RACE: REWARD-GUIDED ADAPTIVE CANDIDATE EXPLORATION FOR SAMPLE-EFFICIENT DIFFUSION FINE-TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Reward-guided diffusion fine-tuning requires informative candidates under a limited scoring budget. Independent sampling provides coverage, while a shared scalar perturbation cannot adapt exploration across latent coordinates. We introduce *Reward-Guided Adaptive Candidate Exploration* (RACE), which learns a state-conditioned, entry-wise perturbation map to construct additional candidates around selected generated trajectories while retaining independent base samples. When paired training images are available, RACE$^+$ uses deterministic inversion to provide reference-derived exploration centers within the same candidate budget. Both variants jointly optimize the generator and perturbation predictor through one reward-weighted objective. Our analysis derives sufficient conditions for improved expected candidate reward, separating parent-selection benefits and perturbation effects and quantifying the roles of reference quality and inversion error. Experiments cover natural-image generation, including a multi-topic COCO subset, remote sensing, and medical imaging. Both variants achieve higher mean scores than DiffusionNFT across all five COCO evaluators. On brain MRI, RACE$^+$ achieves a joint score of 0.836 ± 0.012, compared with 0.793 ± 0.009 for DiffusionNFT. These findings support learning candidate construction as a means of improving diffusion fine-tuning under matched scoring budgets.

1 INTRODUCTION

Reward-guided diffusion fine-tuning adapts pretrained generators to task-specific criteria, from visual preferences in natural images to scene structure in remote sensing and anatomical consistency in medical imaging. Reward-guided objectives specify how scored samples update the generator (Black et al., 2024; Clark et al., 2024; Liu et al., 2026; Zheng et al., 2026), but each update can exploit only the candidates it receives. Since every candidate requires generation and evaluation, their construction matters when reward evaluations are limited. We study whether learning candidate construction improves held-out generation under a fixed scored-candidate budget.

Two aspects of candidate construction are central: the geometry of local exploration and its starting point. Independent draws provide coverage, but do not reuse promising intermediate states for local refinement. A shared scalar perturbation treats all latent coordinates equally, although useful variation can depend on the sample, channel, and spatial location. Exploration centered only on generated samples also lacks direct access to the structural information in paired training images. These observations motivate state-conditioned perturbations together with reference-derived exploration centers.

We introduce *Reward-Guided Adaptive Candidate Exploration* (RACE), which combines independent base samples with additional candidates constructed around selected high-reward trajectories. A predictor uses the parent state and noise level to produce entry-wise mixing coefficients, shaping local perturbations across latent coordinates. All base samples are retained, and the complete candidate pool stays within a fixed scoring budget. The generator and predictor are jointly trained through one reward-weighted objective, making candidate construction part of fine-tuning.

With paired training images, RACE$^+$ uses deterministic inversion to obtain reference-derived intermediate states. These states provide structural exploration centers for a portion of the existing

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.

child budget and share the same perturbation predictor and objective as generated-parent candidates. Reference guidance therefore enters through candidate construction without a separate reconstruction loss. Both inversion and the predictor operate during training; held-out generation starts from fresh noise and the task condition.

Experiments cover natural-image generation, including a broader multi-topic COCO subset (Lin et al., 2014), remote sensing on AID (Xia et al., 2017), and medical imaging. Under matched scoring budgets within each benchmark, both variants improve all five COCO evaluation means over DiffusionNFT. The MRI component comparison gives joint scores of 0.818 ± 0.008 for shared scalar perturbations, 0.828 ± 0.008 for RACE, and 0.836 ± 0.012 for RACE$^+$ (Table 5). These comparisons support the utility of the proposed constructions; Figure 1 summarizes representative results on Architecture & Interior, brain MRI, and AID.

Our contributions are threefold: (i) **Learnable candidate exploration.** RACE jointly learns a state-conditioned, entry-wise perturbation map and the generator while retaining independent samples within a fixed candidate budget. (ii) **Reference-guided construction.** RACE$^+$ incorporates paired-image structure through inversion-derived exploration centers, sharing the predictor and objective with generated-parent exploration. (iii) **Conditional reward analysis.** We derive sufficient conditions for expected candidate-reward improvement, separating parent-selection and perturbation effects and relating the benefit of reference guidance to reference reward, inversion error, and perturbation size.

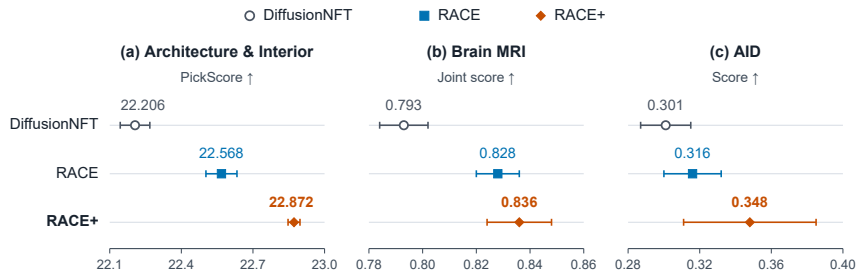

Figure 1: **Matched-budget gains over DiffusionNFT.** RACE learns where to perturb each trajectory; RACE$^+$ additionally relocates part of the same candidate budget around an inverted reference.

2 Background and Related Work

2.1 Flow Matching and Trajectory Representations

Diffusion and flow models learn a vector field connecting noise and data (Ho et al., 2020; Song et al., 2020b; Lipman et al., 2022). We use the rectified interpolation underlying the backbone family (Liu et al., 2022; Esser et al., 2024):

$$x_t = (1-t)x_0 + t\epsilon, \qquad u = \epsilon - x_0, \qquad \epsilon \sim \mathcal{N}(0, \mathbf{I}). \tag{1}$$

Here x denotes a VAE latent (Rombach et al., 2022), and $t \in [0,1]$ is a continuous noise level. The learned field satisfies $dx_t/dt = v_\theta(x_t, t, c)$. Let $F_t(x, c)$ denote continuation of the frozen collection field v_{old} from (x, t) to noise level zero. Its endpoint score is $S_t(x, c) = R(D_{\text{VAE}}(F_t(x, c)), c)$. The interpolation in Eq. (1) specifies a training construction; a numerically integrated model trajectory need not follow that straight line. When a task reward also uses a fixed evaluation reference, that argument is suppressed in R and S_t.

2.2 Reward-Guided Diffusion Fine-Tuning

DPOK and DDPO formulate denoising as sequential control (Fan et al., 2023; Black et al., 2024). DRaFT and AlignProp differentiate rewards through sampling, while Diffusion-DPO adapts preference optimization to diffusion models (Clark et al., 2024; Prabhudesai et al., 2023; Wallace et al., 2024). Flow-GRPO uses group-relative reinforcement learning, with DSPO and V-GRPO exploring score-space and variational alternatives (Liu et al., 2026; Honavar, 2025; Tang et al., 2026).

DiffusionNFT (Zheng et al., 2026) instead uses a reward-weighted forward-process objective,

$$\min_\theta \; \mathcal{L}_{\text{NFT}}(\theta) = \mathbb{E}\big[r\|v_\theta^+(x_t,t,c) - u\|^2 + (1-r)\|v_\theta^-(x_t,t,c) - u\|^2\big], \tag{2}$$

where $v_\theta^\pm = (1 \mp \beta)v_{\text{old}} \pm \beta v_\theta$ and $r \in [0, 1]$. Our framework retains this weighting form and makes candidate construction trainable. The construction and scored endpoints are distinguished explicitly in Section 3.2; the shared objective is analyzed under the resulting joint sample distribution.

