RETHINKING CLASSIFIER-FREE GUIDANCE IN ON-POLICY DIFFUSION DISTILLATION

Anonymous authors
Paper under double-blind review

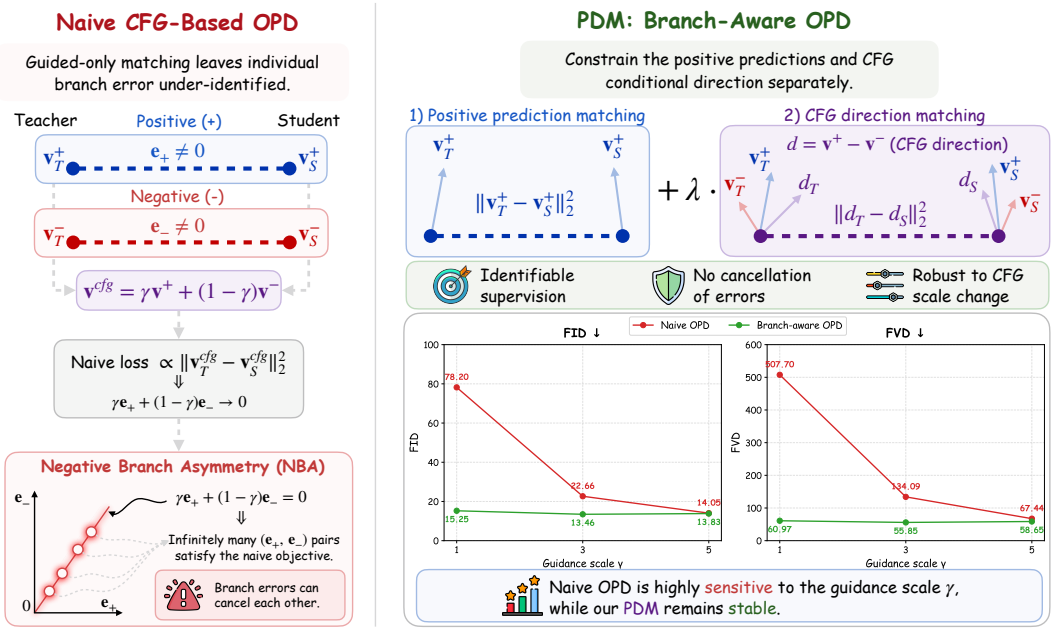


Figure 1: **Naive CFG-based OPD suffers from branch ambiguity.** Matching only the CFG-composed prediction lets positive- and negative-branch errors cancel, admitting infinitely many degenerate solutions. Our branch-aware objective, Positive–Direction Matching (PDM), constrains the positive prediction and the CFG conditional direction separately, restoring identifiable supervision across guidance scales.

ABSTRACT

On-policy distillation (OPD) adapts diffusion models by querying a teacher along trajectories generated by the current student, but how it should behave under classifier-free guidance (CFG), a default component of modern diffusion systems, remains poorly understood. Existing OPD methods naturally extend velocity matching to the CFG-composed prediction, directly matching teacher and student guided velocities. We show that this objective is under-identified at the branch level: positive- and negative-branch errors can compensate in the guided prediction. Through two contrasting cases, we find that naive matching remains effective under shared negative conditioning, where both branch errors decrease jointly. When the model's native CFG schema retains privileged information in the teacher's negative branch that is unavailable to the student, however, this joint reduction breaks down and the composed objective induces antagonistic branch-error dynamics, reducing the positive-branch error while increasing the negative-branch error. We term this failure mode Negative Branch Asymmetry (NBA). To address NBA, we introduce Positive–Direction Matching (PDM), a branch-aware OPD objective that separately constrains the positive prediction and the CFG conditional direction. We apply PDM to dense-to-sparse video control, where naive guided matching is highly sensitive to inference guidance scales, while branch-aware supervision enables more robust and effective knowledge transfer.

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.

1 INTRODUCTION

Classifier-free guidance (CFG) (Ho & Salimans, 2022) drives nearly every modern diffusion model (Labs, 2025; Esser et al., 2024; Wan et al., 2025), yet the velocity it denoises with is never produced by the network directly. Instead, CFG evaluates the model twice: a positive branch under the target text condition and a negative branch under the model's null or negative text condition. It then composes the resulting velocity predictions as $\widetilde{\mathbf{v}} = \gamma\mathbf{v}^+ + (1-\gamma)\mathbf{v}^-$, where the guidance scale γ is a deployment knob chosen at inference, often long after training.

On-policy distillation (OPD) adapts a diffusion model by letting the student generate a denoising trajectory with its current policy, evaluating teacher and student at the same student-visited states, and updating the student to match the teacher's local velocity (Jiang et al., 2026; Fang et al., 2026; Li et al., 2026b). The combination of on-policy state coverage and dense per-step supervision makes OPD an appealing alternative to offline distillation and scalar-reward optimization (Agarwal et al., 2024; Zhong et al., 2026; Wang et al., 2026b).

Faced with a guided teacher, diffusion distillation can handle CFG in different ways. Some methods absorb a CFG-defined teacher field into a single student prediction, thereby internalizing the guidance strength used to construct the target (Luo et al., 2023; Yin et al., 2024b; Zhou et al., 2026). In the setting considered here, however, both teacher and student retain the standard CFG interface of the underlying diffusion model and produce separate positive and negative predictions (Fang et al., 2026; Li et al., 2026b). A natural extension of OPD is then to match their CFG-composed velocities, $\mathcal{L}_{\text{naive}} \propto \|\widetilde{\mathbf{v}}^T - \widetilde{\mathbf{v}}^S\|_2^2$. Because the two branches remain available, the student can still be evaluated with different guidance scales at inference; however, the training loss constrains only their composition at the training scale.

This composed objective is under-identified at the branch level. Let $\mathbf{e}_+ = \mathbf{v}_T^+ - \mathbf{v}_S^+$ and $\mathbf{e}_- = \mathbf{v}_T^- - \mathbf{v}_S^-$ denote the positive- and negative-branch errors. Guided matching constrains only $\gamma\mathbf{e}_+ + (1-\gamma)\mathbf{e}_-$, allowing the two errors to compensate without changing the composed prediction. Branch ambiguity alone does not imply failure. Under shared negative conditioning, shared-parameter updates may reduce both branch errors jointly. Under privileged negative conditioning, however, our experiments show that this joint reduction can break down, exposing the compensation freedom of the composed objective.

To examine how negative-branch conditioning affects OPD optimization, we track the positive- and negative-branch errors throughout training in two image-domain distillation settings. In text-rendering distillation (Li et al., 2026b), both errors decrease jointly and naive matching remains effective. In reference-conditioned distillation (Jiang et al., 2026), where the teacher's negative branch receives privileged information, naive matching instead reduces the positive-branch error while increasing the negative-branch error, inducing *antagonistic branch-error dynamics*. We term this failure mode *Negative Branch Asymmetry* (NBA).

NBA can remain hidden at the guidance scale used for training, because the opposing branch errors compensate in the composed prediction. Once the guidance scale changes at inference, the branches are recomposed with different weights and the learned compensation no longer holds. Guidance-scale sensitivity is therefore a direct observable consequence of NBA, as illustrated in Figure 1: a student can closely match the teacher at γ_{train} yet drift when evaluated at another scale.

The remedy is to supervise before composition. Our primary objective, *Positive–Direction Matching* (PDM), separately constrains the positive prediction and the CFG conditional direction $\mathbf{v}^+ - \mathbf{v}^-$, so that zero loss forces both branch errors to zero. As a foil, we also study *Independent Branch Matching* (IBM), which directly matches the two branches. We then apply branch-aware OPD to dense-to-sparse video control on Wan-VACE (Jiang et al., 2025), where the teacher observes dense per-frame control while the student receives only sparse keyframes.

Our contributions are summarized as follows:

- We show that CFG-composed OPD is under-identified at the branch level and identify when this ambiguity becomes harmful. It remains benign under shared negative conditioning, but produces NBA when privileged negative conditioning breaks joint branch-error reduction and induces antagonistic branch-error dynamics.

- We introduce *Positive–Direction Matching* (PDM), a branch-aware OPD objective that supervises the positive prediction and the CFG conditional direction separately, preventing cross-branch error compensation and improving robustness across inference guidance scales. We include Independent Branch Matching (IBM) as a branch-aware foil.
- We apply branch-aware OPD to dense-to-sparse video control and provide quantitative evidence across pose, depth, and scribble modalities. Naive matching exhibits severe guidance-scale sensitivity, while branch-aware supervision transfers dense control knowledge more reliably.

