HINDSIGHT SELF-DISTILLATION: TEACHING ROBOTS TO ACT FROM THE PRESENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Robot demonstrations reveal the consequences of actions, but a deployed policy must act before those consequences are observed. How can hindsight shape what a policy sees before it acts? We introduce Hindsight Self-Distillation (HSD), which uses training-only future observations to shape the visual summaries used for action generation. A privileged teacher and a current-observation student share video and action experts, with each action branch receiving visual information exclusively through its own summaries. Ground-truth action supervision trains the teacher, which transfers both action-flow predictions and summary representations to the student. An interface-aware analysis separates teacher dependence on unavailable future evidence, information lost through the student interface, and student approximation error. At deployment, the student computes its visual summaries once and reuses their keys and values throughout action denoising. On LIBERO and RoboTwin 2.0, HSD achieves average success rates of 98.50% and 92.31%, respectively.

1 INTRODUCTION

A robot must choose an action before seeing its consequences. A demonstration, by contrast, records both the action and what followed. Subsequent observations reveal how an interaction unfolded, providing hindsight that is unavailable at the moment of action selection. This asymmetry creates a learning opportunity: future outcomes can teach a policy what to extract from the present. Figure 1 illustrates this principle, from demonstrated outcomes to hindsight-shaped visual summaries and current-observation control.

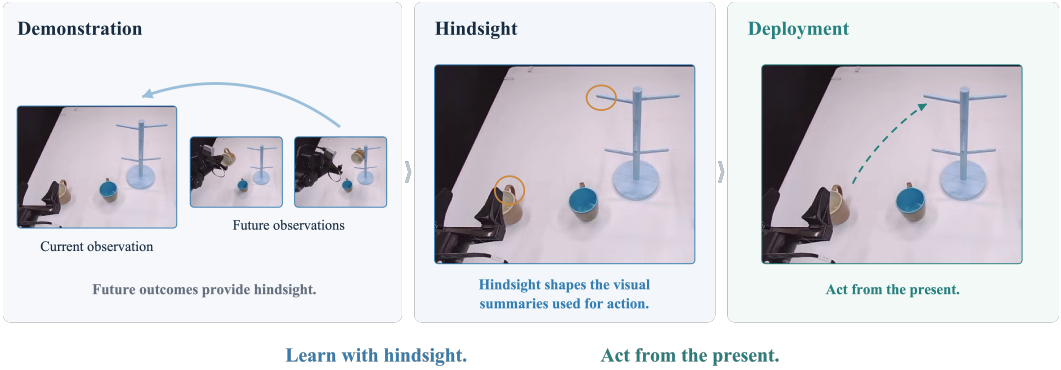


Figure 1: **Learn with hindsight, act from the present.** Future outcomes in demonstrations shape the visual summaries used for action generation. At deployment, HSD constructs these summaries from the current observation. Circles and arrows are conceptual annotations on demonstration frames.

Generalist policies transfer vision-language and robot pretraining to control (Zitkovich et al., 2023; Kim et al., 2025b; Ghosh et al., 2024); video-based policies exploit planning, predictive features, or joint video–action modeling (Du et al., 2023; Hu et al., 2025; Zhu et al., 2025). Fast-WAM removes deployment-time video generation while retaining video co-training (Yuan et al., 2026), and other

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.

methods transfer future-derived supervision into deployable policies (Zheng et al., 2025; Su et al., 2026; Luo et al., 2026b). This raises a sharper question: *how should hindsight shape the visual representation used for action generation?*

A privileged teacher observes one realized continuation, whereas the student must act from a present that may admit many. The transferable object should therefore not be the future itself, but an action-relevant representation of the present shaped by that future. This suggests a simple design principle: *hindsight should shape the visual interface used for action generation, rather than remain an auxiliary prediction target beside the policy*.

We introduce **Hindsight Self-Distillation (HSD)** around this principle. Teacher and student share the Video DiT and Action DiT weights but use separate query banks: teacher queries summarize current and future observations, while student queries summarize only the current observation. Each action branch receives visual information exclusively through its own summaries. Ground-truth actions train the teacher, which supervises the student's action-flow predictions and summaries through detached targets. The student also learns from demonstrated actions. At deployment, it constructs its visual interface once and reuses it throughout action denoising.

At a fixed parameter snapshot, we separate privileged-to-student distribution mismatch into unavailable future evidence, information lost through the visual interface, and approximation error. A squared-loss decomposition further identifies the component of privileged supervision predictable from deployment inputs.

Our contributions are threefold: **(i)** paired, action-supervised summaries that serve as the exclusive visual interfaces for privileged and deployable action generation; **(ii)** joint self-distillation through action-flow supervision and one-way summary alignment; and **(iii)** an interface-aware characterization of privileged transfer that separates unavailable future information from representation loss and student approximation error. We evaluate HSD on LIBERO, RoboTwin 2.0, and four real-world manipulation tasks, with ablations of the transfer objectives and summary-token count, together with inference-latency measurements.

2 Related Work

Generalist policies and generative control. RT-2 and OpenVLA transfer vision-language pretraining to control, while Octo learns from diverse robot demonstrations (Zitkovich et al., 2023; Kim et al., 2025b; Ghosh et al., 2024). Diffusion Policy and π_0 generate action chunks through conditional denoising and flow matching (Chi et al., 2023; Black et al., 2025b).

Video models for action generation. UniPi converts generated videos into actions; VPP extracts predictive video features (Du et al., 2023; Hu et al., 2025). Unified World Models couples video and action diffusion, whereas UVA supports decoupled decoding (Zhu et al., 2025; Li et al., 2025). DreamZero, Cosmos Policy, and LingBot-VA adapt video models for control and world modeling (Ye et al., 2026; Kim et al., 2026; Li et al., 2026). Fast-WAM retains video co-training but encodes current observations in one deployment-time pass (Yuan et al., 2026). HSD uses this deployment pattern and learns the visual summaries supplied to the action expert through hindsight supervision.

Future representations for policy learning. FLARE aligns diffusion-policy features with future-observation embeddings, while FRAPPE uses multiple future-representation targets in staged training (Zheng et al., 2025; Zhao et al., 2026). VLA-JEPA pretrains with latent future prediction; JEPA-VLA integrates pretrained video-predictive embeddings into VLA policies (Sun et al., 2026; Miao et al., 2026). LARA jointly optimizes latent-action and VLA models through alignment (Liu et al., 2026a). JEPA-WAM combines structured latent transitions with action generation, and JEPA Policy learns paired action and future-representation targets in a shared Transformer (Lin et al., 2026; Xu et al., 2026).

Privileged supervision and action prediction. World Guidance learns future conditions and freezes their encoder for deployable-policy training (Su et al., 2026). JALA aligns predictive action embeddings with inverse-dynamics representations; ForeTime-VLA distills offline future targets into a history-conditioned policy (Luo et al., 2026a; Ma et al., 2026b). DECOWAM distills an action-reconstructive bottleneck for residual context conditioning (Ma et al., 2026a); LeapBot-WA

uses asymmetric Anchor–Action DiTs and a static current-observation cache (Liu et al., 2026b). Being-H0.7 jointly trains current-context and future-informed action branches with multi-layer alignment and direct access to shared context (Luo et al., 2026b). Privileged Foresight Distillation transfers future-derived corrections through a residual output adapter (Fang et al., 2026). HSD couples action-flow transfer with alignment of the summaries that provide each branch's entire visual input.

Latent interfaces and self-distillation. Perceiver and BLIP-2 compress visual inputs through learnable queries (Jaegle et al., 2021; Li et al., 2023). Knowledge distillation, FitNets, and Be Your Own Teacher establish prediction transfer, feature supervision, and self-distillation (Hinton et al., 2015; Romero et al., 2015; Zhang et al., 2019); REPA aligns diffusion representations with pretrained visual features (Yu et al., 2025). HSD combines action-supervised queries with within-model transfer.