2.3 Candidate sampling, references, and efficient generation

Feedback-guided sampling is complementary to the update rule. HERO refines diffusion initializations using online human feedback (Hiranaka et al., 2025); trajectory branching, mixed ODE–SDE rollouts, and coefficient-preserving sampling also modify candidate collection (Hu et al., 2025; Li et al., 2025; Wang & Yu, 2025). Inference-time steering instead allocates computation among reward-weighted particles (Singhal et al., 2025). RACE focuses on a state-conditioned, entry-wise construction at intermediate latent states and its joint optimization during fine-tuning. This changes the sampling distribution while LoRA controls the number of adapted generator parameters (Hu et al., 2022).

DDIM, SDEdit, Prompt-to-Prompt, and Null-text Inversion demonstrate how diffusion trajectories can preserve image content (Song et al., 2020a; Meng et al., 2022; Hertz et al., 2022; Mokady et al., 2023). RACE$^+$ uses an inverted reference as a source of training candidates, reallocating existing child slots rather than introducing a separate reconstruction objective.

3 Reference-guided adaptive candidate exploration

RACE$^+$ addresses a collection bottleneck: an update can learn only from the candidates that its current generator proposes. Independent draws provide coverage, but spend no extra effort around promising outputs; reference-only reconstruction supplies structure, but offers limited variation. We combine an independent base pool with learned local exploration around generated and reference trajectories. One perturbation predictor controls both child sources, and one reward-weighted objective updates the predictor and generator. Figure 2 summarizes this collection–learning cycle.

3.1 Coverage under a fixed scoring budget

For condition c, a frozen collection model generates and scores B independent bases. The top-P bases supply parents for K_g generated children; paired training references supply K_r reference children. Every base remains in the training pool:

$$\mathcal{P} = \mathcal{B} \cup \mathcal{C}_g \cup \mathcal{C}_r, \qquad K = B + K_g + K_r. \tag{3}$$

RACE sets $K_r = 0$ and assigns all child slots to generated parents. RACE$^+$ reallocates K_r of those slots to references. Retaining the bases preserves an independent-generation component, while children reuse informative starting states. This is a coverage–refinement trade-off: $B < K$ still provides fewer independent draws than a K-draw baseline. Base scores are cached, and each child is scored once. Thus K includes the parent-selection scores, and no additional top-K filtering discards candidates from the objective.

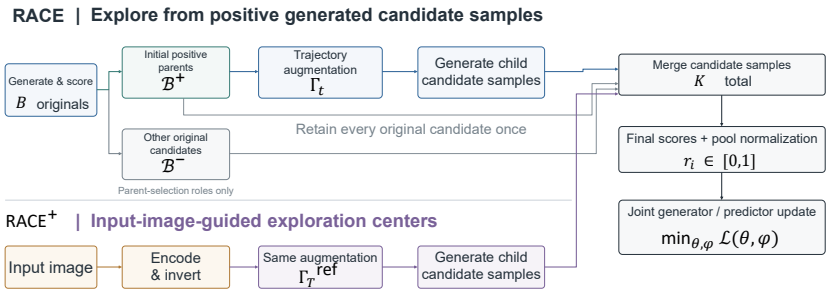

Figure 2: **RACE framework.** Bases provide training samples and selected parents. Generated and reference-derived states share the child construction; all candidates enter one generator–predictor objective.

3.2 Learnable trajectory-local exploration

A single scalar perturbation allocates equal conditional variance to every latent coordinate. This can waste variation in sensitive coordinates while underexploring useful ones. We instead predict

an entry-wise map from the parent state and noise level. For a saved parent state x_t and its clean construction endpoint x_0, recover $\epsilon_t = [x_t - (1-t)x_0]/t$, with $t > 0$. This algebraically reproduces the saved state, including on a curved integrated trajectory; ϵ_t need not be Gaussian. With independent $\tilde{\epsilon}_k \sim \mathcal{N}(0, \mathbf{I})$, define

$$\Gamma_t = \Gamma_\phi(x_t, t) = \gamma_0 + \delta_\Gamma \tanh f_\phi(x_t, t), \qquad \gamma_0 > \delta_\Gamma > 0,$$
$$\epsilon_{\Gamma,k} = \frac{\epsilon_t + \Gamma_t \odot \tilde{\epsilon}_k}{\sqrt{1 + \Gamma_t^{\odot 2}}}. \tag{4}$$

All operations are entry-wise. The bounded map controls perturbation moments, rather than imposing a hard radius on Gaussian draws. The denominator normalizes the mixing coefficients; it does not make a state-conditioned mixture marginally standard Gaussian (Appendix A.1). A constant scalar map recovers

$$\epsilon_{\gamma,k} = (\epsilon_t + \gamma\tilde{\epsilon}_k)/\sqrt{1+\gamma^2}. \tag{5}$$

The child state, construction target, and separately scored endpoint are

$$x_{t,k} = (1-t)x_0 + t\epsilon_{\Gamma,k}, \quad u_k = \epsilon_{\Gamma,k} - x_0, \quad \widehat{x}_{0,k} = F_t(x_{t,k}, c). \tag{6}$$

Here x_0 defines the target, while $\widehat{x}_{0,k}$ determines the child's own reward. This distinction is essential to the shared-loss analysis. The same 19,024-parameter predictor handles both sources (Appendix C.2). Collection levels t are chosen by the collection schedule; the time-conditioned network alone does not establish performance at levels that were never sampled.

For a fixed parent context, $\Delta = x_{t,k} - x_t$ has mean $m = t(A - \mathbf{I})\epsilon_t$ and covariance $\Sigma = t^2 B_\Gamma^2$, where $A = \mathrm{diag}((1+\Gamma_t^{\odot 2})^{-1/2})$ and $B_\Gamma = \mathrm{diag}(\Gamma_t \odot (1+\Gamma_t^{\odot 2})^{-1/2})$. The map therefore changes both the exploration center and the diagonal covariance. Figure 3 separates this state-space geometry from the reward distribution of completed children.

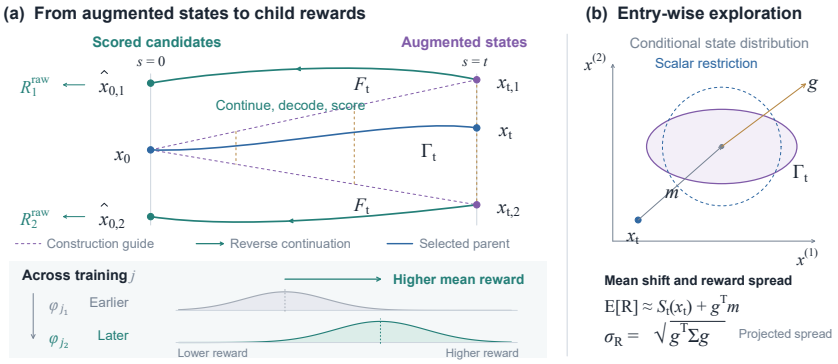

Figure 3: **Trajectory-conditioned exploration.** (a) Construction guides connect x_0 to augmented states $x_{t,k}$, while continuation F_t produces separately scored endpoints $\widehat{x}_{0,k}$. The lower panel schematically compares child-reward distributions under ϕ_{j_1} and ϕ_{j_2}, with $j_1 < j_2$, fixed parent contexts, and a frozen continuation field, in the improvement regime of Theorem 3. (b) The displacement mean m, covariance Σ, and reward gradient g determine the local mean shift and projected spread. The scalar circle is shown at a common illustrative center.