2 PRELIMINARIES

2.1 ON-POLICY DISTILLATION FOR DIFFUSION MODELS

Diffusion on-policy distillation (OPD) trains a student model using states sampled from its own evolving generation process (Jiang et al., 2026; Fang et al., 2026; Li et al., 2026b). Let $\mathbf{x}_t^S \sim p_\theta$ denote a noisy state visited along a trajectory generated by the current student, at which the teacher and student define local denoising transitions conditioned on $\mathbf{c}_T$ and $\mathbf{c}_S$, respectively. The teacher may have access to privileged information unavailable or only partially available to the student, so $\mathbf{c}_T$ and $\mathbf{c}_S$ need not be identical. Existing formulations (Jiang et al., 2026; Fang et al., 2026; Li et al., 2026b) differ in derivation and optimization, but their transition-level objectives all reduce, up to a timestep-dependent weighting factor, to matching the corresponding model predictions:

$$\mathcal{L}_{\mathrm{OPD}} = \mathbb{E}_{\mathbf{x}_t^S \sim p_\theta,\, t}\left[w(t) \left\| \mathbf{v}_T\left(\mathbf{x}_t^S, t, \mathbf{c}_T\right) - \mathbf{v}_S\left(\mathbf{x}_t^S, t, \mathbf{c}_S\right) \right\|_2^2 \right], \tag{1}$$

where $\mathbf{v}_T$ and $\mathbf{v}_S$ denote the teacher and student velocity predictions, and $w(t)$ absorbs the parameterization- and solver-dependent weighting. We use velocity prediction throughout, while the same formulation applies to equivalent noise, score, or flow parameterizations.

2.2 DIFFUSION OPD WITH CLASSIFIER-FREE GUIDANCE

We next consider diffusion models that use classifier-free guidance (CFG) with guidance scale $\gamma > 1$. The velocity used for denoising is

$$\widetilde{\mathbf{v}} = \gamma \mathbf{v}^+ + (1-\gamma)\mathbf{v}^-, \tag{2}$$

where $\mathbf{v}^+$ and $\mathbf{v}^-$ denote the positive and negative branch predictions, respectively. We preserve the CFG schema prescribed by each model's original training and inference pipeline. In many models, the negative branch changes the text condition while retaining auxiliary inputs such as reference images or other control signals.

Define the positive- and negative-branch discrepancies as

$$\mathbf{e}_+ = \mathbf{v}_T^+ - \mathbf{v}_S^+, \qquad \mathbf{e}_- = \mathbf{v}_T^- - \mathbf{v}_S^-. \tag{3}$$

Substituting these discrepancies into the CFG-composed matching objective gives

$$\mathcal{L}_{\mathrm{naive}} = \mathbb{E}_{\mathbf{x}_t^S \sim p_\theta,\, t}\left[w(t) \left\| \gamma \mathbf{e}_+ + (1-\gamma)\mathbf{e}_- \right\|_2^2 \right]. \tag{4}$$

Existing branch-retaining OPD settings (Fang et al., 2026; Li et al., 2026b) are primarily studied in text-to-image distillation, where teacher and student share the same null-text conditioning on the negative branch. Such symmetry is not guaranteed in OPD with privileged information, where the teacher may receive reference or control information unavailable or only partially available to the student. Because the model's native CFG schema may retain this information in the negative branch, privileged negative conditioning arises naturally in practical knowledge-transfer settings. We analyze its consequence next.

3 NEGATIVE BRANCH ASYMMETRY AND BRANCH-AWARE OPD

3.1 NEGATIVE BRANCH ASYMMETRY

Under privileged negative conditioning, the teacher's negative target cannot be directly reproduced from the student's negative input. Nevertheless, Eq. 4 constrains only the CFG-composed discrepancy. At any supervised state, matching the guided predictions requires only $\gamma \mathbf{e}_+ + (1-\gamma)\mathbf{e}_- = \mathbf{0}$,

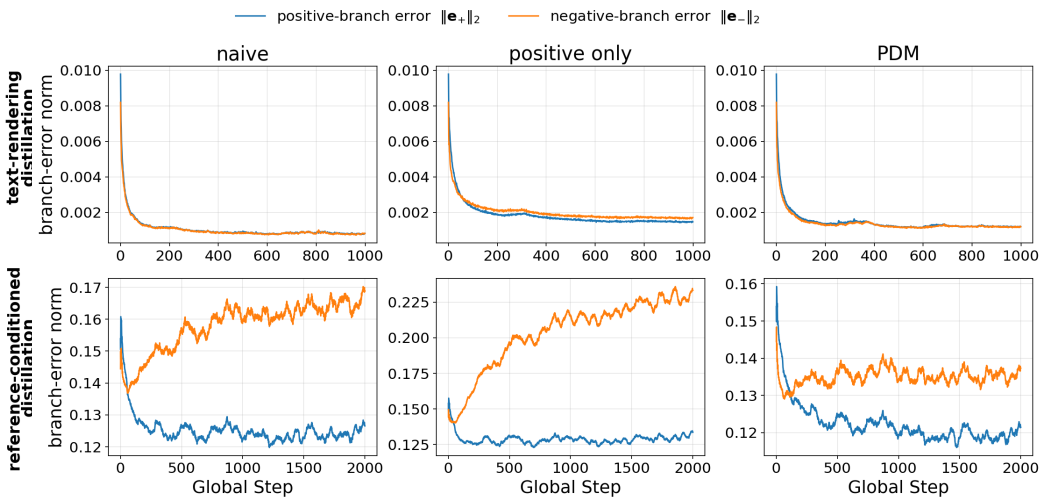


Figure 2: **Branch-error dynamics under shared and privileged negative conditioning.** Positive- and negative-branch ℓ_2 error norms during text-rendering (top) and reference-conditioned (bottom) distillation. Under shared conditioning both errors decrease jointly for all objectives; under privileged reference conditioning, naive and positive-only matching reduce the positive error while increasing the negative one. PDM suppresses this degradation.

or equivalently $\mathbf{e}_+ = \frac{\gamma-1}{\gamma}\mathbf{e}_-$. This condition admits infinitely many non-zero branch-error pairs, allowing the two errors to compensate without changing the guided prediction.

It establishes the branch ambiguity but does not determine its optimization dynamics. We therefore investigate when this ambiguity becomes harmful empirically in Section 4.1. Across two controlled image-domain cases, we observe joint branch-error reduction under shared negative conditioning, but antagonistic branch-error dynamics under privileged negative conditioning.

Definition: Negative Branch Asymmetry (NBA).

Branch ambiguity alone does not constitute a failure. We define NBA as the failure mode in which, under privileged negative conditioning, naive CFG-composed matching induces antagonistic branch-error dynamics, reducing $\|\mathbf{e}_+\|_2$ while increasing $\|\mathbf{e}_-\|_2$. The opposing errors can remain hidden at the training guidance scale and become exposed when the branches are recomposed at another scale.

An absolute performance drop under a guidance-scale change is not by itself evidence of NBA, since the teacher may intrinsically rely on CFG. The relevant signature is excess degradation introduced by naive distillation: the student departs from the teacher's guidance-scale behavior or degrades more strongly than its branch-aware counterpart. Section 4.1 tests these predictions empirically.

3.2 Branch-Aware OPD

To prevent cross-branch error compensation, we supervise the predictions before CFG composition. Let ℓ denote a pointwise distillation loss evaluated at a student-visited state $\mathbf{x}_t^S$ and timestep t. Its corresponding OPD objective is

$$\mathcal{L}_{\mathrm{OPD}}[\ell] = \mathbb{E}_{\mathbf{x}_t^S \sim p_\theta,\, t}\left[w(t)\,\ell(\mathbf{x}_t^S, t)\right]. \tag{5}$$

Positive–Direction Matching. For a model $M \in \{T, S\}$, define its CFG conditional direction as $\mathbf{d}_M = \mathbf{v}_M^+ - \mathbf{v}_M^-$. Our primary objective, Positive–Direction Matching (PDM), separately matches the positive prediction and the conditional direction:

$$\ell_{\mathrm{PDM}} = \|\mathbf{e}_+\|_2^2 + \lambda\,\|\mathbf{d}_T - \mathbf{d}_S\|_2^2 = \|\mathbf{e}_+\|_2^2 + \lambda\,\|\mathbf{e}_+ - \mathbf{e}_-\|_2^2, \tag{6}$$

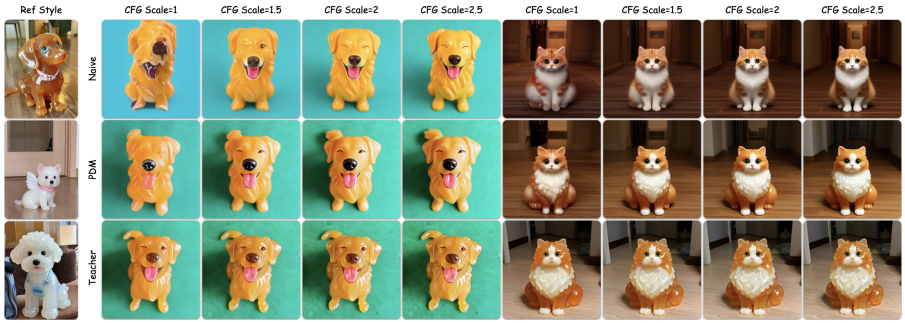


Figure 3: **Distillation-induced CFG sensitivity under privileged reference conditioning.** Left column: reference-style exemplars. For two held-out prompts, the reference-conditioned teacher is compared with text-only students trained by naive matching or PDM at $\gamma_{\text{train}} = 2$, under matched prompts and initial noise. PDM follows the teacher across guidance scales; naive matching drifts, most notably at $\gamma = 1$. More examples in Appendix 7.4.

