World model pretraining and fine-tuning

31M+ sampled trajectories

AgiBot · Open-X · DROID · LIBERO

Pretraining

Prophet

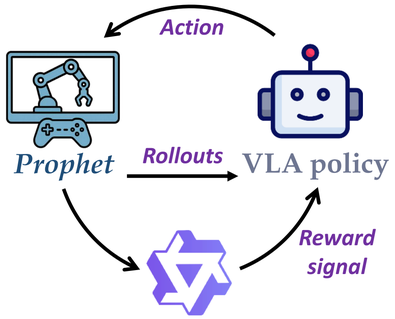
Fine-tuning

Limited new environment data

Generation

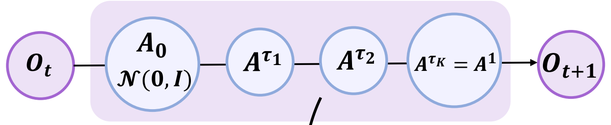
Fast generalization to new manipulations

Reinforcing VLA



Algorithm

Denoising process



Aggregate over k $\ell_{s,c,d} = \sum_{k=1}^{K} \log \pi_\theta^{(k)}(a_{s,c,d}|o_s)$

Per-d ratio

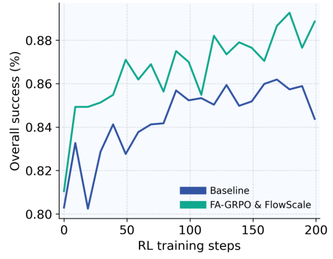
$$r_{s,c,d} = \exp(\ell_{s,c,d} - \ell_{s,c,d}^{old})$$

PPO-clip & sum over d

FA-GRPO

5× faster, 30% higher

FlowScale



$$A_{s,c,k}^{FS} = w_{s,k}\hat{A}_{s,c}$$

From noise schedule

$$\tilde{w}_{s,k} \propto (\sigma_{s,k}^2 + \varepsilon)^p,$$

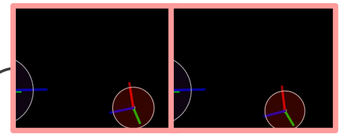
$$\frac{1}{K}\sum_{k=1}^{K} w_{s,k} = 1$$

Normalize-Mix-Clip; *stop gradient*

Prophet - A unified action-to-video robot actuation

Generated or real history frames

Conditional frame



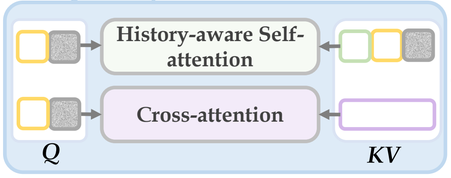
Future action frame (*Optional*)

E → **History latent packing** → *History latent*

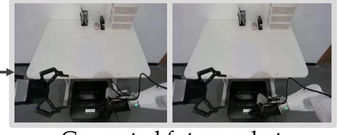
E → *Condition latent with noised future*

E → $c_3 = [p_3, e_3, g_3]$ $c_4 = [p_4, e_4, g_4]$ Future scalar action → *Action condition feature*

Diffusion Transformers (DiT)



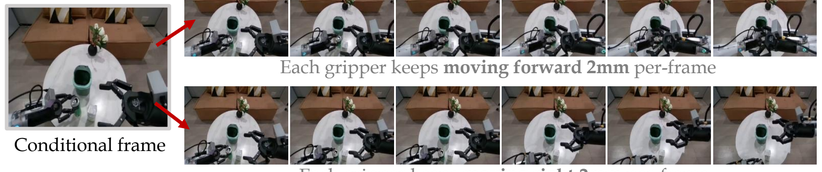
Denoised future latent → D →



Generated future robot manipulation

Generated rollouts

Precise robot control, by pretrained *Prophet*



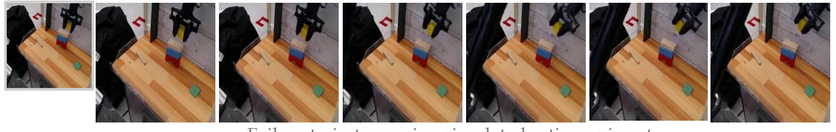
Conditional frame

Each gripper keeps **moving forward 2mm** per-frame

Each gripper keeps **moving right 2mm** per-frame

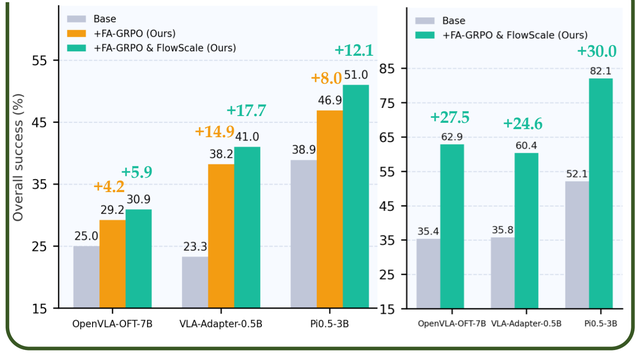
Fast generalization, by fine-tuned *Prophet* on 150 samples exclude "yellow cube" in task

Put the yellow cube on top of the blue cube (**unseen object in fine-tuning**)



Failure trajectory using simulated action as input

Real-world adaptation, by fine-tuned *Prophet* on 100 samples from self-collected data

Stack the bowls (**unseen trajectory in fine-tuning**)

Results on SimplerEnv & Real-robot