An exploration-policy view. For context $z = (x_t, t, c, x_0)$, the map and fresh Gaussian direction induce an exploration policy $\pi_\phi(\Delta \mid z)$ with return $S_t(x_t + \Delta, c)$. Let J_{RACE} be the expected raw reward of a uniformly chosen member of the final K-candidate pool. Define J_{base} using K independent generations from the same frozen generator *without candidate exploration*. Write $\mu_b = J_{\mathrm{base}}$, μ_+ for the mean unperturbed continuation reward of selected parents, $\delta_{\mathrm{sel}} = \mu_+ - \mu_b$, and $w = (K-B)/K$.

Theorem 1 (Reward benefit of candidate exploration)**.** *Fix the generator, evaluator, condition distribution, and parent/time-selection rule. Assume uniform child allocation over selected parents, consistent rewards and continuation to the original parent endpoints, integrable proposal moments,*

and a Hessian bound H_z for S_t on the proposal support. With $g_z = \nabla_x S_t(x_t, c)$ and $G_\phi = \mathbb{E}[S_t(x_t + \Delta, c) - S_t(x_t, c)]$,

$$J_{\mathrm{RACE}} - J_{\mathrm{base}} = w(\delta_{\mathrm{sel}} + G_\phi) \geq w\left[\delta_{\mathrm{sel}} + \mathbb{E}\left(g_z^\top m - \frac{H_z D}{2}\right)\right]. \tag{7}$$

A positive bracket implies strict improvement. Moreover, define $V_\phi(z) = \mathbb{E}_{\pi_\phi} S_t(x_t + \Delta, c)$ and $A_\phi(z, \Delta) = S_t(x_t + \Delta, c) - V_\phi(z)$. For an updated exploration policy under the same context law,

$$J_{\mathrm{RACE}}(\phi') - J_{\mathrm{RACE}}(\phi) = w\mathbb{E}_{z,\Delta\sim\pi_{\phi'}} A_\phi(z, \Delta). \tag{8}$$

Thus expected advantage of at least $\eta > 0$ increases expected candidate reward by at least $w\eta$.

Under these assumptions, top-parent selection gives $\delta_{\mathrm{sel}} \geq 0$. Equation (7) separates this selection benefit from perturbation drift and curvature cost. The comparison concerns candidate collection and does not depend on a particular downstream training objective. It does not, by itself, establish reward improvement after every generator update. Appendix A.2 gives the proof and the nonuniform-allocation extension.

3.3 Reference-guided candidate construction

A generated parent can only contribute structure already present in the collected outputs. A paired image offers a different starting point. To explore around it using the collection field, encode and deterministically invert it:

$$x_0^{\mathrm{ref}} = E_{\mathrm{VAE}}(I^{\mathrm{ref}}), \qquad x_T^{\mathrm{ref}} = x_0^{\mathrm{ref}} + \int_0^T v_{\mathrm{old}}(x_s^{\mathrm{ref}}, s, c)\, ds. \tag{9}$$

Here $T \in (0, 1]$ is the reference branch's continuous inversion level, not a solver-step count or a training round. Recover $\epsilon_T^{\mathrm{ref}} = [x_T^{\mathrm{ref}} - (1-T)x_0^{\mathrm{ref}}]/T$ and apply the same predictor and construction:

$$\begin{aligned} \epsilon_{\Gamma,k}^{\mathrm{ref}} &= \frac{\epsilon_T^{\mathrm{ref}} + \Gamma_T^{\mathrm{ref}} \odot \tilde{\epsilon}_k}{\sqrt{1 + (\Gamma_T^{\mathrm{ref}})^{\odot 2}}}, \qquad \Gamma_T^{\mathrm{ref}} = \Gamma_\phi(x_T^{\mathrm{ref}}, T), \\ x_{T,k}^{\mathrm{ref}} &= (1-T)x_0^{\mathrm{ref}} + T\epsilon_{\Gamma,k}^{\mathrm{ref}}, \qquad u_k^{\mathrm{ref}} = \epsilon_{\Gamma,k}^{\mathrm{ref}} - x_0^{\mathrm{ref}}. \end{aligned} \tag{10}$$

Every completed reference child receives the same task reward as the other candidates. References change the construction context and target while keeping the objective shared. In the stated validation/test protocol, generation starts from fresh Gaussian noise and condition c; inversion and the predictor are training-time components. A held-out image supplied to a fidelity evaluator is not a generator input.

When does reference structure help? Let $D_r = \mathbb{E}[\|x_{T,k}^{\mathrm{ref}} - x_T^{\mathrm{ref}}\|^2 \mid I^{\mathrm{ref}}, c, T]$ and $e_{\mathrm{rt}} = \|F_T(x_T^{\mathrm{ref}}, c) - x_0^{\mathrm{ref}}\|$. If F_T is L_F-Lipschitz over the required support, the endpoint-defined target $\widehat{u}_k^{\mathrm{ref}} = (x_{T,k}^{\mathrm{ref}} - F_T(x_{T,k}^{\mathrm{ref}}, c))/T$ satisfies

$$\mathbb{E}\|\widehat{u}_k^{\mathrm{ref}} - u_k^{\mathrm{ref}}\| \leq (e_{\mathrm{rt}} + L_F\sqrt{D_r})/T. \tag{11}$$

Thus useful reference exploration requires accurate round trips and controlled perturbation. Locality is quantitative, not an automatic consequence of using a real image.

Let $\mathcal{R}(y, c) = R(D_{\mathrm{VAE}}(y), c)$, and let J be the expected mean raw reward of a complete K-candidate pool.

Theorem 2 (Reference-guided candidate reward)**.** *At fixed generator and predictor snapshots, replace K_r generated children with reference children, leaving all other candidates unchanged in distribution. Let J_g, J_r be the means of the replaced and reference children, and $J_{\mathrm{in}} = \mathbb{E}\mathcal{R}(x_0^{\mathrm{ref}}, c)$. If F_T and $\mathcal{R}$ are L_F- and L_R-Lipschitz over the required support, with integrable bounds, then*

$$\begin{aligned} J_{\mathrm{RACE}^+} - J_{\mathrm{RACE}} &= \frac{K_r}{K}(J_r - J_g) \\ &\geq \frac{K_r}{K}\left[J_{\mathrm{in}} - \mathbb{E}\{L_R(e_{\mathrm{rt}} + L_F\sqrt{D_r})\} - J_g\right]. \end{aligned} \tag{12}$$

A positive bracket means that the reference advantage exceeds construction penalties. It improves the pool mean, without requiring every child to beat its reference. The result compares collection distributions, not separately trained final generators. Appendix A.3 proves the bound and states the additional conditions for threshold-discovery gains.

3.4 One objective for generator and predictor

After collection, normalize cached base scores and new child scores together:

$$r_i = \frac{1}{2} + \frac{1}{2}\,\mathrm{clip}_{[-1,1]}\left(\frac{r_i^{\mathrm{raw}} - \mu_{\mathcal{P}}}{Z_{\mathcal{P}}}\right), \quad \mu_{\mathcal{P}} = K^{-1}\sum_i r_i^{\mathrm{raw}}, \quad Z_{\mathcal{P}} > 0. \tag{13}$$

Above-mean and below-mean candidates receive $r_i > 1/2$ and $r_i < 1/2$; constant-score groups receive $1/2$. These are continuous weights for all K samples, not fixed-size positive/negative partitions. Pool-wide normalization lets both child sources compete with bases under the same relative reward scale.