Table 1: **Intrinsic CFG sensitivity in text-rendering distillation.** OCR reward (%, ↑) on 1,018 held-out prompts. All models exhibit a similar response to the inference guidance scale, showing that the drop at $\gamma = 1$ is inherited from the teacher rather than introduced by naive distillation.

Method	γ=1	γ=2	γ=3	γ=4	γ=4.5
Teacher	75.24	92.91	94.16	94.43	94.09
OPD (Naive)	73.87	92.72	94.52	94.43	94.11
OPD (PDM)	74.48	93.73	93.99	94.02	94.38

where $\lambda > 0$ controls the direction-matching strength. The first term anchors the positive prediction, while the second preserves the CFG conditional direction. Zero PDM loss requires $\mathbf{e}_+ = \mathbf{e}_- = \mathbf{0}$, eliminating the compensation freedom of guided-only matching.

Baseline: Independent Branch Matching. To separate the benefit of branch-level supervision from the particular design of PDM, we include Independent Branch Matching (IBM) as a baseline:

$$\ell_{\text{IBM}} = \gamma^2 \left\|\mathbf{e}_+\right\|_2^2 + (\gamma - 1)^2 \left\|\mathbf{e}_-\right\|_2^2 . \tag{7}$$

IBM retains the per-branch weights of the expanded naive objective but removes its cross-branch interaction term. Like PDM, it has a unique branch-level zero-loss solution for $\gamma > 1$, but directly matches the two branches rather than preserving the positive prediction and conditional direction. We compare the empirical behavior of these two branch-aware formulations in Table 2.

3.3 Efficient Supervision for Dense-to-Sparse Video Control

Full-trajectory OPD is particularly expensive for video because it requires a teacher evaluation at every denoising state. For our dense-to-sparse video application, we independently find that supervising only an early portion of the student trajectory substantially reduces this cost while maintaining performance.

Let $\{\mathbf{x}_{t_k}^S\}_{k=0}^{N-1}$ denote the states visited by the student in denoising order. We supervise only the first K states:

$$\mathcal{L}_{\text{OPD}}^{(K)} = \mathbb{E}_{\{\mathbf{x}_{t_k}^S\}_{k=0}^{N-1} \sim p_\theta} \left[\frac{1}{K} \sum_{k=0}^{K-1} w(t_k)\, \ell(\mathbf{x}_{t_k}^S, t_k) \right]. \tag{8}$$

The student still completes the full denoising rollout, but teacher supervision is computed only for these K states. Here $K = N$ recovers full-trajectory supervision, while smaller K reduces training cost. This design is used only for the dense-to-sparse video experiments; the image-domain case studies follow their respective standard supervision protocols. We study the effect of K in Section 4.3.

Table 2: **Main results on dense-to-sparse video control.** Models are jointly trained on pose, depth, and scribble controls and evaluated separately on each modality at the training guidance scale $\gamma = 5$. **Bold** and underlined values denote the best and second-best results, respectively, excluding the teacher.

Pose-Control	MPJPE ↓		PCK@0.2 ↑		PCK@0.1 ↑		Quality ↓	
Method	All	Key	All	Key	All	Key	FID	FVD
Teacher	3.03	2.82	96.02	96.46	91.15	92.23	14.13	56.01
Student	5.92	4.10	86.78	91.89	69.97	82.53	15.81	81.41
SFT	5.02	3.31	90.36	94.77	76.72	88.55	14.11	63.90
Off-policy	5.14	3.44	89.78	94.09	75.46	87.48	14.49	68.50
OPD (Naive)	4.43	3.14	91.84	95.25	80.28	89.35	13.49	65.12
OPD (IBM)	4.20	2.88	92.93	96.11	82.27	90.95	**13.26**	65.72
OPD (PDM)	**4.13**	**2.83**	**92.98**	**96.12**	**82.48**	**91.03**	13.52	**62.77**
Depth-Control	**CORR ↑**		**RMSE ↓**		**SI-RMSE ↓**		**Quality ↓**	
Method	All	Key	All	Key	All	Key	FID	FVD
Teacher	96.98	97.58	7.50	6.36	6.19	5.23	11.02	36.89
Student	83.21	91.27	16.87	11.13	14.36	9.37	15.03	71.33
SFT	88.39	96.17	13.83	7.80	11.96	6.52	12.87	**58.13**
Off-policy	88.25	96.01	13.98	8.00	12.10	6.64	12.66	59.35
OPD (Naive)	90.58	96.56	12.39	7.47	10.69	6.16	12.08	72.20
OPD (IBM)	91.14	96.91	12.03	7.11	10.37	5.85	11.76	60.38
OPD (PDM)	**91.24**	**97.13**	**11.95**	**6.91**	**10.31**	**5.68**	**11.64**	60.60
Scribble-Control	**Recall ↑**		**Precision ↑**		**F1 ↑**		**Quality ↓**	
Method	All	Key	All	Key	All	Key	FID	FVD
Teacher	93.26	94.40	94.26	95.27	93.73	94.81	9.56	24.22
Student	64.20	82.93	65.50	83.78	64.70	83.24	12.29	69.13
SFT	67.23	89.03	69.52	90.66	68.27	89.79	10.69	50.56
Off-policy	70.98	91.55	72.15	91.73	71.49	91.60	11.60	52.90
OPD (Naive)	74.78	90.77	75.38	91.34	75.00	90.99	10.53	59.95
OPD (IBM)	75.92	92.23	**76.47**	92.70	76.13	92.43	**10.12**	52.67
OPD (PDM)	**76.42**	**93.25**	76.13	**92.86**	**76.22**	**93.03**	10.34	**50.53**

4 EXPERIMENTS

We first investigate when the branch ambiguity of CFG-composed matching develops into NBA. We then evaluate branch-aware OPD in dense-to-sparse video control and ablate its main design choices.

4.1 WHEN DOES NBA EMERGE?

4.1.1 EXPERIMENTAL SETUP

We contrast two image-domain distillation settings that differ only in negative conditioning. In text-rendering distillation (Li et al., 2026b), teacher and student share both branches, $\mathbf{c}_T^+ = \mathbf{c}_S^+ = \mathbf{y}$ and $\mathbf{c}_T^- = \mathbf{c}_S^- = \varnothing$. In reference-conditioned distillation (Jiang et al., 2026), the teacher observes a reference image $\mathbf{r}$ unavailable to the text-only student:

$$\begin{aligned} \mathbf{c}_T^+ &= (\mathbf{y}, \mathbf{r}), \quad \mathbf{c}_S^+ = \mathbf{y}, \\ \mathbf{c}_T^- &= (\varnothing, \mathbf{r}), \quad \mathbf{c}_S^- = \varnothing, \end{aligned} \tag{9}$$

so it receives privileged information on its negative branch.

To diagnose branch coupling, we train naive matching, PDM, and a positive-only ablation ($\ell_+ = \|\mathbf{e}_+\|_2^2$) that isolates how updates toward the positive target affect the negative branch, and track both branch errors throughout training. The positive-only run is used only for this optimization

Table 3: **Guidance-scale generalization on pose control.** Models are trained at $\gamma_{\text{train}} = 5$ and evaluated at different inference guidance scales.

	MPJPE ↓		PCK@0.2 ↑		PCK@0.1 ↑		Quality ↓	
Method	All	Key	All	Key	All	Key	FID	FVD
Naive ($\gamma = 5$)	4.62	3.22	91.35	95.01	79.39	89.54	14.05	67.44
Naive ($\gamma = 3$)	4.94	3.31	90.23	94.53	77.92	88.22	22.66	134.09
Naive ($\gamma = 1$)	8.98	6.23	80.08	87.42	69.14	81.57	78.20	507.70
IBM ($\gamma = 5$)	4.28	2.91	92.67	95.95	81.69	91.03	12.99	53.83
IBM ($\gamma = 3$)	4.42	3.04	92.09	95.53	81.41	90.81	13.69	56.85
IBM ($\gamma = 1$)	4.91	3.43	90.92	94.69	80.66	89.98	18.00	71.17
PDM ($\gamma = 5$)	4.27	2.75	92.71	96.38	82.12	91.66	13.83	58.65
PDM ($\gamma = 3$)	4.17	2.73	92.81	96.61	82.27	91.81	13.46	55.85
PDM ($\gamma = 1$)	4.48	2.90	92.14	95.64	81.72	91.15	15.25	60.97

diagnostic; generation comparisons use naive matching and PDM under the same protocol. Models, data, guidance scales, and full implementation details are given in Appendix 7.1.