Privileged learning through a visual interface. Privileged learning and generalized distillation formalize training-only inputs (Vapnik & Vashist, 2009; Lopez-Paz et al., 2016). In partially observed control, teacher predictions may depend on unavailable evidence (Warrington et al., 2021). We use conditional-information and conditional-expectation identities (Cover & Thomas, 2006) to characterize transfer through HSD's visual interface at a fixed parameter snapshot.

3 METHOD

HSD trains the visual interface of a current-observation policy with demonstration hindsight. The Video DiT converts observations and learnable queries into visual summaries; the Action DiT uses those summaries to predict an action chunk. During training, the teacher sees current and future observations, and the student sees only the current observation. The streams share both experts but use separate query banks. Ground-truth actions train the teacher, whose action-flow predictions and summary states supervise the student. Deployment retains the student and caches its summaries across action-denoising steps (Figure 2).

3.1 SETUP

Each demonstration provides a deployment context $C = (o_t, s_t, \ell)$, future observations F, and an action chunk $A \in \mathbb{R}^{H \times d_a}$. Here o_t contains the current camera views, s_t is proprioception, ℓ is the instruction, H is the action horizon, and d_a is the action dimension. Figure 2 denotes F schematically by $o_{t+1:t+h}$, with observation horizon h distinct from H. The teacher receives $\widetilde{F}$, including future corruption and its conditioning metadata. Expectations use the training law $\mathbb{P}$, covering data sampling, noise, and conditioning augmentation.

We use Fast-WAM's separate video and action experts (Yuan et al., 2026), with the Wan2.2-TI2V-5B video backbone (Wan Team et al., 2025; Wan Team, 2025). Teacher and student share both experts; initialization and training settings appear in Section 4.1.

Action flow matching (Lipman et al., 2023) follows Fast-WAM's data-to-noise convention. For $\epsilon \sim \mathcal{N}(0, I)$ and $\tau \in [0, 1]$,

$$A_\tau = (1 - \tau)A + \tau\epsilon, \qquad U_\tau = \frac{\mathrm{d}A_\tau}{\mathrm{d}\tau} = \epsilon - A. \tag{1}$$

Both branches receive the same A_τ and τ, with $\tau = 0$ denoting data and $\tau = 1$ noise. The weighted flow error is

$$\ell_a(\widehat{U}, U; \tau) = \sum_{k=1}^{Hd_a} \Omega_k(\widehat{U}_k - U_k)^2, \qquad \Omega_k \geq 0, \tag{2}$$

where k indexes flattened action coordinates. The weights Ω_k include timestep weighting, padding masks, and loss normalization; they are shared by both branches and independent of the target selector below. Their dependence on other training variables is implicit.

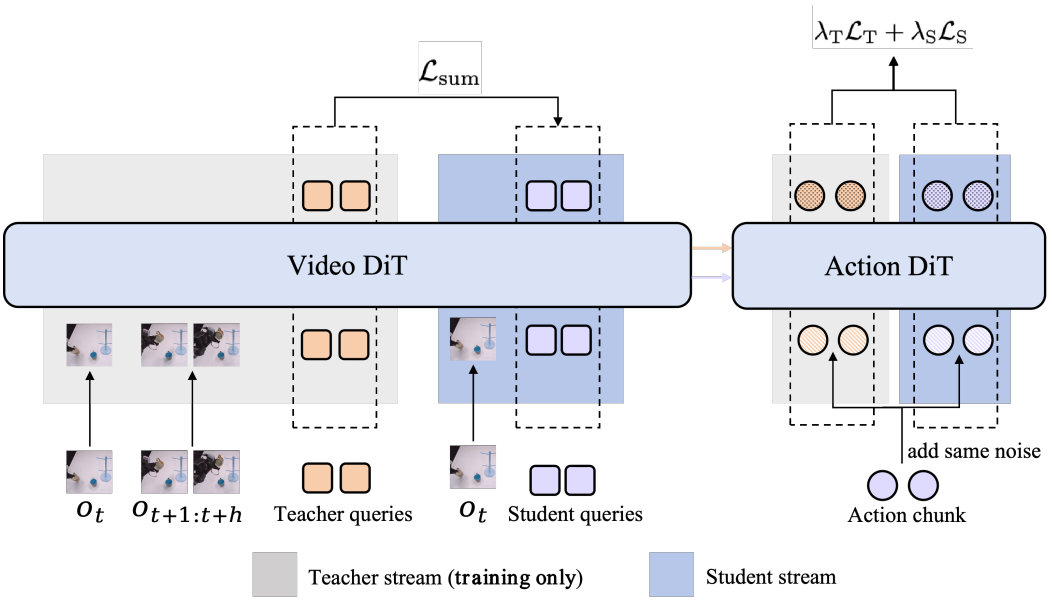


Figure 2: **Hindsight Self-Distillation.** Gray and blue backgrounds denote the training-only teacher and deployable student, which share the Video DiT and Action DiT weights. Teacher queries (orange) summarize current and future observations; student queries (purple) summarize the current observation. Branch-specific summary keys and values provide the sole visual input to each action stream. Both streams receive A_τ at the same flow time τ. $\mathcal{L}_T$ uses ground-truth flow; $\mathcal{L}_S$ selects ground-truth flow or a detached teacher prediction per chunk. $\mathcal{L}_{sum}$ aligns normalized terminal summaries. Encoders, nonvisual conditions, and auxiliary video co-training are omitted. Deployment retains only the student.

3.2 Paired visual summaries

Let z_0, z_f be current and privileged future visual tokens. Independently initialized query banks $Q^T, Q^S \in \mathbb{R}^{K\times D}$ each contain K learned tokens of width D. Define $Z_0^b = Q^b$ and Z_j^b as their hidden states after Video DiT layer j, for $j = 1, \dots, L$ and $b \in \{T, S\}$. Teacher summaries read z_0, z_f; student summaries read only z_0. Each group also attends within itself. Student summaries use fixed clean-current visual timestep conditioning, independent of the action chunk and action flow time.

Summary-only visual conditioning. At action layer j, the Video DiT projects the incoming states Z_{j-1}^b into keys and values (K_j^b, V_j^b) before updating them to Z_j^b. These form the complete visual interface

$$\mathcal{K}^b = \{(K_j^b, V_j^b)\}_{j=1}^L, \qquad b \in \{T, S\}. \tag{3}$$

The shared action expert predicts

$$\begin{aligned} \widehat{U}_\tau^T &= v_\theta(A_\tau, \tau, s_t, \ell; \mathcal{K}^T(C, \widetilde{F})), \\ \widehat{U}_\tau^S &= v_\theta(A_\tau, \tau, s_t, \ell; \mathcal{K}^S(C)). \end{aligned} \tag{4}$$

Here θ includes the shared expert weights and branch-specific query banks. Each action stream retains language, proprioception, and action self-attention, but accesses visual information only through its own summaries. Action supervision therefore trains this interface directly.

Information flow. Both branches run in one packed forward pass with the mask in Figure 6. Current tokens read only one another; future tokens read current and future tokens. Neither summary group reads actions or the other group. The current images repeated in Figure 2 represent a shared current-token group. Nonvisual conditioning contains no privileged inputs.