Bases use fresh forward-process pairs from their scored endpoints. Children retain their stored construction endpoints and random directions, rebuilding Eqs. (6) and (10) with the live predictor. For pairs $(x_i(\phi), u_i(\phi))$, define $d_i = v_\theta(x_i, t_i, c) - v_{\mathrm{old}}(x_i, t_i, c)$ and

$$v_i^{\pm} = v_{\mathrm{old}}(x_i, t_i, c) \pm \beta d_i, \qquad \beta > 0. \tag{14}$$

Both parameter blocks minimize one objective:

$$\min_{\theta,\phi}\ \mathcal{L}(\theta,\phi) = \frac{1}{K}\sum_i \left[r_i\|v_i^+ - u_i\|^2 + (1-r_i)\|v_i^- - u_i\|^2\right]. \tag{15}$$

Using the same reward weighting couples candidate construction to generator training without adding a separately weighted reference criterion. With $a_i = u_i - v_{\mathrm{old}}(x_i, t_i, c)$, each term expands as

$$\ell_i = \beta^2\|d_i\|^2 - 2\beta(2r_i - 1)d_i^\top a_i + \|a_i\|^2. \tag{16}$$

Here $a_i = (\bar{x}_{0,i} - x_{0,i})/t_i$, where $\bar{x}_{0,i} = x_i - t_i v_{\mathrm{old}}(x_i, t_i, c)$ is a one-step estimate, not the fully continued endpoint. The cross term supplies reward-dependent feedback through both d_i and a_i; the final term also depends on ϕ and cannot be discarded as a constant.

Collection rewards, ranking, parent context, and auxiliary draws remain fixed during optimization. Child states and targets remain differentiable:

$$\nabla_\phi \ell_i = (D_\phi x_i)^\top \nabla_{x_i}\ell_i + (D_\phi u_i)^\top \nabla_{u_i}\ell_i. \tag{17}$$

Both fields retain input Jacobians; among the two vector fields, only the live generator's parameters are updated. Reverse continuation and reward evaluation are outside this graph. Collection copies refresh between rounds. Whenever both fields are identical, $d_i = D_\phi d_i = 0$ and that predictor gradient is reward-independent. Later reward-dependent gradients need not automatically align with endpoint reward. Appendix C.2 specifies the graph; Appendix A.5 analyzes its consequences.

Algorithm 1 RACE candidate collection and shared update

Require: Collection field and predictor, condition c, $K = B + K_g + K_r$, $P \le B$

1: Generate and score B originals; retain all and select the top-P parents.
2: Construct K_g augmented states using Eqs. (4)–(6); continue to clean endpoints.
3: If $K_r > 0$, invert the input image to level T; construct and continue K_r children by Eq. (10).
4: Merge K candidates; obtain final raw scores and recompute weights by Eq. (13).
5: Jointly update θ, ϕ through Eq. (15); soft-update collection models.

3.5 From perturbations to child-reward improvement

The desired collection statistic is the raw reward of *all* children. For a fixed context law ρ, continuation field, and evaluator, define $\mathcal{J}(\phi) = \mathbb{E}_{z\sim\rho,\tilde{\epsilon}} S_t(x_t + \Delta_\phi(z,\tilde{\epsilon}), c)$ with $z = (x_t, t, c, x_0)$. Under a local linear reward model, its conditional mean is $S_t(x_t, c) + g^\top m$, whereas $\sqrt{g^\top \Sigma g}$ controls spread. Widening the distribution alone does not increase this mean. Proposition 4 derives the exact drift–curvature sensitivity of expected reward to each learned coefficient, connecting the geometry in Figure 3(b) to the reward shift in (a).

Theorem 3 (Conditional mean-reward improvement)**.** *Let $\phi_{j+1} = \phi_j - \eta_j h_j$ be the effective predictor update from Eq.* (15)*, including optimizer preconditioning. Given the training history, set $\chi_j = -\langle \nabla_\phi \mathcal{J}(\phi_j), \mathbb{E}_j h_j \rangle$ and $V_j = \mathbb{E}_j \|h_j\|^2$. If $\mathcal{J}$ has L_J-Lipschitz gradient along the update segments and $V_j < \infty$, then*

$$\mathbb{E}_j[\mathcal{J}(\phi_{j+1})] - \mathcal{J}(\phi_j) \geq \eta_j \chi_j - \tfrac{1}{2} L_J \eta_j^2 V_j. \tag{18}$$

For $\chi_j > 0$, $L_J > 0$, and $0 < \eta_j < 2\chi_j/(L_J V_j)$, the expected mean increases.

The positive-alignment condition is additional to loss minimization. Appendix A.5 gives its explicit Gaussian reward-gradient identity, a counterexample to unconditional improvement, and corrections for changing collection models and copy lag. Figure 3(a) illustrates the conditional improvement regime, not a numerical learning curve. Appendix C.4 defines the online and fixed-context measurements needed to test this mechanism separately from held-out generator performance.

4 Experiments

We compare reference-guided RACE$^+$, generated-only RACE, and independent-candidate fine-tuning under matched per-prompt candidate budgets. Quantitative evaluation covers natural images, medical images, and remote sensing; the MRI component comparison examines successive method variants. Qualitative examples illustrate image properties and residual artifacts.

4.1 Evaluation setup

We use Stable Diffusion 3.5 Medium with LoRA (Esser et al., 2024; Hu et al., 2022). Architecture & Interior provides controlled natural-image development; MRI, CT, and AID test specialized domains. The larger experimental protocol uses a COCO image–caption subset (Lin et al., 2014): 1,200 training, 160 validation, and 240 test prompts.

Matched budgets within each benchmark. The smaller benchmarks use $K = 8$: DiffusionNFT generates eight independent candidates, RACE uses $4+4$, and RACE$^+$ uses $4+2+2$ base/generated-child/reference-child candidates. COCO uses $K = 24$: the corresponding allocations are $24 + 0 + 0$, $20 + 4 + 0$, and $20 + 2 + 2$. COCO therefore enlarges the independent base pool while keeping four child candidates. This tests the same construction with a smaller augmentation fraction. Within each benchmark, methods share K, including the initial parent-selection scores; the two benchmark settings are not a controlled K ablation (Table 7).

Scoring and reporting. Tables report evaluator scores before within-pool normalization, rather than the training weights r_i. Natural-image evaluation uses PickScore (Kirstain et al., 2023), CLIP similarity with OpenAI CLIP ViT-L/14 (Radford et al., 2021), HPSv2.1 (Wu et al., 2023), the improved aesthetic predictor (Schuhmann, 2022), and ImageReward (Xu et al., 2023). Specialized scores are domain-specific and are detailed in Appendix D.3. Means and standard deviations are computed across training seeds when repeated runs are available. Paired references supply training candidates for RACE$^+$ only; held-out generation follows the noise-initialized inference protocol in Section 3.3.

4.2 Natural-image quality and prompt-diversity scaling

Controlled topic. On Architecture & Interior, RACE improves over DiffusionNFT in PickScore, CLIP similarity, HPSv2.1, and Aesthetic Score, while ImageReward is slightly lower (0.728 versus 0.731). RACE$^+$ achieves the highest mean scores across all five evaluators, including a PickScore of 22.872 and an ImageReward of 0.782 (Table 1).