4.1.2 Branch-Error Dynamics

Figure 2 reveals two distinct optimization regimes. In text-rendering distillation, positive-only training reduces both branch errors, showing that positive-branch updates also improve the negative branch. Naive matching and PDM exhibit the same joint error reduction, indicating a benign regime.

Reference-conditioned distillation exhibits the opposite behavior. Positive-only training reduces the positive-branch error but substantially increases the negative-branch error, showing that positive-branch updates no longer benefit the negative branch under privileged conditioning. Naive matching follows the same antagonistic pattern, decreasing the positive error while increasing the negative error. PDM instead reduces the positive error while preventing sustained negative-error growth. The antagonistic trajectory under naive matching is the optimization signature of NBA, while PDM restores joint branch-level supervision.

4.1.3 Intrinsic and Distillation-Induced CFG Sensitivity

We next examine how these two regimes manifest under inference-time guidance shifts. An absolute performance drop away from the training scale is not sufficient evidence of NBA, since the teacher itself may intrinsically rely on CFG. Table 1 shows that, in text-rendering distillation, both students track the teacher throughout the guidance sweep: at $\gamma = 1$ the teacher, naive student, and PDM student obtain OCR rewards of 75.24, 73.87, and 74.48, and all improve similarly as guidance increases.

Reference-conditioned distillation behaves differently. Figure 3 shows that the teacher preserves the reference-supplied style across guidance scales and that PDM closely tracks this behavior, whereas naive matching departs from the teacher with visual distortion and style drift, most pronounced at $\gamma = 1$ where CFG reduces to the positive prediction. This excess degradation relative to both the teacher and PDM isolates the sensitivity introduced by naive distillation.

Finding: When Does Branch Ambiguity Become Harmful?

Under shared negative conditioning, both branch errors decrease jointly and the student follows the teacher's intrinsic CFG response. Under privileged negative conditioning, naive matching induces antagonistic branch-error dynamics and excess guidance-scale sensitivity, while PDM suppresses both. Thus, excess degradation relative to the teacher or a branch-aware student—rather than an absolute CFG-free performance drop—is the diagnostic signature of NBA.

4.2 Application to Dense-to-Sparse Video Control

4.2.1 Experimental Setup

We apply branch-aware OPD to dense-to-sparse video control using Wan-VACE (Jiang et al., 2025). The teacher receives a dense control sequence $\mathbf{p}_{\text{dense}}$, while the student observes only sparse keyframes $\mathbf{p}_{\text{sparse}}$:

$$\begin{aligned} \mathbf{c}_T^+ &= (\mathbf{y}, \mathbf{p}_{\text{dense}}), \quad \mathbf{c}_S^+ = (\mathbf{y}, \mathbf{p}_{\text{sparse}}), \\ \mathbf{c}_T^- &= (\varnothing, \mathbf{p}_{\text{dense}}), \quad \mathbf{c}_S^- = (\varnothing, \mathbf{p}_{\text{sparse}}). \end{aligned} \tag{10}$$

We evaluate pose, depth, and scribble control on a 600-clip benchmark derived from OpenHumanVid (Li et al., 2025) under the dense-to-four-keyframe setting, against the unadapted sparse-control student, SFT on data-derived diffusion states, and off-policy teacher matching, all at equal architecture and training budget. PDM uses $\lambda = 1$ and $K = 8$ unless stated otherwise. Main results use joint training over all three modalities, while guidance-scale and hyperparameter studies use pose-only models to reduce repeated training cost, so the two protocols are not directly comparable. Benchmark construction, metrics, and implementation details are given in Appendix 7.1.

4.2.2 Main Results

Table 2 reports results at the training guidance scale $\gamma = 5$. PDM provides the most consistent control-fidelity improvements across pose, depth, and scribble, while IBM also outperforms naive matching on most control metrics. The performance of IBM supports the benefit of removing cross-branch compensation, while the stronger overall results of PDM support its positive–direction parameterization.

Because these gains are measured at $\gamma = \gamma_{\text{train}}$, they establish the value of branch-aware OPD independently of guidance-scale shifts: such supervision improves dense-to-sparse knowledge transfer itself, rather than merely correcting off-scale degradation.

4.2.3 Guidance-Scale Generalization

Table 3 evaluates pose-only models trained at $\gamma_{\text{train}} = 5$ and tested at different guidance scales. Naive matching degrades sharply as the inference scale moves away from training, reaching its largest drop at $\gamma = 1$. In contrast, PDM and IBM remain stable across the sweep, showing that branch-aware supervision prevents the excess guidance sensitivity induced by composed matching. PDM provides the most consistent control fidelity across the evaluated scales.

Figure 4 qualitatively illustrates the excess guidance-scale sensitivity of naive OPD expected under NBA. Additional qualitative comparisons on pose, depth, and scribble-controlled generation are provided in Appendix 7.4.

4.3 Ablation Studies

We ablate the two main design choices of PDM in the dense-to-sparse video application; the full result tables are given in Appendix 7.2. For the direction-matching weight, $\lambda = 0$ reduces PDM to positive-only matching, and $\lambda = 1$ gives the best overall control fidelity, including the lowest keyframe MPJPE and the highest PCK@0.1, while larger values improve some distributional-quality metrics but weaken control accuracy. For the supervision horizon, increasing K raises training cost from 23 seconds per step at $K = 1$ to 790 seconds under full-trajectory supervision ($K = 50$), and $K = 8$ gives the strongest control fidelity while remaining substantially cheaper than full-trajectory supervision. We therefore use $\lambda = 1$ and $K = 8$ throughout.

5 Related Work

Prior diffusion distillation is dominated by step compression: progressive distillation (Salimans & Ho, 2022), consistency models (Song et al., 2023; Kim et al., 2024; Luo et al., 2023), and distribution matching (Yin et al., 2024b; Sauer et al., 2024; Yin et al., 2024a) all target one- or few-step

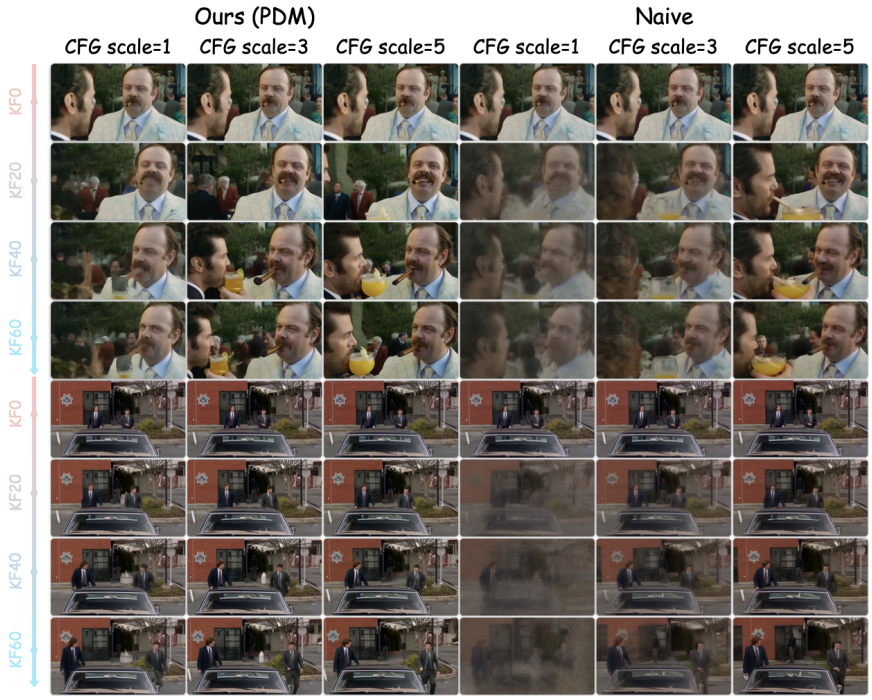


Figure 4: **Guidance-scale robustness in dense-to-sparse video control.** Models are trained at $\gamma = 5$. PDM remains stable across inference scales, whereas naive OPD degrades off-scale.

generation, and most of them absorb the CFG-composed teacher prediction into a single student output (Luo et al., 2023; Yin et al., 2024b; Wu et al., 2026; Huang et al., 2025). Diffusion OPD instead queries the teacher on states visited by the current student (Jiang et al., 2026; Fang et al., 2026; Li et al., 2026b; Zhou et al., 2026), sharing the principle that the student generates the trajectory while the teacher supplies dense local supervision. Our setting differs from both lines. We do not compress a guided teacher into a CFG-free student: teacher and student each retain the native CFG interface and construct their own guided prediction from separate positive and negative branches. Existing branch-retaining OPD is studied only under shared negative conditioning, where the two branch errors can be reduced jointly even though the negative error need not vanish, so the under-identification of composed matching stays benign and goes unexamined. We study the missing regime in which the teacher's negative branch carries privileged information, show that joint reduction then breaks down into antagonistic branch-error dynamics (NBA), and introduce branch-aware objectives that remove the compensation freedom. A detailed discussion of both lines is given in Appendix 7.3.