3.3 JOINT LEARNING AND TWO-LEVEL DISTILLATION

Learning the teacher. Ground-truth flow supervision trains the teacher's action predictor and visual summaries:

$$\mathcal{L}_{\mathrm{T}} = \mathbb{E}\big[\ell_a(\widehat{U}_\tau^{\mathrm{T}}, U_\tau; \tau)\big]. \tag{5}$$

Action-level transfer. For each example, sample $B \sim \mathrm{Bernoulli}(\rho)$ independently of all other training variables, including data, noise, flow time, future corruption, and loss weights. One selector chooses the target for the entire action chunk:

$$Y_{\tau,B} = (1-B)U_\tau + B\,\mathrm{sg}(\widehat{U}_\tau^{\mathrm{T}}), \qquad \mathcal{L}_{\mathrm{S}} = \mathbb{E}\big[\ell_a(\widehat{U}_\tau^{\mathrm{S}}, Y_{\tau,B}; \tau)\big], \tag{6}$$

where sg denotes stop-gradient. Sharing A_τ, τ ensures that both targets supervise the same denoising state. Averaging over B weights the ground-truth and detached-teacher errors by $1-\rho$ and ρ, respectively.

Summary-level transfer. We align the terminal states before the video prediction head:

$$\mathcal{L}_{\mathrm{sum}} = \frac{1}{KD}\mathbb{E}\left[\left\|\mathrm{LN}_0(Z_L^{\mathrm{S}}) - \mathrm{sg}\big(\mathrm{LN}_0(Z_L^{\mathrm{T}})\big)\right\|_F^2\right]. \tag{7}$$

Here LN_0 is non-affine LayerNorm over each token's features. Tokens are paired by index, without a learned projector or assignment. This loss aligns terminal representations, while action conditioning uses the layer-wise interface $\mathcal{K}^b$.

Overall objective. An auxiliary noisy-video stream retains the backbone flow loss $\mathcal{L}_{\mathrm{vid}}$, excluding the conditioned first frame. It has no forward path into student summaries. The total loss is

$$\mathcal{L} = \lambda_{\mathrm{vid}}\mathcal{L}_{\mathrm{vid}} + \lambda_{\mathrm{T}}\mathcal{L}_{\mathrm{T}} + \lambda_{\mathrm{S}}\mathcal{L}_{\mathrm{S}} + \lambda_{\mathrm{sum}}\mathcal{L}_{\mathrm{sum}}. \tag{8}$$

Teacher targets are detached in both transfer losses. Ground-truth teacher supervision updates its queries and the shared experts; student losses update student queries and shared experts. Thus each transfer gradient follows the student path, while shared-weight updates continue to change both branches.

3.4 DEPLOYMENT

At each replanning step, one Video DiT pass constructs $\mathcal{K}^{\mathrm{S}}(C)$ from the current observation and student queries. Fixed clean-current conditioning and independence from action tokens and flow time make this interface reusable while the action expert integrates

$$\frac{\mathrm{d}A_\tau}{\mathrm{d}\tau} = v_\theta(A_\tau, \tau, s_t, \ell; \mathcal{K}^{\mathrm{S}}(C)), \qquad A_1 \sim \mathcal{N}(0, I), \quad \tau : 1 \to 0. \tag{9}$$

Only the action expert runs at each denoising step. The cache is refreshed when C changes; teacher computation, future inputs, and auxiliary video denoising are removed.

3.5 WHAT HINDSIGHT TRANSFERS

Fix the model parameters and condition on the stochastic choices in the current-observation encoder; action-sampling noise remains random. All laws and losses below use this conditional setting, and the identities can be averaged over the encoder choices. The student conditions on the deterministic interface $G = g(C) = (s_t, \ell, \mathcal{K}^{\mathrm{S}}(C))$. Let $q_{\mathrm{T}}(a \mid C, \widetilde{F})$ denote the teacher's action distribution and define

$$\mathbb{Q}(C, \widetilde{F}, A^\dagger) = \mathbb{P}(C, \widetilde{F}) q_{\mathrm{T}}(A^\dagger \mid C, \widetilde{F}),$$

where $A^\dagger$ is a teacher-sampled action, distinct from the demonstration A.

Proposition 1 (Privileged transfer through a visual interface)**.** *For any student distribution $p_{\mathrm{S}}(a \mid G)$ with finite expected forward KL,*

$$\begin{aligned}\mathbb{E}_{C,\widetilde{F}} D_{\mathrm{KL}}\Big(q_{\mathrm{T}}(\cdot \mid C, \widetilde{F}) \,\|\, p_{\mathrm{S}}(\cdot \mid G)\Big) \\ = I_{\mathbb{Q}}(A^\dagger; \widetilde{F} \mid C) + I_{\mathbb{Q}}(A^\dagger; C \mid G) \\ + \mathbb{E}_G D_{\mathrm{KL}}(\mathbb{Q}(\cdot \mid G) \,\|\, p_{\mathrm{S}}(\cdot \mid G)).\end{aligned} \tag{10}$$

The three terms measure the teacher's dependence on unavailable future evidence, teacher-action information discarded by G, and student distribution mismatch, respectively (Cover & Thomas, 2006); Appendix A gives the proof. Over unrestricted conditional distributions, the optimum is $\mathbb{Q}(\cdot \mid G)$. With the full context C, it becomes $\bar{q}(a \mid C) = \mathbb{E}_{\widetilde{F}|C}[q_{\mathrm{T}}(a \mid C, \widetilde{F})]$, averaging over futures consistent with C.

For HSD's summary regression, set $R = \mathrm{LN}_0(Z_L^{\mathrm{T}})$, $r_{\mathrm{S}}(C) = \mathrm{LN}_0(Z_L^{\mathrm{S}})$, and $m(C) = \mathbb{E}_{\mathbb{P}}[R \mid C]$. For square-integrable targets,

$$KD\mathcal{L}_{\mathrm{sum}} = \mathbb{E}_{\mathbb{P}}\|R - m(C)\|_F^2 + \mathbb{E}_{\mathbb{P}}\|m(C) - r_{\mathrm{S}}(C)\|_F^2. \tag{11}$$

The first term is target variation unpredictable from C; the second measures approximation of the predictable mean. This mean is optimal over unrestricted square-integrable predictors, but a normalized student may not represent it exactly. For action-flow transfer, the regression context also includes A_τ and τ (Appendix B). The KL identity characterizes teacher-induced distributions under $\mathbb{Q}$, whereas HSD optimizes regression under $\mathbb{P}$.

4 Experiments

We evaluate the deployed current-observation student on LIBERO, RoboTwin 2.0, and four real-world tasks. We measure success and inference latency, and ablate the two transfer objectives and the summary-token count.

4.1 Experimental setup

Benchmarks and evaluation. LIBERO comprises Spatial, Object, Goal, and LIBERO-10 (Long) (Liu et al., 2023); we run 50 trials per task across 40 tasks. On RoboTwin 2.0 (Chen et al., 2025), we follow Fast-WAM's multi-task recipe with 2,500 clean-scene and 25,000 randomized-scene demonstrations, train on both conditions, and evaluate 100 trials per task and condition (Yuan et al., 2026). Scores average over tasks within each suite or condition; benchmark averages equally weight the four LIBERO suites or two RoboTwin conditions.

Training recipe. We use Fast-WAM's AdamW recipe: learning rate 10^{-4}, weight decay 0.01, cosine decay, mixed precision, and gradient clipping at 1.0 (Yuan et al., 2026). HSD trains for 5 epochs with global batch size 128 on LIBERO and 10 epochs with global batch size 512 on RoboTwin, using 5% warmup. LIBERO ablations share the data, sampling, and training duration.

Table 1: LIBERO success rates (%). PT marks additional embodied pretraining. DP, MDT, and π_0 scores follow Kim et al. (2025a); MDT uses 2% language annotations. $\pi_{0.5}$ scores follow Kim et al. (2026). Published averages retain source rounding. The lower block reports our experiments. Bold marks the highest displayed scores, including ties.