COCO scaling. On the multi-topic COCO benchmark, both RACE and RACE$^+$ achieve higher mean scores than DiffusionNFT across all five evaluators. RACE$^+$ obtains the highest reported means, reaching 22.641 PickScore, 0.273 CLIP similarity, 0.291 HPSv2.1, 5.573 Aesthetic Score, and 1.013 ImageReward (Table 1). These results extend the comparison beyond the controlled single-topic setting, with candidate budgets matched within each benchmark. The COCO configuration and checkpoint-selection protocol are described in Appendix D.2 and Table 10.

Table 1: **Natural-image generation.** Results on controlled Architecture & Interior ($K = 8$) and multi-topic COCO ($K = 24$). Fine-tuned results are mean $\pm$ standard deviation over three training seeds. Best and second-best results within each benchmark are shown in **bold** and underlined, respectively.

Method	PickScore ↑	CLIP similarity ↑	HPSv2.1 ↑	Aesthetic ↑	ImageReward ↑
Architecture & Interior ($K = 8$)					
SD3.5-M	21.269	0.2628	0.2626	5.520	0.685
DiffusionNFT	22.206 ± 0.062	0.266 ± 0.0019	0.268 ± 0.0045	5.533 ± 0.077	0.731 ± 0.051
RACE	22.568 ± 0.065	0.268 ± 0.0005	0.271 ± 0.0027	5.549 ± 0.034	0.728 ± 0.057
RACE$^+$	$\mathbf{22.872 \pm 0.025}$	$\mathbf{0.272 \pm 0.0028}$	$\mathbf{0.288 \pm 0.0022}$	$\mathbf{5.584 \pm 0.036}$	$\mathbf{0.782 \pm 0.051}$
COCO multi-topic ($K = 24$)					
SD3.5-M	22.495	0.268	0.281	5.416	0.968
DiffusionNFT	22.549 ± 0.077	0.270 ± 0.002	0.282 ± 0.004	5.483 ± 0.011	0.982 ± 0.022
RACE	22.603 ± 0.011	0.271 ± 0.000	0.286 ± 0.007	5.491 ± 0.107	0.990 ± 0.032
RACE$^+$	$\mathbf{22.641 \pm 0.032}$	$\mathbf{0.273 \pm 0.000}$	$\mathbf{0.291 \pm 0.002}$	$\mathbf{5.573 \pm 0.021}$	$\mathbf{1.013 \pm 0.021}$

4.3 Medical image generation

Brain MRI and CT evaluate the methods under structurally constrained conditional generation. Brain MRI contains 306 paired image rows from CN, MCI, and AD, with subject-disjoint splits of 214 training, 44 validation, and 48 test images. CT uses paired images with axial-position conditions. For both domains, the reported joint score has the form

$$R_{\text{joint}}(I, c, I^{\text{ref}}) = \frac{1}{2} R_{\text{align}}(I, c) + \frac{1}{2} R_{\text{fid}}(I, I^{\text{ref}}). \tag{19}$$

Brain MRI. The MRI joint metric combines axial-depth alignment and paired-reference fidelity. CN, MCI, and AD specify the conditioning categories, but diagnosis classification is not part of the reported alignment score. RACE improves the mean joint score from 0.793 to 0.828, while RACE$^+$ reaches 0.836, corresponding to relative gains of $+4.4\%$ and $+5.4\%$ over DiffusionNFT, respectively (Table 2).

Table 2: **Brain MRI generation.**

Method	Joint ↑	Δ vs. NFT ↑
SD3.5-M	0.734	-7.4%
DiffusionNFT	0.793 ± 0.009	0.0%
RACE	0.828 ± 0.008	$+4.4\%$
RACE$^+$	$\mathbf{0.836 \pm 0.012}$	$\mathbf{+5.4\%}$

CT. The CT joint metric combines axial-position alignment and paired-reference fidelity. Because the underlying component functions differ, CT values near 0.2 are neither 20% accuracy nor directly comparable to MRI values near 0.8. The CT mean increases from 0.212 to 0.225 for RACE and 0.259 for RACE$^+$, corresponding to relative gains of $+6.1\%$ and $+22.2\%$ over DiffusionNFT, respectively (Table 3). Different CT map initializations are documented in Table 9; the variant difference is not a reference-only ablation.

Table 3: **CT generation.**

Method	Joint ↑	Δ vs. NFT ↑
SD3.5-M	0.210	-0.9%
DiffusionNFT	0.212 ± 0.068	0.0%
RACE	0.225 ± 0.029	$+6.1\%$
RACE$^+$	$\mathbf{0.259 \pm 0.061}$	$\mathbf{+22.2\%}$

4.4 Remote-sensing generation

AID contains 10,000 aerial images across 30 scene categories (Xia et al., 2017). RACE and RACE$^+$ improve over DiffusionNFT by +5.0% and +15.6%, respectively (Table 4). All methods use the same AID evaluator and score transformation described in Appendix D.3.

Table 4: **AID remote-sensing generation.**

Method	Score ↑	Δ vs. NFT ↑
SD3.5-M	0.280	-7.0%
DiffusionNFT	0.301 ± 0.014	0.0%
RACE	0.316 ± 0.016	$+5.0\%$
RACE$^+$	$\mathbf{0.348 \pm 0.037}$	$\mathbf{+15.6\%}$

4.5 Qualitative comparison

The cross-domain examples in Figure 4 illustrate differences in anatomical appearance, scene layout, and natural-image composition, while also showing residual artifacts. Additional COCO examples in Figure 8 compare street scenes with light trails, downhill skiing, and horned mountain goats beside grass and snow. The reconstruction sequence in Figure 7 separately examines reference preservation and variation along the reference-guided construction route.

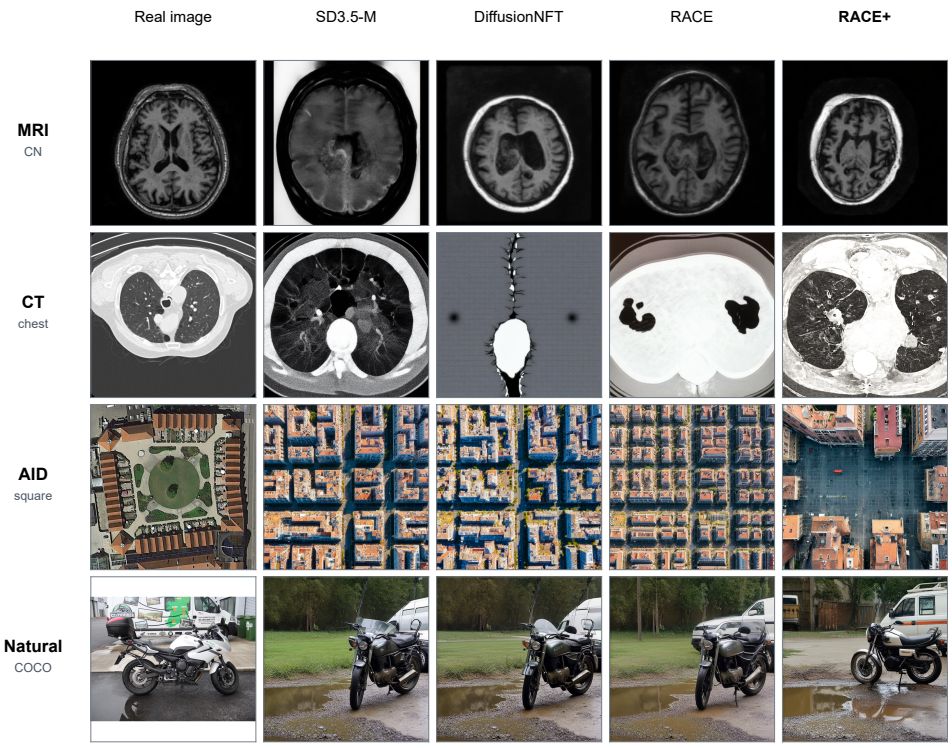

Figure 4: **Cross-domain qualitative comparison.** Rows show brain MRI, CT, AID remote sensing, and natural-image examples. Columns show a real image and outputs from SD3.5-M, DiffusionNFT, RACE, and RACE$^+$. References are not generator inputs during held-out inference. Full MRI examples and additional layouts appear in Appendix B.