6 Conclusions and Limitations

We show that CFG-composed OPD is under-identified at the branch level, but that this ambiguity does not always cause failure: under shared negative conditioning the two branch errors can decrease jointly, whereas privileged negative conditioning breaks this joint reduction and drives naive matching into antagonistic branch-error dynamics, a failure mode we identify as Negative Branch Asymmetry (NBA). To address it, we introduce Positive–Direction Matching (PDM), which separately supervises the positive prediction and CFG conditional direction. Contrasting image-domain studies validate when NBA emerges, while dense-to-sparse video control demonstrates that branch-aware supervision improves knowledge transfer and robustness across inference guidance scales.

Despite these results, the relative behavior of branch-aware objectives remains incompletely understood. PDM and IBM share the same branch-level zero-loss solution under non-degenerate weights, so PDM's observed advantage is currently empirical rather than theoretical. Future work may characterize their different optimization behavior and extend NBA analysis to other guidance mechanisms and teacher–student conditioning asymmetries.

AI USE STATEMENT

Generative AI tools were used solely to polish the writing of this manuscript, including improving clarity, grammar, and style. They were not used to generate scientific content, claims, results, citations, or ideas. The authors reviewed all AI-assisted edits and take full responsibility for the final content of the paper.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

Cheng Cui, Ting Sun, Manhui Lin, Tingquan Gao, Yubo Zhang, Jiaxuan Liu, Xueqing Wang, Zelun Zhang, Changda Zhou, Hongen Liu, et al. Paddleocr 3.0 technical report. *arXiv preprint arXiv:2507.05595*, 2025.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Zhen Fang, Wenxuan Huang, Yu Zeng, Yiming Zhao, Shuang Chen, Kaituo Feng, Yunlong Lin, Lin Chen, Zehui Chen, Shaosheng Cao, et al. Flow-opd: On-policy distillation for flow matching models. *arXiv preprint arXiv:2605.08063*, 2026.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Yubo Huang, Hailong Guo, Fangtai Wu, Weiqiang Wang, Shifeng Zhang, Shijie Huang, Qijun Gan, Lin Liu, Sirui Zhao, Enhong Chen, et al. Live avatar: Streaming real-time audio-driven avatar generation with infinite length. *arXiv preprint arXiv:2512.04677*, 2025.

Dengyang Jiang, Xin Jin, Dongyang Liu, Zanyi Wang, Mingzhe Zheng, Ruoyi Du, Xiangpeng Yang, Qilong Wu, Zhen Li, Peng Gao, et al. D-opsd: On-policy self-distillation for continuously tuning step-distilled diffusion models. *arXiv preprint arXiv:2605.05204*, 2026.

Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17191–17202, 2025.

Dongjun Kim, Chieh-Hsin Lai, WeiHsiang Liao, Naoki Murata, Yuhta Takida, Toshimitsu Uesaka, Yutong He, Yuki Mitsufuji, and Stefano Ermon. Consistency trajectory models: Learning probability flow ode trajectory of diffusion. In *International Conference on Learning Representations*, volume 2024, pp. 44493–44525, 2024.

Black Forest Labs. FLUX.2: Frontier Visual Intelligence. `https://bfl.ai/blog/flux-2`, 2025.

Bingnan Li, Chen-Yu Wang, Haiyang Xu, Xiang Zhang, Ethan Armand, Divyansh Srivastava, Shan Xiaojun, Zeyuan Chen, Jianwen Xie, and Zhuowen Tu. Overlaybench: A benchmark for layout-to-image generation with dense overlaps. *Advances in Neural Information Processing Systems*, 38, 2026a.

Hui Li, Mingwang Xu, Yun Zhan, Shan Mu, Jiaye Li, Kaihui Cheng, Yuxuan Chen, Tan Chen, Mao Ye, Jingdong Wang, et al. Openhumanvid: A large-scale high-quality dataset for enhancing human-centric video generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 7752–7762, 2025.

Quanhao Li, Junqiu Yu, Kaixun Jiang, Yujie Wei, Zhen Xing, Pandeng Li, Ruihang Chu, Shiwei Zhang, Yu Liu, and Zuxuan Wu. Diffusionopd: A unified perspective of on-policy distillation in diffusion models. *arXiv preprint arXiv:2605.15055*, 2026b.

Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023.

René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence*, 44(3):1623–1637, 2020.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pp. 87–103. Springer, 2024.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. 2023.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Haozhe Wang, Weijia Feng, Jinpeng Yu, Che Liu, Ping Nie, Fangzhen Lin, Jiaming Liu, Ruihua Huang, Jimmy Lin, Wenhu Chen, et al. Search beyond what can be taught: Evolving the knowledge boundary in agentic visual generation. *arXiv preprint arXiv:2607.05382*, 2026a.

Haozhe Wang, Cong Wei, Weiming Ren, Jiaming Liu, Fangzhen Lin, and Wenhu Chen. RationalRewards: Reasoning rewards scale visual generation both training and test time. *arXiv preprint arXiv:2604.11626*, 2026b. URL https://arxiv.org/abs/2604.11626.

Fangtai Wu, Hailong Guo, Shijie Huang, Jiayi Song, Yubo Huang, Mushui Liu, Zhao Wang, Yunlong Yu, Jiaming Liu, and Ruihua Huang. Collectionlora: Collecting 50 effects in 1 lora via multi-teacher on-policy distillation. *arXiv preprint arXiv:2605.25378*, 2026.

Zhendong Yang, Ailing Zeng, Chun Yuan, and Yu Li. Effective whole-body pose estimation with two-stages distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4210–4220, 2023.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. *Advances in neural information processing systems*, 37:47455–47487, 2024a.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6613–6623. IEEE, 2024b.

Qianhao Yuan, Jie Lou, Xing Yu, Hongyu Lin, Le Sun, Xianpei Han, and Yaojie Lu. Vision-opd: Learning to see fine details for multimodal llms via on-policy self-distillation. *arXiv preprint arXiv:2605.18740*, 2026.

Wenliang Zhao, Lujia Bai, Yongming Rao, Jie Zhou, and Jiwen Lu. Unipc: A unified predictor-corrector framework for fast sampling of diffusion models. *Advances in Neural Information Processing Systems*, 36:49842–49869, 2023.

Qiyong Zhong, Mao Zheng, Mingyang Song, Xin Lin, Jie Sun, Houcheng Jiang, Xiang Wang, and Junfeng Fang. Sod: Step-wise on-policy distillation for small language model agents. *arXiv preprint arXiv:2605.07725*, 2026.

Wei Zhou, Xiongwei Zhu, Zelin Xu, Bo Dong, Lixue Gong, Yongyuan Liang, Meng Chu, Leigang Qu, Lingdong Kong, Wei Liu, et al. Danceopd: On-policy generative field distillation. *arXiv preprint arXiv:2606.27377*, 2026.

7 APPENDIX

7.1 ADDITIONAL EXPERIMENTAL DETAILS

7.1.1 TEXT RENDERING DISTILLATION

Models. The base model is Stable Diffusion 3.5-Medium (Esser et al., 2024). Both the teacher and the student are LoRA adapters (Hu et al., 2021) of rank $r = 32$ and scale $\alpha = 64$, applied to the query/key/value and output projections of every self- and cross-attention block in the MM-DiT transformer; all other weights are frozen. We use the same OCR teacher model used in DiffusionOPD (Li et al., 2026b); the student is initialized from the base model (identity adapter) and is the only module updated during distillation.

Objectives. We compare, under identical rollouts and hyper-parameters, the naive CFG-composed objective, Positive-Direction Matching with $\lambda = 2$ and the positive-only ablation. Only the pointwise objective differs across methods. The positive- and negative-branch errors reported in Figure 2 are per-transition ℓ_2 norms $\|\mathbf{e}_\pm\|_2$ in transition-mean space, averaged over the supervised timesteps and logged every optimizer step.

Optimization. We use AdamW (β_1=0.9, β_2=0.999, ϵ=10^{-8}, weight decay 10^{-4}) with a constant learning rate of 3×10^{-4} and gradient clipping at norm 1.0, keeping an EMA of the trainable parameters (decay 0.9, updated every 8 steps). Training runs on 8×H100 GPUs in fp16 mixed precision with a per-GPU sampling batch of 3 prompts, one image per prompt, and 3 sampling batches accumulated per update, giving an effective batch of $8 \times 3 \times 3 = 72$ trajectories ($\times K$ transitions) per step. Each round performs a single optimizer update; we train for 1000 updates with seed 42.