Method	PT	Spatial	Object	Goal	Long	Avg.
Previously reported results						
Diffusion Policy (Chi et al., 2023)	No	78.3	92.5	68.3	50.5	72.40
MDT (Reuss et al., 2024)	No	78.5	87.5	73.5	64.8	76.10
OpenVLA-OFT (Kim et al., 2025a)	No	94.3	95.2	91.7	86.5	91.90
π_0 (Black et al., 2025b)	Yes	96.8	98.8	95.8	85.2	94.20
$\pi_{0.5}$ (Black et al., 2025a)	Yes	98.8	98.2	98.0	92.4	96.90
OpenVLA-OFT (Kim et al., 2025a)	Yes	97.6	98.4	97.9	94.5	97.10
Cosmos Policy (Kim et al., 2026)	No	98.1	**100.0**	98.2	97.6	**98.50**
Fast-WAM (Yuan et al., 2026)	No	98.2	**100.0**	97.0	95.2	97.60
Fast-WAM-Joint (Yuan et al., 2026)	No	**99.6**	99.4	98.2	96.8	**98.50**
Fast-WAM-IDM (Yuan et al., 2026)	No	98.8	97.8	97.8	97.6	98.00
Experiments in this work						
Fast-WAM (Yuan et al., 2026)	No	97.0	99.4	96.6	94.8	96.95
HSD (ours)	No	98.6	99.2	**98.4**	**97.8**	**98.50**

Model and observations. The video backbone is Wan2.2-TI2V-5B, with hidden width $D = 3072$. The action expert has hidden width 1024, and both experts use $L = 30$ layers. The action expert follows the Fast-WAM initialization procedure, with action horizon $H = 32$ and $K = 32$ summary tokens per branch. LIBERO uses two 224×224 camera images concatenated horizontally; RoboTwin 2.0 uses its benchmark-specific multi-camera interface. The teacher-target probability is $\rho = 0.5$, all four objective weights are one, and the future-conditioning corruption probability is 0.1. Deployment uses one Video DiT forward pass and ten action-denoising steps.

Comparison methods. Tables 1 and 2 separate published results from our evaluations. PT marks robot or human manipulation pretraining beyond the target benchmark, excluding generic visual, video, and vision-language initialization. Our Fast-WAM reproduction is the primary reference.

4.2 Benchmark results

LIBERO. HSD reaches 98.50% average success, exceeding our Fast-WAM reproduction by 1.55 percentage points (Table 1). Long improves most, from 94.8% to 97.8%. HSD matches the reported average of Cosmos Policy and Fast-WAM-Joint using current-observation summaries without deployment-time video denoising.

Table 2: RoboTwin 2.0 success rates (%) under clean-and-randomized co-training. PT marks additional embodied pretraining. Motus and LingBot-VA scores follow their CVPR and RSS versions, respectively. Avg. is the mean of the two displayed conditions. Bold marks column maxima.

Method	PT	Clean	Randomized	Avg.
Previously reported results				
π_0 (Black et al., 2025b)	Yes	65.92	58.40	62.16
$\pi_{0.5}$ (Black et al., 2025a)	Yes	82.74	76.76	79.75
Motus (Bi et al., 2026)	No	72.80	77.00	74.90
Motus (Bi et al., 2026)	Yes	88.66	87.02	87.84
LingBot-VA (Li et al., 2026)	Yes	91.99	91.11	91.55
Fast-WAM (Yuan et al., 2026)	No	91.88	91.78	91.83
Fast-WAM-Joint (Yuan et al., 2026)	No	90.84	90.32	90.58
Fast-WAM-IDM (Yuan et al., 2026)	No	91.16	91.34	91.25
Our evaluation				
HSD (ours)	No	**92.48**	**92.14**	**92.31**

RoboTwin 2.0. HSD reaches 92.48% in Clean and 92.14% in Randomized scenes, averaging 92.31% (Table 2). These exceed the published Fast-WAM scores by 0.60 and 0.36 percentage points, respectively.

4.3 Real-world experiments

We evaluate four bimanual YAM tasks: tube insertion into a rack, slipper placement on a shoe rack, color sorting (gray blocks into the left basket, pink blocks into the right), and pen placement in a container (Figure 3). For each task, we use 50 demonstrations recording an external view, two wrist views, and both arms' joint and gripper states.

Across all four tasks, both methods train for 20,000 steps with a learning rate of 10^{-4} and a global batch size of 128, and undergo 20 evaluation trials per task. A trial succeeds if its final state meets the task goal, such as inserting the tube into the rack. HSD averages 40.00% success versus Fast-WAM's 28.75%, a gain of 11.25 percentage points (Figure 4). Color sorting improves from 45% to 65% and pen placement from 30% to 50%; tube insertion improves from 10% to 15%, while slipper placement remains at 30%.

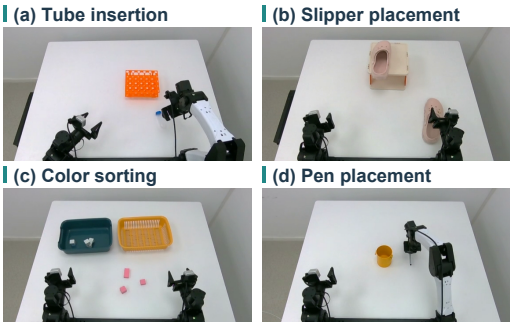


Figure 3: **Four real-world YAM tasks.** Demonstration frames show (a) tube insertion, (b) slipper placement, (c) color sorting, and (d) pen placement.

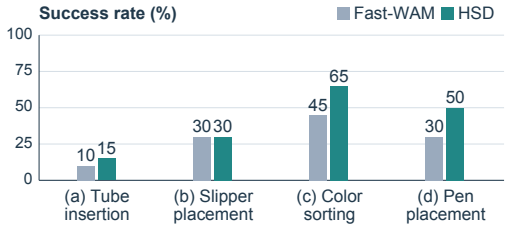


Figure 4: **Real-world success rates.** Each method uses 20,000 training steps and 20 evaluation trials per task. Average success is 28.75% for Fast-WAM and 40.00% for HSD.

4.4 Component ablations

Table 3: Two-level transfer ablation on LIBERO. Numerical results are success rates (%). The teacher, paired summary interfaces, and video co-training are retained in all rows. "No explicit transfer" disables both distillation objectives but retains shared-parameter learning.

Variant	Action transfer	Summary transfer	Spatial	Object	Goal	Long	Avg.
No explicit transfer	No	No	**98.6**	**99.4**	97.6	93.8	97.35
Action only	Yes	No	98.0	99.2	97.8	95.0	97.50
Summary only	No	Yes	**98.6**	**99.4**	97.0	93.4	97.10
HSD	Yes	Yes	**98.6**	99.2	**98.4**	**97.8**	**98.50**

Action-level and summary-level transfer. We independently ablate action-level and summary-level transfer (Table 3). Action transfer uses teacher targets with probability $\rho = 0.5$ when enabled, and summary transfer uses weight $\lambda_{\text{sum}} = 1$; disabling either sets its parameter to zero. Teacher supervision, paired summaries, and video co-training are retained throughout.

Joint transfer achieves 98.50% average success, compared with 97.35% without explicit transfer, 97.50% with action transfer alone, and 97.10% with summary transfer alone. Relative to no explicit transfer, HSD gains 1.15 percentage points overall and 4.0 points on Long.

Table 4: Summary-token count on LIBERO. Both branches use K tokens. Entries are success rates (%); bold marks the highest score in each column.

K	Spatial	Object	Goal	Long	Avg.
8	97.6	98.8	**98.8**	96.4	97.90
16	98.2	**100.0**	97.8	94.8	97.70
32	**98.6**	99.2	98.4	**97.8**	**98.50**

Number of summary tokens. Table 4 compares $K \in \{8, 16, 32\}$ summary tokens per branch. The corresponding average success rates are 97.90%, 97.70%, and 98.50%. The effect is not monotonic: $K = 8$ performs best on Goal, while $K = 16$ reaches 100.0% on Object. We use $K = 32$ for its highest overall average and Long success of 97.8%, compared with 96.4% for $K = 8$ and 94.8% for $K = 16$.