4.6 Ablations and budget accounting

Table 5: **MRI component comparison at $K = 8$.** The scalar construction is the restricted map in Eq. (5); RACE uses a state-conditioned, entry-wise map; RACE$^+$ additionally uses reference-guided candidates. Means $\pm$ standard deviations are across training seeds.

Method	MRI Joint ↑
DiffusionNFT	0.793 ± 0.009
Shared scalar perturbation	0.818 ± 0.008
RACE	$\underline{0.828 \pm 0.008}$
RACE$^+$	$\mathbf{0.836 \pm 0.012}$

The scalar construction increases the mean by 0.025 over NFT; RACE increases it by another 0.010, and RACE$^+$ by 0.008 over RACE. These differences quantify successive method comparisons. These benchmark means do not establish a training-time increase in child reward. Appendix C.4 specifies the all-child statistic and fixed-context comparison.

5 Conclusion

RACE learns state-conditioned candidate exploration around generated trajectories, while RACE$^+$ incorporates inverted training references within the same candidate budget. Both jointly optimize the generator and perturbation predictor through one reward-weighted objective. Our analysis identifies sufficient conditions for candidate-reward improvement and quantifies the roles of perturbation geometry, reference quality, and inversion error. Experiments on natural images, remote sensing, and medical imaging support adaptive candidate construction for diffusion fine-tuning under matched candidate budgets.

AI USE STATEMENT

Generative AI tools were used to assist with literature search, consistency checking, mathematical derivation and proof checking, language editing, LaTeX restructuring, figure authoring, and small numerical validation scripts. Generative AI tools were not used to execute diffusion-model training or empirical benchmark evaluation. All AI-assisted manuscript content, derivations, implementation descriptions, and empirical claims were reviewed and verified by the authors, who take responsibility for the final content.

ETHICS STATEMENT

The medical-image experiments are research evaluations of synthetic image generation and are not validations for diagnosis or clinical use. Similarity to a reference image does not establish privacy protection or clinical correctness. No patient-level identifiers are reported in the manuscript.

REPRODUCIBILITY STATEMENT

Appendix A provides the theoretical derivations, Appendix C details candidate collection, predictor architecture, gradient paths, and inference procedures, and Appendix D documents the reported experimental configurations, evaluator definitions, and aggregation rules.

REFERENCES

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations*, volume 2024, pp. 4965–4987, 2024.

Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *SIAM review*, 60(2):223–311, 2018.

Kevin Clark, Paul Vicol, Kevin Swersky, and David J Fleet. Directly fine-tuning diffusion models on differentiable rewards, 2024. *URL https://arxiv. org/abs/2309.17400*, 2024.

Evan W. Damron, Mahmut S. Gokmen, Mitchell A. Klusty, Caroline N. Leach, Emily B. Collier, and V. K. Cody Bumgardner. Dale-ct: Depth-aware 2d slice encoders learn an anatomical world model of chest ct. *arXiv preprint arXiv:2606.07775*, 2026.

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.

Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Dpok: Reinforcement learning for fine-tuning text-to-image diffusion models. *Advances in neural information processing systems*, 36:79858–79885, 2023.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

Ayano Hiranaka, Shang-Fu Chen, Chieh-Hsin Lai, Dongjun Kim, Naoki Murata, Takashi Shibuya, WeiHsiang Liao, Shao-Hua Sun, and Yuki Mitsufuji. Hero: Human-feedback efficient reinforcement learning for online diffusion model finetuning. In *International Conference on Learning Representations*, volume 2025, pp. 54942–54971, 2025.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Vasant G Honavar. Dspo: Direct score preference optimization for diffusion model alignment. *ICLR*, 2025.

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. In *International Conference on Learning Representations*, 2022.

Zijing Hu, Fengda Zhang, Long Chen, Kun Kuang, Jiahui Li, Kaifeng Gao, Jun Xiao, Xin Wang, and Wenwu Zhu. Towards better alignment: Training diffusion models with reinforcement learning against sparse rewards. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23604–23614. IEEE, 2025.

Muhammad Uzair Khattak, Shahina Kunhimon, Muzammal Naseer, Salman Khan, and Fahad Shahbaz Khan. Unimed-clip: Towards a unified image-text pretraining paradigm for diverse medical imaging modalities. *arXiv preprint arXiv:2412.10372*, 2024.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *Advances in neural information processing systems*, 36:36652–36663, 2023.

Junzhe Li, Yutao Cui, Tao Huang, Weijie Kong, Yiming Cheng, Chuxuan Zeng, Yinping Ma, Chun Fan, Miles Yang, Zhao Zhong, et al. Mixgrpo: Unlocking flow-based grpo efficiency with mixed ode-sde. *arXiv preprint arXiv:2507.21802*, 2025.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Fan Liu, Delong Chen, Zhangqingyun Guan, Xiaocong Zhou, Jiale Zhu, Qiaolin Ye, Liyong Fu, and Jun Zhou. Remoteclip: A vision language foundation model for remote sensing. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–16, 2024. doi: 10.1109/TGRS.2024.3390838.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. *Advances in neural information processing systems*, 38:40783–40818, 2026.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations. In *International Conference on Learning Representations*, 2022.

Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6038–6047. IEEE, 2023.

Mihir Prabhudesai, Anirudh Goyal, Deepak Pathak, and Katerina Fragkiadaki. Aligning text-to-image diffusion models with reward backpropagation. 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 8748–8763, 2021.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 10674–10685. ieee, 2022.

Christoph Schuhmann. Improved aesthetic predictor: Clip+mlp aesthetic score predictor. `https://github.com/christophschuhmann/improved-aesthetic-predictor`, 2022.

Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen McKeown, and Rajesh Ranganath. A general framework for inference-time scaling and steering of diffusion models. *arXiv preprint arXiv:2501.06848*, 2025.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020b.

Bingda Tang, Yuhui Zhang, Xiaohan Wang, Jiayuan Mao, Ludwig Schmidt, and Serena Yeung-Levy. V-grpo: Online reinforcement learning for denoising generative models is easier than you think. *arXiv preprint arXiv:2604.23380*, 2026.

Felix Voigtlaender. A general version of price's theorem: F. voigtlaender. *Journal of theoretical probability*, 34(3):1474–1485, 2021.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8228–8238. IEEE, 2024.

Feng Wang and Zihao Yu. Coefficients-preserving sampling for reinforcement learning with flow matching. *arXiv preprint arXiv:2509.05952*, 2025.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023.