Data and reward. Prompts are drawn from the OCR text-rendering set (19,652 train / 1018 test) used in DiffusionOPD (Li et al., 2026b), where the target string s to be rendered is given in quotes. For evaluation, we read the text $\hat{s}$ from the generated image with PaddleOCR (Cui et al., 2025) and score

$$R_{\text{OCR}} \;=\; 1 - \frac{\text{Lev}(\hat{s}, s)}{\max(|\hat{s}|,\, |s|)} \;\in\; [0, 1], \tag{11}$$

where $\text{Lev}(\hat{s}, s)$ is the Levenshtein (edit) distance between the two strings—the minimum number of single-character insertions, deletions, or substitutions to turn $\hat{s}$ into s—and $|\cdot|$ denotes string length. A score of 1 means the rendered text is read back exactly; 0 means no characters match.

Evaluation. Unless noted, models are evaluated on the full 1018-prompt test set with 40 denoising steps in fp16. For the guidance-scale generalization study we sweep the inference guidance $\gamma \in \{1, 2, 3, 4, 4.5\}$ while keeping the $\gamma_{\text{train}} = 4.5$ training configuration fixed.

7.1.2 REFERENCE-CONDITIONED DISTILLATION

Data and task construction. Following the style-learning experiment in Jiang et al., we construct a style-distillation setting spanning six distinct visual styles, denoted by the placeholders [A]–[F]. For each style we provide four training demonstrations (Wu et al., 2026; Wang et al., 2026a), each pairing a text prompt with a reference image that exposes the target style to the teacher, yielding $6 \times 4 = 24$ training demonstrations in total. The demonstrations are organized by style: every demonstration for style [A] shares the same target appearance, and likewise for [B]–[F]. We train the LoRA for 2000 gradient steps, drawing a batch of 16 sampled records per gradient step; different visits use independently sampled initial noise and denoising states, so repeated exposure to a demonstration still yields stochastic training observations. For qualitative evaluation, we construct 12 held-out test prompts that invoke the same style placeholders while changing the surrounding content, composition, or interactions. The goal is to test whether the LoRA can distill the style knowledge exposed by the seen demonstrations into its weights and, at inference, transfer a learned style to unseen text prompts. The resulting evaluation tests whether the text-only student internalizes the reference-supplied style through OPD rather than reproducing any particular training scene.

Model and conditioning. We use the multi-step FLUX.2-klein-4B-base model (Labs, 2025) at 512×512 resolution. Each reference image is converted to RGB, bicubically resized to cover the target canvas, center-cropped, and normalized to $[-1, 1]$ before VAE encoding. The student follows the original text-to-image interface and receives only the prompt. The teacher uses the same diffusion backbone in its image-editing interface: it receives the student-visited noisy latent together with the encoded reference-image latents, and its text condition appends the instruction "Make the overall style consistent with the reference image." For the teacher's negative prediction, we replace the text with the null prompt while retaining the reference-image latents. The student's negative prediction uses the same null prompt without reference latents.

Parameterization and teacher update. We freeze the VAE, text encoder, and base diffusion-transformer weights and optimize LoRA adapters with rank 64 and scaling parameter $\alpha = 128$. The adapters are applied to the query, key, value, and output projections of the self- and context-attention modules, together with the corresponding feed-forward projections in each transformer block. We maintain separate student and teacher adapters initialized from the same weights. After each student update, we update the teacher adapter with exponential moving average decay 0.9999. Teacher predictions are evaluated without gradients.

On-policy objectives. For each batch, the student initializes a trajectory from Gaussian noise under the text-only condition. We use the native FLUX.2-klein (Labs, 2025) scheduler with a 30-step denoising grid. At every supervised state, the teacher and student construct their own positive and negative predictions. The student rollout uses its CFG-composed velocity at $\gamma_{\text{train}} = 2$, while the teacher is evaluated at the same student-visited state. We implement naive OPD by matching the teacher and student after CFG composition. PDM instead matches the teacher's positive prediction and the CFG conditional direction separately, with $\lambda = 2$. All compared objectives use the same base model, reference–prompt pairs, student rollout states, and optimization budget.

Optimization. We train with AdamW using learning rate 10^{-4}, $\beta_1 = 0.9$, $\beta_2 = 0.999$, zero weight decay, and gradient-norm clipping at 1.0. Training uses eight processes with a per-process batch size of 2 and no gradient accumulation, giving a global batch size of 16. We use bfloat16 mixed precision for the trainable model and VAE, gradient checkpointing, and DeepSpeed ZeRO-2.

Inference and qualitative evaluation. At inference, we discard the teacher adapter and reference image; the student receives only the held-out text prompt. We generate 512×512 images with 40 denoising steps and evaluate guidance scales $\gamma \in \{1, 1.5, 2, 2.5\}$. For a given evaluation prompt, all objectives and guidance scales use the same initial noise, isolating changes caused by the training objective and CFG composition. We assess the resulting images qualitatively by inspecting whether they preserve the defining visual knowledge while following the changed context of the evaluation prompt and maintaining generation quality. This experiment is an illustrative qualitative case study; we do not interpret it as an aggregate estimate of performance.

7.1.3 Dense-to-Sparse Video Benchmark

Our dense-to-sparse video benchmark is derived from OpenHumanVid (OHV) (Li et al., 2025). Starting from the 12-part OHV release ($\approx$ 91K clips with DWPose annotations), we first filter clips according to minimum length and motion magnitude. We then split videos into non-overlapping 81-frame clips at 832×480 resolution while preserving pixel alignment between RGB and control streams.

Difficulty stratification. To reduce evaluation bias toward easier samples (Li et al., 2026a), we compute a motion score based on frame-to-frame differences of rendered DWPose skeletons. The clips are divided into six difficulty buckets according to this score, and we randomly sample 100 clips from each bucket to form the 600-clip test set. The remaining clips are used for training and validation. Please refer to Table 4 for more details.

7.1.4 Sparse Keyframe Control

All video experiments target the dense-to-sparse setting: the teacher receives the control signal at *every* frame, while the student receives it only at a sparse set of pixel-space keyframes. Following the

Bucket	Motion score	#Test	#Train
`diff1.5-2.0`	$[1.5, 2.0)$	100	2,194
`diff2.0-2.5`	$[2.0, 2.5)$	100	6,355
`diff2.5-3.5`	$[2.5, 3.5)$	100	8,992
`diff3.5-5.0`	$[3.5, 5.0)$	100	4,851
`diff5.0-7.0`	$[5.0, 7.0)$	100	1,319
`diff7.0+`	$[7.0, \infty)$	100	364
Total		600	$\approx$24K

Table 4: Difficulty-stratified benchmark. The test set draws 100 clips from each of six motion-difficulty buckets (600 total); motion score is the mean frame-to-frame pixel difference of the rendered DWPose skeleton.

Base model & adapter	
Base model	Wan2.1-VACE-1.3B (frozen)
Adapter	LoRA, rank 32, $\alpha = 32$
LoRA target	self/cross-attn $\{q, k, v, o\}$, ffn $\{0, 2\}$ per VACE block
Precision	bf16 autocast, gradient checkpointing
Optimization	
Optimizer	AdamW, $\beta = (0.9, 0.999)$, wd $= 0$, $\epsilon = 10^{-8}$
Learning rate	1×10^{-4} (constant)
Training steps	2000
Devices / batch	8 GPUs (DDP), per-GPU batch 1
Diffusion & rollout	
Solver / schedule	UniPC, 50 steps, `shift`$= 16$
OPD rollout K	8 active steps ("8-of-50")
Guidance scale γ	5.0
Data	
Resolution / length	480×832, 81 frames
Keyframes	pixel $\{0, 20, 40, 60\}$ (0 = reference)
Train / val / test	$\approx$24K / 24 / 600 clips

Table 5: Training and inference hyperparameters, shared across all distillation objectives.

masked video-to-video (MV2V) interface of the VACE (Jiang et al., 2025) base model, the student input frame stack F and mask M (over 81 frames at 480×832, in the $[-1, 1]$ range) are built as

- $F[0]$ = the reference image (appearance anchor), with $M[0] = 0$ (given, not generated);
- $F[k]$ = the control frame at each pose keyframe $k \in \{20, 40, 60\}$;
- all remaining frames are blank (-1), with $M[\,\cdot\,] = 1$ (to be synthesized).

We use the uniform layout `pixel_kf`$= \{0, 20, 40, 60\}$ throughout: index 0 is reserved for the reference image, so the model is conditioned on pose at three interior keyframes out of 81 frames. Keyframes are indexed in *pixel* space; under the VACE VAE ($4\times$ temporal, $8\times$ spatial compression) this maps non-linearly to latent frames, so all masks are constructed in pixel space and then compressed. This same sparse construction is used both inside the on-policy rollout during training and at inference; the teacher instead receives the dense per-frame control.