4.5 Inference latency

We measure mean latency per action chunk on one NVIDIA H20 with BF16, batch size 1, and cached language conditioning, averaging 100 runs after 30 warm-up runs. HSD takes 303.0 ms, close to Fast-WAM's 307.5 ms (Figure 5). By caching its visual summaries across ten action-denoising steps, HSD retains Fast-WAM-like latency while improving task success.

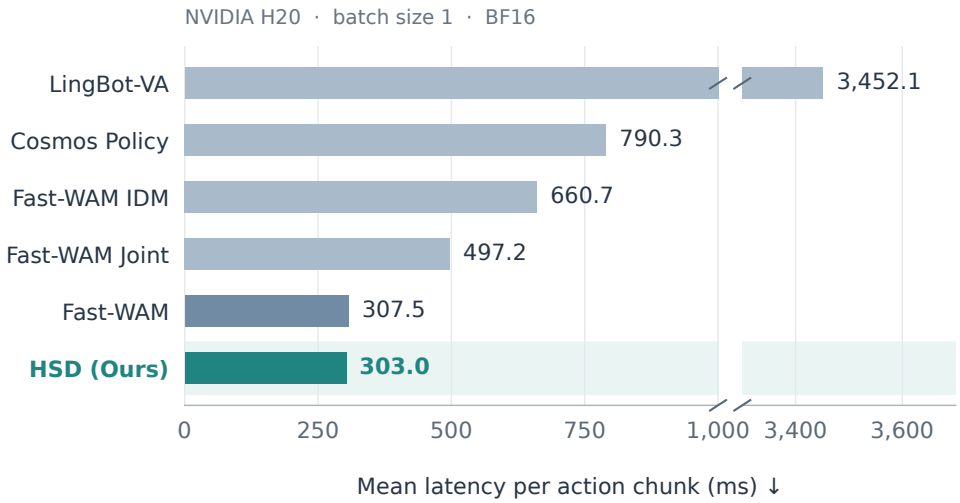


Figure 5: **Inference latency.** Mean milliseconds per action chunk on one NVIDIA H20 (BF16, batch size 1), using model-specific inference settings. Lower is better.

5 Conclusion

HSD learns with hindsight and acts from the present. An action-supervised teacher transfers both flow predictions and visual summaries; the student computes its current-observation interface once and reuses it across action denoising. HSD reaches 98.50% on LIBERO and 92.31% on RoboTwin 2.0, improves real-world success from 28.75% to 40.00%, and retains Fast-WAM-like inference latency. Fixed-snapshot decompositions characterize the targets available for transfer.

Limitations. Hindsight transfer depends on which teacher targets are predictable from the current context. The analysis holds at a fixed snapshot, while closed-loop gains are assessed empirically. Real-world evaluation uses 20 trials per task.

AI USE STATEMENT

We used AI tools to assist with manuscript writing and revision, including language polishing, improving clarity and organization, and checking LaTeX formatting and bibliographic metadata. The authors are responsible for the final text, claims, and reported results.

REPRODUCIBILITY STATEMENT

Sections 3 and 4.1 describe the model, training objectives, and experimental setup. Appendices A and B give the assumptions and proofs of the theoretical identities. Appendices C and D describe attention, caching, training settings, evaluation, and latency measurement.

REFERENCES

Hongzhe Bi, Hengkai Tan, Shenghao Xie, Zeyuan Wang, Shuhe Huang, Haitian Liu, Ruowen Zhao, Yao Feng, Chendong Xiang, Yinze Rong, Hongyan Zhao, Hanyu Liu, Zhizhong Su, Lei Ma, Hang Su, and Jun Zhu. Motus: A Unified Latent Action World Model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 35101–35113, June 2026. doi: 10.48550/arXiv.2512.13030. URL https://openaccess.thecvf.com/content/CVPR2026/html/Bi_Motus_A_Unified_Latent_Action_World_Model_CVPR_2026_paper.html.

Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: A Vision-Language-Action Model with Open-World Generalization. In *Proceedings of The 9th Conference on Robot Learning*, volume 305, pp. 17–40. PMLR, 2025a. doi: 10.48550/arXiv.2504.16054. URL https://proceedings.mlr.press/v305/black25a.html.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, Laura Smith, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A Vision-Language-Action Flow Model for General Robot Control. In *Proceedings of Robotics: Science and Systems*, 2025b. doi: 10.15607/RSS.2025.XXI.010. URL https://www.roboticsproceedings.org/rss21/p010.html.

Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu Gu, Weiliang Deng, Yubin Guo, Tian Nian, Xuanbing Xie, Qiangyu Chen, Kailun Su, Tianling Xu, Guodong Liu, Mengkang Hu, Huan-ang Gao, Kaixuan Wang, Zhixuan Liang, Yusen Qin, Xiaokang Yang, Ping Luo, and Yao Mu. RoboTwin 2.0: A Scalable Data Generator and Benchmark with Strong Domain Randomization for Robust Bimanual Robotic Manipulation, 2025. URL https://arxiv.org/abs/2506.18088.

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin CM Burchfiel, and Shuran Song. Diffusion Policy: Visuomotor Policy Learning via Action Diffusion. In *Proceedings of Robotics: Science and Systems*, 2023. doi: 10.15607/RSS.2023.XIX.026. URL https://www.roboticsproceedings.org/rss19/p026.html.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 2 edition, 2006. doi: 10.1002/047174882X. URL https://onlinelibrary.wiley.com/doi/book/10.1002/047174882X.

Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Josh Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning Universal Policies via Text-Guided Video Generation. In *Advances in Neural Information Processing Systems*, volume 36, pp. 9156–9172, 2023. doi: 10.52202/075280-0403. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1d5b9233ad716a43be5c0d3023cb82d0-Abstract-Conference.html.

Pengcheng Fang, Hongli Chen, and Xiaohao Cai. Privileged Foresight Distillation: Zero-Cost Future Correction for World Action Models, 2026. URL https://arxiv.org/abs/2604.25859.

Dibya Ghosh, Homer Rich Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, You Liang Tan, Lawrence Yunliang Chen, Quan Vuong, Ted Xiao, Pannag R Sanketi, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An Open-Source Generalist Robot Policy. In *Proceedings of Robotics: Science and Systems*, 2024. doi: 10.15607/RSS.2024.XX.090. URL https://www.roboticsproceedings.org/rss20/p090.html.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the Knowledge in a Neural Network, 2015. URL https://arxiv.org/abs/1503.02531.

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. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 24328–24346. PMLR, 2025. doi: 10.48550/arXiv.2412.14803. URL https://proceedings.mlr.press/v267/hu25g.html.

Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General Perception with Iterative Attention. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 4651–4664. PMLR, 2021. doi: 10.48550/arXiv.2103.03206. URL https://proceedings.mlr.press/v139/jaegle21a.html.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-Tuning Vision-Language-Action Models: Optimizing Speed and Success. In *Proceedings of Robotics: Science and Systems*, 2025a. doi: 10.15607/RSS.2025.XXI.017. URL https://www.roboticsproceedings.org/rss21/p017.html.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An Open-Source Vision-Language-Action Model. In *Proceedings of The 8th Conference on Robot Learning*, volume 270, pp. 2679–2713. PMLR, 2025b. doi: 10.48550/arXiv.2406.09246. URL https://proceedings.mlr.press/v270/kim25c.html.

Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, and Jinwei Gu. Cosmos Policy: Fine-Tuning Video Models for Visuomotor Control and Planning, 2026. URL https://arxiv.org/abs/2601.16163.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 19730–19742. PMLR, 2023. doi: 10.48550/arXiv.2301.12597. URL https://proceedings.mlr.press/v202/li23q.html.

Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Luyao Zhang, Mingrui Yu, Zelin Gao, Nan Xue, Boyu Zhou, Xing Zhu, Mingyu Ding, Yujun Shen, and Yinghao Xu. Causal World Modeling for Robot Control. In *Proceedings of Robotics: Science and Systems*, 2026. doi: 10.15607/RSS.2026.XXII.016. URL https://www.roboticsproceedings.org/rss22/p016.html.

Shuang Li, Yihuai Gao, Dorsa Sadigh, and Shuran Song. Unified Video Action Model. In *Proceedings of Robotics: Science and Systems*, 2025. doi: 10.15607/RSS.2025.XXI.074. URL https://www.roboticsproceedings.org/rss21/p074.html.

Yihan Lin, Jiawei He, Shifeng Bao, Chen Zhao, Yang Li, Xiaobo Wang, Yan Wang, Cheng Chi, and Jing Zhang. JEPA-WAM: Learning Vision-Language-Action Policies with Joint-Embedding World Modeling, 2026. URL https://arxiv.org/abs/2608.09381.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow Matching for Generative Modeling. In *International Conference on Learning Representations*, 2023. doi: 10.48550/arXiv.2210.02747. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking Knowledge Transfer for Lifelong Robot Learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 44776–44791, 2023. doi: 10.52202/075280-1939. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/8c3c666820ea055a77726d66fc7d447f-Abstract-Datasets_and_Benchmarks.html.

Mengya Liu, Baoxiong Jia, Jiangyong Huang, Jingze Zhang, and Siyuan Huang. LARA: Latent Action Representation Alignment for Vision-Language-Action Models. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026a. doi: 10.48550/arXiv.2606.07100. URL https://icml.cc/virtual/2026/poster/61232.

Pei Liu, Nan Zheng, Lang Zhang, Daojie Peng, Yanan Zhang, Feilong Kong, Mingyue Feng, Jiachao Liu, Yaonong Wang, Qifeng Chen, and Jun Ma. LeapBot-WA: World-Anchor Action Models via Predictive Latent Alignments, 2026b. URL https://arxiv.org/abs/2607.23969.

David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying distillation and privileged information. In *International Conference on Learning Representations*, 2016. doi: 10.48550/arXiv.1511.03643. URL https://arxiv.org/abs/1511.03643.

Hao Luo, Ye Wang, Wanpeng Zhang, Haoqi Yuan, Yicheng Feng, Haiweng Xu, Sipeng Zheng, and Zongqing Lu. Joint-Aligned Latent Action: Towards Scalable VLA Pretraining in the Wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 35047–35058, 2026a. doi: 10.48550/arXiv.2602.21736. URL https://openaccess.thecvf.com/content/CVPR2026/html/Luo_Joint-Aligned_Latent_Action_Towards_Scalable_VLA_Pretraining_in_the_Wild_CVPR_2026_paper.html.

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, 2026b. URL https://arxiv.org/abs/2605.00078.

Siyuan Ma, Boshi Zhang, Yutian Zhang, Qinglian Wu, Jiaqi Zhai, Dong Wei, and Qiaojun Yu. DECOWAM: Decoupled Whole-Body World-Action Model for Legged Mobile Manipulation, 2026a. URL https://arxiv.org/abs/2608.20114.

Siyuan Ma, Yutian Zhang, Boshi Zhang, Qinglian Wu, Jiaqi Zhai, Dong Wei, and Xiaojin Huang. ForeTime-VLA: Causal Future-Token Distillation from a World Action Model for Conveyor-Belt Manipulation, 2026b. URL https://arxiv.org/abs/2608.20735.

Shangchen Miao, Ningya Feng, Jialong Wu, Ye Lin, Xu He, Dong Li, and Mingsheng Long. JEPA-VLA: Video Predictive Embedding is Needed for VLA Models, 2026. URL https://arxiv.org/abs/2602.11832.

Moritz Reuss, Ömer Erdinç Yağmurlu, Fabian Wenzel, and Rudolf Lioutikov. Multimodal Diffusion Transformer: Learning Versatile Behavior from Multimodal Goals. In *Proceedings of Robotics: Science and Systems*, 2024. doi: 10.15607/RSS.2024.XX.121. URL https://www.roboticsproceedings.org/rss20/p121.html.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. FitNets: Hints for Thin Deep Nets. In *International Conference on Learning Representations*, 2015. doi: 10.48550/arXiv.1412.6550. URL https://arxiv.org/abs/1412.6550.

Stephane Ross, Geoffrey Gordon, and Drew Bagnell. A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15, pp. 627–635. PMLR, 2011. doi: 10.48550/arXiv.1011.0686. URL https://proceedings.mlr.press/v15/ross11a.html.

Yue Su, Sijin Chen, Haixin Shi, Mingyu Liu, Zhengshen Zhang, Ningyuan Huang, Weiheng Zhong, Zhengbang Zhu, Yuxiao Liu, and Xihui Liu. World Guidance: World Modeling in Condition Space for Action Generation, 2026. URL https://arxiv.org/abs/2602.22010.

Jingwen Sun, Wenyao Zhang, Zekun Qi, Shaojie Ren, Zezhi Liu, Hanxin Zhu, Guangzhong Sun, Xin Jin, and Zhibo Chen. VLA-JEPA: Enhancing Vision-Language-Action Model with Latent World Model. In *Computer Vision – ECCV 2026*, volume 17064, pp. 478–497, 2026. doi: 10.1007/978-3-032-37196-6_26.

Vladimir Vapnik and Akshay Vashist. A new learning paradigm: Learning using privileged information. *Neural Networks*, 22(5–6):544–557, 2009. doi: 10.1016/j.neunet.2009.06.042. URL https://doi.org/10.1016/j.neunet.2009.06.042.

Wan Team. Wan2.2-TI2V-5B, 2025. URL https://huggingface.co/Wan-AI/Wan2.2-TI2V-5B.

Wan Team, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and Advanced Large-Scale Video Generative Models, 2025. URL https://arxiv.org/abs/2503.20314.

Andrew Warrington, Jonathan W Lavington, Adam Scibior, Mark Schmidt, and Frank Wood. Robust Asymmetric Learning in POMDPs. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 11013–11023. PMLR, 2021. doi: 10.48550/arXiv.2012.15566. URL https://proceedings.mlr.press/v139/warrington21a.html.

Jie Xu, Kangjin Yu, Ziyi Jin, Junjie Gao, Liqing Chen, Yixian Li, Shuai Tian, and Zhongpu Xia. JEPA Policy: Diffusion-Free Imitation Learning via Paired Action and Future Representation Prediction, 2026. URL https://arxiv.org/abs/2609.09630.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi "Jim" Fan, and Joel Jang. World Action Models are Zero-shot Policies, 2026. URL https://arxiv.org/abs/2602.15922.

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation Alignment for Generation: Training Diffusion Transformers Is Easier Than You Think. In *International Conference on Learning Representations*, pp. 87400–87442, 2025. doi: 10.48550/arXiv.2410.06940.

Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-WAM: Do World Action Models Need Test-time Future Imagination?, 2026. URL https://arxiv.org/abs/2603.16666.

Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Be Your Own Teacher: Improve the Performance of Convolutional Neural Networks via Self Distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3712–3721, 2019. doi: 10.1109/ICCV.2019.00381. URL https://www.computer.org/csdl/proceedings-article/iccv/2019/480300d712/1hVlMcF0InK.