Gui-Song Xia, Jingwen Hu, Fan Hu, Baoguang Shi, Xiang Bai, Yanfei Zhong, Liangpei Zhang, and Xiaoqiang Lu. Aid: A benchmark data set for performance evaluation of aerial scene classification. *IEEE transactions on geoscience and remote sensing*, 55(7):3965–3981, 2017.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Kaiwen Zheng, Huayu Chen, Haotian Ye, Haoxiang Wang, Qinsheng Zhang, Kai Jiang, Hang Su, Stefano Ermon, Jun Zhu, and Ming-Yu Liu. Diffusionnft: Online diffusion reinforcement with forward process. In *International Conference on Learning Representations*, volume 2026, pp. 134129–134150, 2026.

A THEORETICAL ANALYSIS

The analysis follows the three method questions: how the learned map changes candidate geometry, when generated or reference-centered candidates improve the pool, and how the shared objective updates the predictor. The implementation and evaluation sections give the corresponding collection, gradient, and measurement procedures. Unless stated otherwise, a context $z = (x_t, t, c, x_0)$, the continuation field, and the predictor snapshot are fixed; expectations then concern only the independent auxiliary Gaussian directions. Population comparisons average over the specified context law. The assumptions are stated with each result, and proofs of the main-text theorems refer to their original labels rather than redefining them.

A.1 PERTURBATION GEOMETRY AND STATE CONDITIONING

Table 6: Notation used in the method and analysis.

Symbol	Meaning
θ, ϕ	Trainable generator and perturbation-predictor parameters.
x_0, x_t	Clean construction endpoint and saved parent state at noise level $t > 0$.
$\epsilon, \tilde{\epsilon}_k$	Standard Gaussian draw and independent auxiliary direction for child k.
ϵ_t	Effective component $[x_t - (1-t)x_0]/t$; not assumed Gaussian.
$\Gamma_t = \Gamma_\phi(x_t, t)$	State-conditioned, entry-wise perturbation map.
$\gamma, \gamma_0, \delta_\Gamma$	Scalar coefficient, map initialization center, and map variation range.
$x_{t,k}, u_k$	Child state and construction target $\epsilon_{\Gamma,k} - x_0$.
$\widehat{x}_{0,k}$	Scored endpoint $F_t(x_{t,k}, c)$, generally different from x_0.
$F_t, \mathcal{R}, S_t$	Frozen continuation, latent reward, and their composition $S_t(x, c) = \mathcal{R}(F_t(x, c), c)$.
r_i^{raw}, r_i	Endpoint reward and normalized training weight in $[0, 1]$.
m, Σ, D	Conditional displacement mean, covariance, and mean squared norm.
T, e_{rt}	Reference inversion level and latent round-trip error.
B, P, K_g, K_r, K	Numbers of bases, selected parents, generated children, reference children, and all scored candidates.

Conditional distribution. Let $A = \mathrm{diag}((\mathbf{1} + \Gamma_t^{\odot 2})^{-1/2})$ and $B_\Gamma = \mathrm{diag}(\Gamma_t \odot (\mathbf{1} + \Gamma_t^{\odot 2})^{-1/2})$. The parent identity $x_t = (1-t)x_0 + t\epsilon_t$ and Eq. (6) give

$$\Delta_k = x_{t,k} - x_t = t\big[(A - \mathbf{I})\epsilon_t + B_\Gamma \tilde{\epsilon}_k\big]. \tag{20}$$

Proposition 1 (Conditional perturbation distribution)**.** *For a fixed context and map, Δ_k is Gaussian, with*

$$m = t(A - \mathbf{I})\epsilon_t, \qquad \Sigma = t^2 B_\Gamma^2, \qquad D = \mathbb{E}[\|\Delta_k\|^2 \mid z] = \|m\|^2 + \mathrm{tr}(\Sigma). \tag{21}$$

Proof. In Eq. (20), the first term is deterministic and the second is a linear transformation of $\tilde{\epsilon}_k \sim \mathcal{N}(0, \mathbf{I})$. Its covariance is $t^2 B_\Gamma B_\Gamma^\top = t^2 B_\Gamma^2$; the expression for D follows from the bias–variance decomposition. □

A constant scalar map has isotropic conditional covariance. An entry-wise map changes the coordinate variances $t^2\Gamma_j^2/(1+\Gamma_j^2)$ and the mean simultaneously. Under a linearized reward, the conditional mean and variance are $S_t(x_t, c) + g^\top m$ and $g^\top \Sigma g$, respectively, where $g = \nabla_x S_t(x_t, c)$. Increasing spread alone therefore does not increase the mean of all children.

Coefficient normalization and state dependence. For a deterministic map independent of two standard Gaussian components, $A^2 + B_\Gamma^2 = \mathbf{I}$ gives a standard Gaussian marginal. This conclusion does not extend to a map that observes a selected parent. For example, let $\Gamma(\epsilon) = \gamma_+$ for $\epsilon \geq 0$ and γ_- otherwise, with distinct positive coefficients. For independent $\epsilon, \tilde{\epsilon} \sim \mathcal{N}(0, 1)$ and $a_\pm = (1+\gamma_\pm^2)^{-1/2}$,

$$\mathbb{E}\left[\frac{\epsilon + \Gamma(\epsilon)\tilde{\epsilon}}{\sqrt{1 + \Gamma(\epsilon)^2}}\right] = \frac{a_+ - a_-}{\sqrt{2\pi}} \neq 0. \tag{22}$$

Thus normalized coefficients do not imply a standard Gaussian marginal for the state-conditioned construction. Inversion also supplies an effective component $\epsilon_T^{\mathrm{ref}}$ with no assumed Gaussian law.

Scalar restriction and expressivity. Writing $\gamma_\pm = \gamma_0 \pm \delta_\Gamma$, the bounded map satisfies $\gamma_- < \Gamma_j < \gamma_+$ with $\gamma_- > 0$, so each coordinate variance is between $t^2\gamma_-^2/(1+\gamma_-^2)$ and $t^2\gamma_+^2/(1+\gamma_+^2)$. These are moment bounds, not a hard radius on individual Gaussian draws. For an integrable utility $U(z, \Gamma)$ and a map class $\mathcal{F}$ containing the constant scalar maps,

$$\sup_{f \in \mathcal{F}} \mathbb{E}_z U(z, f(x_t, t)) \geq \sup_{\gamma \in (\gamma_-, \gamma_+)} \mathbb{E}_z U(z, \gamma \mathbf{1}). \tag{23}$$

This function-class comparison does not assert that optimization attains the supremum. The implemented map is diagonal in latent coordinates and observes (x_t, t); it does not implement an arbitrary full covariance or separately observe c and x_0.

Bounded diagonal allocation. The scalar restriction can obscure why coordinate-wise control is useful. To isolate its representational effect, fix a parent and let d_x be the latent dimension. Set

$$s_{\min} = t^2 \frac{\gamma_-^2}{1+\gamma_-^2}, \qquad s_{\max} = t^2 \frac{\gamma_+^2}{1+\gamma_+^2}.$$

Consider diagonal covariances with $s_{\min} \leq \Sigma_{jj} \leq s_{\max}$ and a fixed trace $C \in [d_x s_{\min}, d_x s_{\max}]$. This trace constraint is an analytic comparison, not an additional training constraint. Write $w_j = g_j^2$, order $w_{(1)} \geq \cdots \geq w_{(d_x)}$, and define $V = C - d_x s_{\min}$, $a_s = s_{\max} - s_{\min} > 0$, and $k = \lfloor V/a_s \rfloor$.