7.1.5 Evaluation Metrics

All methods are evaluated independently on the 600-clip test set. Scores are computed per clip and averaged over the dataset.

Control fidelity (per modality). For each control modality we *re-extract* the control signal from the generated video with the corresponding annotator and compare it to the control that was fed in.

- **Pose.** We run DWPose (Yang et al., 2023) on the generated video and use the 18 body joints, in normalized $[0, 1]$ image coordinates. A joint is counted only when both the GT and predicted confidence exceed 0.3; frames with fewer than 3 valid joints are skipped. We report *MPJPE*

(mean per-joint ℓ_2 error in normalized coordinates) and *PCK@τ* for $\tau \in \{0.2, 0.1\}$, where a joint is correct if its error is below $\tau \cdot s$ and s is the per-frame bounding-box size $s = \max(\Delta x, \Delta y)$ of the GT skeleton. Metrics are reported over *all* frames and over the *keyframe* subset separately.

- **Depth.** We re-extract depth from the generated frames with DPT-Hybrid (MiDaS) (Ranftl et al., 2020) and compare against the dense depth control (same annotator). Since these are *relative* depths, we remove the scale/shift ambiguity with a closed-form least-squares affine alignment $a \cdot \hat{d} + b \approx d$ and report the affine-invariant *SI-RMSE* (primary), the raw RMSE, and the Pearson correlation.
- **Scribble.** We re-extract neural line-art from the generated frames with the VACE scribble annotator and compare against the dense line-art control. Line pixels are binarized as deviations from the per-frame median ($> 40/255$), and we report a tolerant boundary *F1* (precision/recall) with a dilation tolerance of 2 pixels.

Distributional quality. Independent of control adherence, we measure perceptual and temporal realism against the GT distribution: *FID* (clean-fid, InceptionV3 features, 8 frames per clip) captures per-frame appearance, and *FVD* (I3D pretrained on Kinetics-400, 16 frames linspace-sampled per clip) captures spatio-temporal consistency.

7.1.6 Implementation and Hyperparameters

All methods fine-tune the same Wan2.1-VACE-1.3B (Jiang et al., 2025) base model with LoRA (Hu et al., 2021) adapters (the base weights are frozen), so differences are attributable to the training objective alone. LoRA (rank 32, $\alpha = 32$) is applied to the query/key/value/output projections of both self- and cross-attention and the two feed-forward linears within every VACE block. We train with AdamW (lr $= 10^{-4}$, $\beta = (0.9, 0.999)$, no weight decay) for 2000 steps on 8 GPUs (DDP, per-GPU batch 1), under bf16 mixed precision with gradient checkpointing. Inference and the on-policy rollout use the UniPC solver (Zhao et al., 2023) on a 50-step schedule (`shift`$= 16$). All models are distilled and deployed at classifier-free guidance scale $\gamma = 5$. Videos are 81 frames at 480×832. Table 5 lists the full configuration.

Off-policy distillation baseline. The off-policy baseline uses the same teacher–student prediction-matching loss as OPD but evaluates it on standard data-derived diffusion states. Given a ground-truth video latent $\mathbf{x}_0$, we sample t and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and construct $\mathbf{x}_t^{\text{off}} = \alpha_t \mathbf{x}_0 + \sigma_t \boldsymbol{\epsilon}$. While SFT matches the student prediction at this state to the ground-truth velocity target, the off-policy baseline replaces that target with the teacher prediction under dense control. The student is evaluated at the same state under sparse control. It therefore differs from OPD only in the state distribution, using noised ground-truth latents instead of student-visited states.

7.2 Ablation Studies

This section reports the full result tables for the two ablations summarized in Section 4.3. Both use the pose-only training protocol, so the numbers are not directly comparable to the jointly trained models in Table 2.

Effect of the direction-matching weight. Table 6 studies the direction-matching weight λ in PDM. When $\lambda = 0$, PDM reduces to positive-only matching. We find that $\lambda = 1$ provides the best overall control fidelity, including the lowest keyframe MPJPE and the highest PCK@0.1. Larger values improve some distributional-quality metrics but weaken control accuracy. We therefore use $\lambda = 1$ as the default.

Effect of the supervision horizon. Table 7 studies the number of supervised states K in the dense-to-sparse video application. Increasing K raises training cost from 23 seconds per step at $K = 1$ to 790 seconds under full-trajectory supervision ($K = 50$). Among the evaluated settings, $K = 8$ provides the strongest control fidelity while remaining substantially more efficient than full-trajectory supervision. We therefore use $K = 8$ for the main video experiments.

Table 6: **Effect of the conditional-direction weight λ in the pose-only ablation setting. Bold** and underlined numbers denote the best and second-best results, respectively.

	MPJPE ↓		PCK@0.2 ↑		PCK@0.1 ↑		Quality ↓	
Method	All	Key	All	Key	All	Key	FID	FVD
PDM ($\lambda = 0$)	**4.24**	2.81	**92.71**	96.24	81.80	91.05	13.97	59.72
PDM ($\lambda = 1$)	4.27	**2.75**	**92.71**	**96.38**	**82.12**	**91.66**	13.83	58.65
PDM ($\lambda = 3$)	4.36	2.95	92.08	95.77	81.25	90.39	13.33	61.79
PDM ($\lambda = 5$)	4.40	3.00	92.27	95.96	80.89	90.75	12.97	**56.85**
PDM ($\lambda = 10$)	4.49	3.07	91.60	95.35	80.21	89.93	**12.96**	56.91

Table 7: **Effect of the OPD supervision horizon K in the pose-only ablation setting.** Wall-clock training time is reported in seconds per optimization step. **Bold** and underlined numbers denote the best and second-best results, respectively.

	MPJPE ↓		PCK@0.2 ↑		PCK@0.1 ↑		Quality ↓		Time ↓
Method	All	Key	All	Key	All	Key	FID	FVD	s/step
PDM ($K = 1$)	4.33	2.98	92.53	95.68	81.19	90.57	13.70	**54.41**	∼**23**
PDM ($K = 2$)	4.45	3.11	92.18	95.57	80.78	90.51	13.89	60.63	∼38
PDM ($K = 4$)	**4.27**	2.86	92.59	95.98	81.71	91.13	**13.58**	57.08	∼69
PDM ($K = 8$)	**4.27**	**2.75**	**92.71**	**96.38**	**82.12**	**91.66**	13.83	58.65	∼132
PDM ($K = 50$)	4.65	3.14	91.34	95.25	80.81	90.09	13.91	61.95	∼790

7.3 Extended Related Work

7.3.1 Diffusion Model Distillation

Although our goal is not few-step generation, prior diffusion distillation provides the closest context for understanding how teacher supervision and classifier-free guidance are typically transferred to a student. Most existing methods focus on reducing the number of denoising steps required at inference. Progressive Distillation recursively compresses multiple teacher steps into a single student step (Salimans & Ho, 2022). Consistency-based methods, including Consistency Models (Song et al., 2023; Kim et al., 2024) and Latent Consistency Models (Luo et al., 2023), train a model to produce consistent outputs across states along the same probability-flow trajectory, enabling one- or few-step generation. Distribution Matching Distillation (DMD) matches the student distribution to the teacher through their score functions (Yin et al., 2024b), while other approaches further combine diffusion supervision with adversarial training (Sauer et al., 2024; Yin et al., 2024a).

Many of these methods also distill classifier-free guidance into the student. Rather than retaining separate positive and negative branches at inference, they use the CFG-composed teacher prediction as the distillation target, thereby baking the effect of guidance into a single student prediction (Luo et al., 2023; Yin et al., 2024b; Wu et al., 2026; Huang et al., 2025). This design improves sampling efficiency by avoiding two model evaluations for CFG and, more broadly, transfers the strongly guided teacher distribution into a compact few-step generator.

Our setting differs in both objective and purpose. We do not primarily seek to compress a multi-step guided teacher into a CFG-free few-step student. Instead, we study diffusion OPD in which the teacher and student each construct their own guided prediction from positive and negative branches. Matching only their final guided predictions is under-identified at the branch level. This ambiguity becomes harmful when the teacher's negative branch contains privileged information unavailable to the student, preventing the two branch errors from being reduced jointly.