Han Zhao, Jingbo Wang, Wenxuan Song, Shuai Chen, Yang Liu, Yan Wang, Haoang Li, and Donglin Wang. FRAPPE: Infusing World Modeling into Generalist Policies via Multiple Future Representation Alignment, 2026.

Ruijie Zheng, Jing Wang, Scott Reed, Johan Bjorck, Yu Fang, Fengyuan Hu, Joel Jang, Kaushil Kundalia, Zongyu Lin, Loïc Magne, Avnish Narayan, You Liang Tan, Guanzhi Wang, Qi Wang, Jiannan Xiang, Yinzhen Xu, Seonghyeon Ye, Jan Kautz, Furong Huang, Yuke Zhu, and Linxi Fan. FLARE: Robot Learning with Implicit World Modeling. In *Proceedings of The 9th Conference on Robot Learning*, volume 305, pp. 3952–3971. PMLR, 2025. doi: 10.48550/arXiv.2505.15659. URL https://proceedings.mlr.press/v305/zheng25a.html.

Chuning Zhu, Raymond Yu, Siyuan Feng, Benjamin Burchfiel, Paarth Shah, and Abhishek Gupta. Unified World Models: Coupling Video and Action Diffusion for Pretraining on Large Robotic Datasets. In *Proceedings of Robotics: Science and Systems*, 2025. doi: 10.15607/RSS.2025.XXI.015. URL https://www.roboticsproceedings.org/rss21/p015.html.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, Quan Vuong, Vincent Vanhoucke, Huong Tran, Radu Soricut, Anikait Singh, Jaspiar Singh, Pierre Sermanet, Pannag R. Sanketi, Grecia Salazar, Michael S. Ryoo, Krista Reymann, Kanishka Rao, Karl Pertsch, Igor Mordatch, Henryk Michalewski, Yao Lu, Sergey Levine, Lisa Lee, Tsang-Wei Edward Lee, Isabel Leal, Yuheng Kuang, Dmitry Kalashnikov, Ryan Julian, Nikhil J. Joshi, Alex Irpan, Brian Ichter, Jasmine Hsu, Alexander Herzog, Karol Hausman, Keerthana Gopalakrishnan, Chuyuan Fu, Pete Florence, Chelsea Finn, Kumar Avinava Dubey, Danny Driess, Tianli Ding, Krzysztof Marcin Choromanski, Xi Chen, Yevgen Chebotar, Justice Carbajal, Noah Brown, Anthony Brohan, Montserrat Gonzalez Arenas, and Kehang Han. RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control. In *Proceedings of The 7th Conference on Robot Learning*, volume 229, pp. 2165–2183. PMLR, 2023. doi: 10.48550/arXiv.2307.15818. URL https://proceedings.mlr.press/v229/zitkovich23a.html.

A Analysis of privileged transfer

A.1 Conditional-information decomposition

At a fixed parameter snapshot, all conditionals in this proof are under $\mathbb{Q}$ unless otherwise indicated. We assume regular conditional distributions and the finite expected KL in Proposition 1. Since $G = g(C)$,

$$\log \frac{q_{\mathrm{T}}(A^\dagger \mid C, \widetilde{F})}{p_{\mathrm{S}}(A^\dagger \mid G)} = \log \frac{q_{\mathrm{T}}(A^\dagger \mid C, \widetilde{F})}{\mathbb{Q}(A^\dagger \mid C)} + \log \frac{\mathbb{Q}(A^\dagger \mid C)}{\mathbb{Q}(A^\dagger \mid G)} + \log \frac{\mathbb{Q}(A^\dagger \mid G)}{p_{\mathrm{S}}(A^\dagger \mid G)}.$$

Taking expectation under $\mathbb{Q}$ yields, respectively, $I_{\mathbb{Q}}(A^\dagger; \widetilde{F} \mid C)$, $I_{\mathbb{Q}}(A^\dagger; C \mid G)$, and $\mathbb{E}_G D_{\mathrm{KL}}(\mathbb{Q}(\cdot \mid G) \| p_{\mathrm{S}}(\cdot \mid G))$. The conditional-independence identity $\mathbb{Q}(A^\dagger \mid C, G) = \mathbb{Q}(A^\dagger \mid C)$ follows because G is deterministic given C. This proves Equation (10). With an unrestricted distribution family, the final term is minimized by $p_{\mathrm{S}}(\cdot \mid G) = \mathbb{Q}(\cdot \mid G)$; replacing G with C gives $\bar{q}(\cdot \mid C)$.

A.2 Squared-loss summary decomposition

For $m(C) = \mathbb{E}_{\mathbb{P}}[R \mid C]$,

$$R - r_{\mathrm{S}}(C) = (R - m(C)) + (m(C) - r_{\mathrm{S}}(C)).$$

The expected cross term vanishes by conditional expectation:

$$\mathbb{E}_{\mathbb{P}} \langle R - m(C), m(C) - r_{\mathrm{S}}(C) \rangle_F = 0.$$

Expanding the square proves Equation (11). Detaching R changes differentiation, not the value of this identity. The conditional mean is the optimum over unrestricted square-integrable predictors. A parameterized, normalized summary stream can have nonzero approximation error because the conditional mean need not lie in its realizable set.

A.3 Scope of the analysis

The information decomposition uses the teacher-induced action law $\mathbb{Q}$, whereas the regression decompositions use the training law $\mathbb{P}$. All results hold at a fixed parameter snapshot, conditional on current-encoder randomness as in Section 3.5; averaging both sides over those choices preserves the equalities. They characterize distributions and regression targets, not the shared model's optimization trajectory. Closed-loop success can also depend on policy-induced state-distribution shift (Ross et al., 2011). The token count specifies an architectural interface size; it is not a Shannon information-capacity bound on real-valued representations.

B Action-flow regression through the summary interface

Let T_τ denote the numerical value of the detached teacher velocity target at a fixed snapshot, and define

$$X = (C, A_\tau, \tau), \qquad X_G = (G, A_\tau, \tau), \qquad m_X = \mathbb{E}_{\mathbb{P}}[T_\tau \mid X], \quad m_G = \mathbb{E}_{\mathbb{P}}[T_\tau \mid X_G].$$

For the unweighted squared loss and any square-integrable student predictor $v_{\mathrm{S}}(X_G)$,

$$\mathbb{E}\|T_\tau - v_{\mathrm{S}}(X_G)\|_F^2 = \mathbb{E}\|T_\tau - m_X\|_F^2 + \mathbb{E}\|m_X - m_G\|_F^2 + \mathbb{E}\|m_G - v_{\mathrm{S}}(X_G)\|_F^2. \tag{12}$$

Because X_G is a deterministic function of X, the tower property gives $\mathbb{E}[m_X \mid X_G] = m_G$. The three summands in $T_\tau - v_{\mathrm{S}} = (T_\tau - m_X) + (m_X - m_G) + (m_G - v_{\mathrm{S}})$ are therefore pairwise orthogonal in L^2. The terms describe target variation unexplained by the full regression context, predictable target variation discarded by the interface, and student approximation error within that interface.

Timestep weighting and padding. Using the coordinate weights Ω_k defined in Equation (2), assume $0 < \mathbb{E}_{\mathbb{P}}\Omega_k < \infty$ and finite weighted second moments for the target and predictor. Ordinary square integrability under $\mathbb{P}$ alone is not sufficient if the weights are unbounded. Define the tilted law

$$\mathrm{d}\mathbb{P}_k^{\Omega} = \frac{\Omega_k}{\mathbb{E}_{\mathbb{P}}\Omega_k}\,\mathrm{d}\mathbb{P}.$$

Apply the scalar version of Equation (12) under $\mathbb{P}_k^{\Omega}$, multiply by $\mathbb{E}_{\mathbb{P}}\Omega_k$, and sum over k. Coordinates with zero expected weight contribute zero. This gives the weighted decomposition without assuming that the padding mask is measurable from deployment information. Its projection targets are weighted conditional means, such as

$$m_{G,k}^{\Omega} = \frac{\mathbb{E}_{\mathbb{P}}[\Omega_k T_{\tau,k} \mid X_G]}{\mathbb{E}_{\mathbb{P}}[\Omega_k \mid X_G]},$$

where the denominator is positive. Values on zero-weight events are immaterial.

