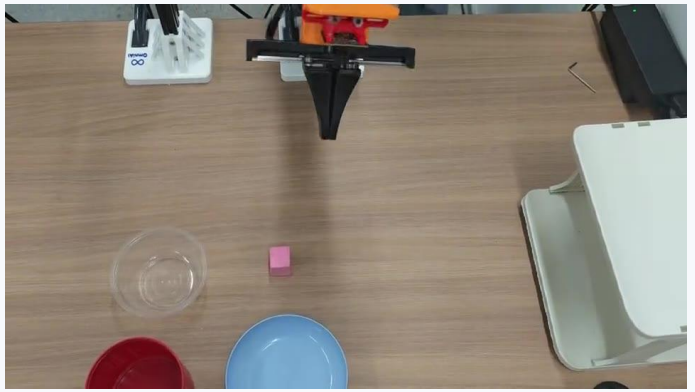
Perturbation demos are kept as silent project-page clips.

Drawer alignment

Human moves the drawer during execution.

Block to blue bowl

Target bowl is shifted mid-task.



Block to white bowl

Object/target perturbation during execution.

VLA-Corrector

Resources

Code	https://github.com/ZJU-OmniAI/vla-corrector
Project page	https://zju-omniai.github.io/vla-corrector/
Paper	https://arxiv.org/abs/2607.01804
Email	panyi0304@gmail.com、zhangwenqi@zju.edu.cn