7.3.2 On-Policy Distillation for Diffusion Models

Recent work extends on-policy distillation to diffusion and flow models by querying the teacher on states sampled from the current student, and to other modalities (Yuan et al., 2026). D-OPSD performs on-policy self-distillation using privileged context, allowing a student to learn from a stronger teacher evaluated along the student's own denoising trajectory (Jiang et al., 2026). Flow-OPD distills task-specialized flow-matching teachers into a unified student through on-policy sampling and dense per-step supervision (Fang et al., 2026). DiffusionOPD formulates this process through transition-

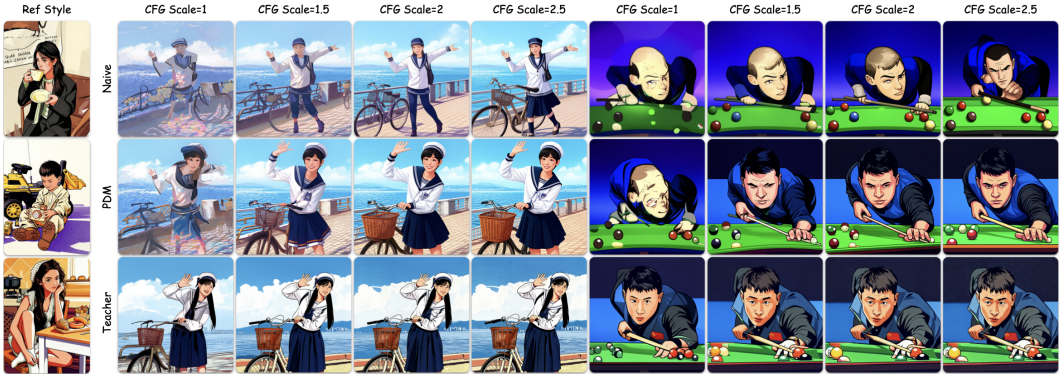


Figure 5: **Additional qualitative results for reference-conditioned distillation.** The left column shows reference-style exemplars, followed by outputs for two held-out prompts across inference guidance scales. Text-only students are trained at $\gamma_{\text{train}} = 2$ using naive matching or PDM, and the reference-conditioned teacher is shown for comparison. PDM more consistently follows the teacher and preserves the reference-supplied style, whereas naive matching exhibits stronger appearance drift and visual artifacts, particularly at $\gamma = 1$.

level KL divergence and shows that it reduces to prediction matching under standard diffusion transitions (Li et al., 2026b). Concurrently, DanceOPD treats generative capabilities as velocity fields and internalizes operator-defined fields such as CFG into a single student prediction through on-policy field matching (Zhou et al., 2026). Unlike the branch-retaining setting studied here, such a student no longer exposes separate positive and negative predictions at inference.

Despite differences in derivation and optimization, these methods share the same central principle: the student generates the trajectory, while the teacher provides dense local supervision at student-visited states. Their studied configurations, however, either collapse CFG into a single student prediction or retain separate branches while sharing the same negative conditioning between teacher and student. In the latter case, the negative-branch error need not be exactly zero, but shared conditioning allows the two branch errors to be reduced jointly. Existing work therefore does not examine how privileged information in the teacher's negative branch changes the optimization of a branch-retaining student.

Our work studies this missing regime. We show that CFG-composed matching permits cross-branch error compensation and identify when this ambiguity becomes harmful: under shared negative conditioning, both branch errors can decrease jointly, whereas privileged negative conditioning can induce antagonistic branch-error dynamics. We term this failure mode NBA. We further introduce branch-aware OPD objectives, with PDM as the primary formulation, to prevent such error compensation and improve robustness across inference guidance scales.

7.4 More Qualitative Results

Figures 5 and 6 provide additional qualitative evidence of distillation-induced CFG sensitivity under privileged conditioning. In reference-conditioned distillation, PDM more closely follows the teacher and preserves the reference-supplied style across guidance scales, whereas naive matching exhibits appearance drift and visual artifacts. In dense-to-sparse video control, PDM likewise maintains subject appearance and controlled structure as the inference scale changes, while naive matching degrades substantially. These results further demonstrate the excess guidance sensitivity associated with NBA.

Figures 7, 8 and 9 extend the qualitative comparison to pose, depth and scribble control under the default guidance scale. Across all three modalities, PDM follows the sparse structural conditions more faithfully while maintaining coherent visual content. These results show that the benefit of branch-aware supervision is consistent across different forms of conditioning asymmetry.

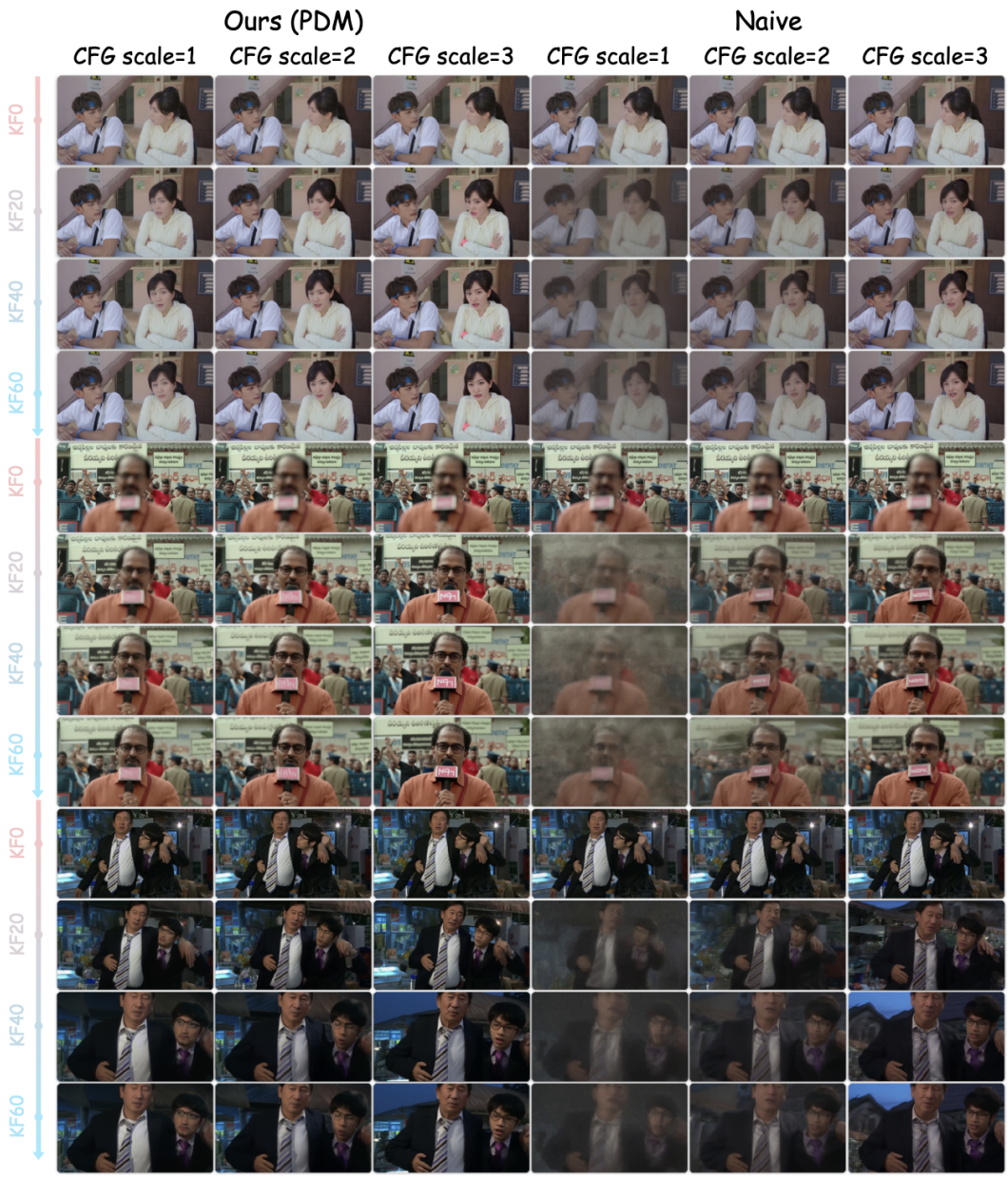


Figure 6: **Additional qualitative visualization of NBA in dense-to-sparse video control.** Models are trained at $\gamma_{\text{train}} = 5$ and evaluated at lower inference guidance scales. Naive CFG-based OPD exhibits severe blurring and structural degradation as γ_{test} moves away from the training configuration, revealing its sensitivity to a specific CFG composition. PDM consistently preserves subject appearance and video structure across guidance scales by separately matching the positive prediction and CFG conditional direction.



Figure 7: **Qualitative comparison on dense-to-sparse pose control distillation under the default guidance scale $\gamma = 5$.** The teacher receives dense pose guidance, while the student is conditioned only on sparse keyframes. PDM achieves more consistent structure and visual quality compared with other distillation objectives.



Figure 8: **Qualitative comparison on dense-to-sparse depth control distillation under the default guidance scale $\gamma = 5$.** The teacher receives dense depth guidance, while the student is conditioned only on sparse keyframes. PDM achieves more consistent structure and visual quality compared with other distillation objectives.



Figure 9: **Qualitative results on dense-to-sparse scribble control distillation with $\gamma = 5$.** Compared with other distillation objectives, Positive–Direction Matching better preserves the scribble-guided structure and generates more consistent visual results under sparse control supervision.