Mixture target. For the Bernoulli selector independent of all other training variables, the coordinate-wise optimum of the mixed squared loss is

$$v_k^*(X_G) = \frac{(1-\rho)\mathbb{E}[\Omega_k U_{\tau,k} \mid X_G] + \rho\mathbb{E}[\Omega_k T_{\tau,k} \mid X_G]}{\mathbb{E}[\Omega_k \mid X_G]}.$$

For unweighted error this is the corresponding mixture of ordinary conditional means. Thus teacher bias can also affect the regression target; direct action supervision of the teacher grounds the target-learning process in demonstrations.

Random selection and a blended target. Let $\bar{Y}_\tau = (1-\rho)U_\tau + \rho T_\tau$. Conditional on the numerical target values and the selector-independent coordinate weights,

$$\mathbb{E}_B \ell_a(\widehat{U}, Y_{\tau,B}; \tau) = \ell_a(\widehat{U}, \bar{Y}_\tau; \tau) + \rho(1-\rho)\ell_a(U_\tau, T_\tau; \tau). \tag{13}$$

The last term is independent of the student prediction when the teacher target is detached. Random target selection and deterministic target blending therefore give the same expected gradient with respect to the student prediction, although their stochastic gradient variances differ. The Bernoulli selector specifies a training estimator; this identity does not imply that it preserves action modes or guarantees an advantage over blending.

C Attention mask and summary caching

C.1 Attention mask

Figure 6 specifies the block attention used by the teacher and student streams. The current observation is shown in both streams of Figure 2; the packed sequence uses one shared, future-independent current visual group.

Current visual tokens read only current visual tokens. The teacher and student summary groups attend within their own groups and do not read actions or each other. The auxiliary video-denoising stream is used for co-training and has no forward path into student visual conditioning; shared nonvisual conditions remain independent of privileged inputs.

C.2 Current-observation encoding

The current visual group is computed from the current observation:

$$z_0 = \mathcal{E}_{\mathrm{cur}}(o_t).$$

Together with the attention mask and future-independent conditioning, this makes the student's prediction invariant to the future supplied to the teacher at fixed parameters and randomness:

$$\widehat{U}_\tau^{\mathrm{S}}(C, A_\tau, \tau; \widetilde{F}_1) = \widehat{U}_\tau^{\mathrm{S}}(C, A_\tau, \tau; \widetilde{F}_2).$$

The dependence of the selected current latent must satisfy this property at the encoder output, before DiT attention is applied.

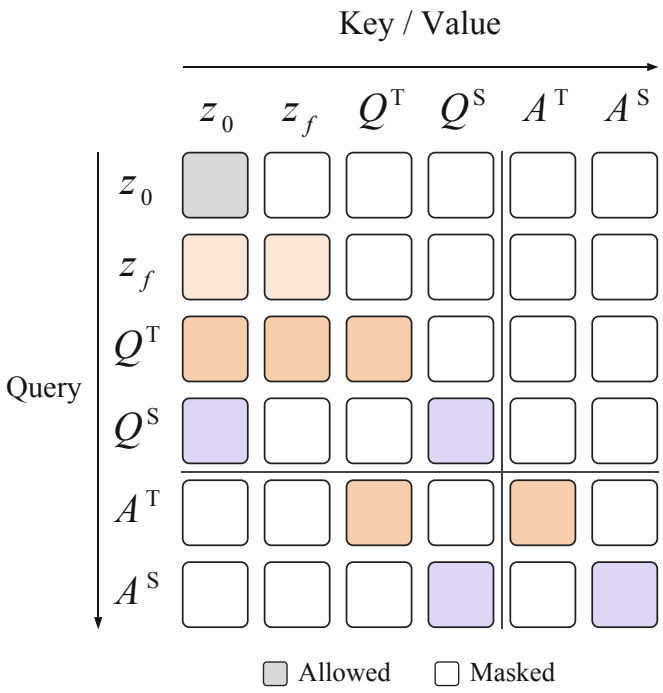


Figure 6: **Block attention for paired visual summaries and action prediction.** Rows denote query groups and columns denote key/value groups; filled blocks indicate allowed attention. z_0 and z_f are current and privileged future visual tokens. Q^{T} and Q^{S} label summary-token groups, including their hidden states across layers. A^{T} and A^{S} label action-token groups initialized from the same noisy chunk A_τ. Teacher summaries access current and future visual tokens, while student summaries access only current visual tokens. Each summary group also attends within itself. Each action stream reads its own summaries and actions. Nonvisual conditioning and the auxiliary video-denoising stream are omitted.

C.3 Summary cache

Student summaries use a fixed clean-current visual timestep condition. Their attention inputs and modulation are independent of both the noisy action chunk and the action flow time, giving

$$\mathcal{K}^{\mathrm{S}}(C; A_\tau, \tau) = \mathcal{K}^{\mathrm{S}}(C).$$

At each replanning step, the Video DiT processes the current observation and student queries once. The layer-wise summary keys and values are cached and reused across all action-denoising steps; the Action DiT receives the changing A_τ and τ. A new deployment context triggers cache reconstruction.

D EXPERIMENTAL DETAILS

Table 5 summarizes HSD's training and evaluation settings on LIBERO and RoboTwin 2.0. Real-world settings are given in Section 4.3.

Table 5: HSD training and evaluation settings on LIBERO and RoboTwin 2.0.

Setting	LIBERO	RoboTwin 2.0
Video initialization	Wan2.2-TI2V-5B	Wan2.2-TI2V-5B
Optimizer	AdamW	AdamW
Learning rate	10^{-4}	10^{-4}
Training epochs	5	10
Global batch size	128	512
Learning-rate decay	Cosine	Cosine
Warmup fraction	5%	5%
Weight decay	0.01	0.01
Gradient clipping	1.0	1.0
Action horizon H	32	32
Summary tokens per branch K	32	32
Video/action hidden width	3072 / 1024	3072 / 1024
Expert layers L	30	30
Teacher-target probability ρ	0.5	0.5
Loss weights $(\lambda_{\text{vid}}, \lambda_{\text{T}}, \lambda_{\text{S}}, \lambda_{\text{sum}})$	$(1, 1, 1, 1)$	$(1, 1, 1, 1)$
Future corruption probability	0.1	0.1
Action-denoising steps	10	10
Trials per task per condition	50	100
Camera views and final layout	Two, 224×448	Three, 384×320

Evaluation. We compute each task's success rate and average across tasks within each LIBERO suite or RoboTwin condition. The LIBERO average is the unweighted mean of the four suite scores; the RoboTwin average is the mean of Clean and Randomized scores. Evaluation uses 50 trials per LIBERO task and 100 trials per RoboTwin task and condition.

Latency measurement. All models use one NVIDIA H20, BF16, batch size 1, and cached language conditioning; means cover 100 runs after 30 warm-up runs. Inference settings broadly follow the official recipes. Fast-WAM and HSD use two 224×224 camera views and ten action-denoising steps, with $H = 32$. Fast-WAM-Joint generates video jointly, while Fast-WAM-IDM first performs ten video-denoising steps. LingBot-VA uses two 128×128 views, 20 video-denoising steps, and 50 action-denoising steps. Cosmos Policy uses two 224×224 views and five sampling steps to jointly predict actions, future states, and values.