Proposition 2 (Capped variance allocation)**.** *Under the preceding constraints,*

$$\max_{\Sigma} g^\top \Sigma g = s_{\min} \sum_{j=1}^{d_x} w_j + a_s \sum_{j=1}^{k} w_{(j)} + (V - k a_s) w_{(k+1)}, \tag{24}$$

where the last term is zero when $k = d_x$. The isotropic allocation has value $(C/d_x)\|g\|^2$. The maximum is strictly larger when $d_x s_{\min} < C < d_x s_{\max}$ and the squared gradient coordinates are not all equal.

Proof. Write $\Sigma_{jj} = s_{\min} + v_j$, with $0 \leq v_j \leq a_s$ and $\sum_j v_j = V$. If $w_i > w_j$, $v_i < a_s$, and $v_j > 0$, transferring a positive amount from v_j to v_i strictly increases $\sum_j w_j v_j$. Filling the largest coordinates first gives Eq. (24). At an interior trace budget, the isotropic solution permits such a transfer whenever two squared gradient coordinates differ. At either endpoint all allocations are fixed. Since the predictor's coefficient interval is open, boundary solutions are interpreted as suprema in its closure. □

This result concerns projected spread $g^\top \Sigma g$, not mean reward alone: the actual map also changes m, and nonlinear rewards introduce curvature. It therefore explains the flexibility of entry-wise exploration without claiming that variance maximization is the training objective.

For example, consider two equally likely contexts distinguishable from (x_t, t), with reward gradients $(1, 0)$ and $(0, 1)$ and trace $C = s_{\min} + s_{\max}$. Context-dependent allocations $(s_{\max}, s_{\min})$ and $(s_{\min}, s_{\max})$ achieve projected variance $s_{\max}$ in both contexts; any fixed allocation averages $(s_{\min} + s_{\max})/2$. This illustrates why state dependence can matter. The shared convolutional predictor need not realize every such idealized allocation, and the full reward change still includes drift and curvature.

A.2 Candidate-pool reward

Proof of Theorem 1. Retained bases have expected reward μ_b, while a generated child has expected reward $\mu_+ + G_\phi$. With $w = (K - B)/K$,

$$J_{\mathrm{RACE}} = (1 - w)\mu_b + w(\mu_+ + G_\phi), \qquad J_{\mathrm{base}} = \mu_b.$$

Subtracting gives the equality in Eq. (7). Uniform allocation over the top-P bases makes their realized average at least the full base-pool average, so $\delta_{\mathrm{sel}} \geq 0$. Under the stated Hessian bound, Taylor's theorem gives

$$S_t(x_t + \Delta, c) - S_t(x_t, c) \geq g_z^\top \Delta - \tfrac{1}{2} H_z \|\Delta\|^2.$$

Take the conditional expectation using Proposition 1, then average over contexts to obtain the lower bound in Eq. (7). Integrability of the displayed terms and a Hessian bound along all reached segments are required. An untruncated Gaussian proposal requires these bounds over its full support.

For the policy comparison, $\mathbb{E}_{\Delta\sim\pi_{\phi'}(\cdot|z)}A_\phi(z,\Delta) = V_{\phi'}(z) - V_\phi(z)$. The retained-base contribution cancels; averaging under the unchanged context law and multiplying by w proves Eq. (8). □

Allocation and interpretation. For nonuniform child allocation, let $a_p \geq 0$ be the fraction of generated children assigned to selected parent p, with $\sum_p a_p = 1$. The same decomposition uses $\mu_+ = \mathbb{E}\sum_p a_p S_{t_p}(x_{t_p}, c)$. Its selection gap need not be nonnegative. Retaining bases preserves an independent-generation component, but $B < K$ still gives fewer independent starts than the baseline. The bound tests whether the selected-parent advantage and perturbation benefit offset this trade-off. It compares candidate pools under fixed collection models, not the final performance of separately trained generators.

The corresponding comparison of two predictor snapshots keeps the base distribution, allocation, and continuation model unchanged. A positive expected advantage in Eq. (8) is a condition on the new proposal distribution. It is not supplied automatically by increasing child variance or decreasing the downstream regression loss.

A.3 Reference locality, construction targets, and reward

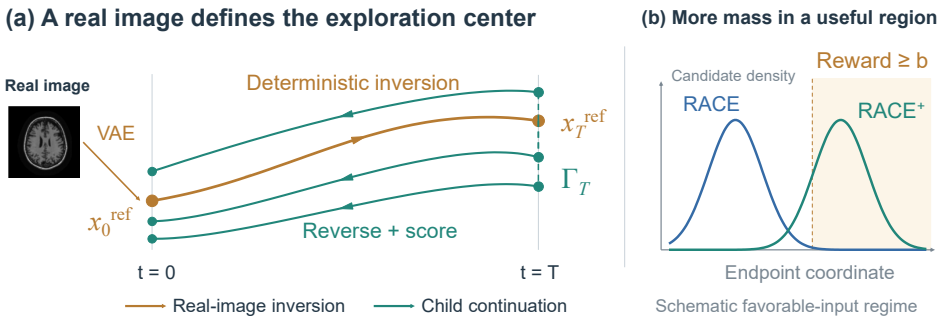

Figure 5: **Reference-guided exploration.** Inversion provides an input-derived construction center. The schematic favorable-input regime additionally requires controlled round-trip error, perturbation size, and a sufficient reward margin.

Let $\Delta = x_{T,k}^{\text{ref}} - x_T^{\text{ref}}$, $D_r = \mathbb{E}[\|\Delta\|^2 \mid I^{\text{ref}}, c, T]$, and $e_{\text{rt}} = \|F_T(x_T^{\text{ref}}, c) - x_0^{\text{ref}}\|$.

Proposition 3 (Reference locality)**.** *For $\alpha \in (0,1)$ and $D_r > 0$, suppose F_T is L_F-Lipschitz on the radius-$\sqrt{D_r/\alpha}$ ball about x_T^{ref}. With probability at least $1-\alpha$,*

$$\|F_T(x_T^{\text{ref}} + \Delta, c) - x_0^{\text{ref}}\| \leq e_{\text{rt}} + L_F\sqrt{D_r/\alpha}. \tag{25}$$

Proof. Let $\rho_r = \sqrt{D_r/\alpha}$. Markov's inequality gives $\Pr(\|\Delta\| > \rho_r) \leq D_r/\rho_r^2 = \alpha$. On the complementary event, both states lie in the specified neighborhood and

$$\begin{aligned}\|F_T(x_T^{\text{ref}} + \Delta, c) - x_0^{\text{ref}}\| &\leq \|F_T(x_T^{\text{ref}} + \Delta, c) - F_T(x_T^{\text{ref}}, c)\| + e_{\text{rt}} \\ &\leq L_F\|\Delta\| + e_{\text{rt}} \leq L_F\rho_r + e_{\text{rt}}.\end{aligned}$$

If $D_r = 0$, then $\Delta = 0$ almost surely and the discrepancy is the round-trip error. □

If the continuity bound holds over the full proposal support, conditional Jensen's inequality instead gives $\mathbb{E}\|F_T(x_T^{\text{ref}} + \Delta, c) - x_0^{\text{ref}}\| \leq e_{\text{rt}} + L_F\sqrt{D_r}$. This expectation bound needs stronger support assumptions than the high-probability local statement.

Construction and scored endpoints. The target uses x_0, whereas the reward uses $\widehat{x}_{0,k} = F_t(x_{t,k}, c)$. In particular,

$$\widehat{u}_k - u_k = \frac{x_0 - \widehat{x}_{0,k}}{t}, \qquad \widehat{u}_k = \frac{x_{t,k} - \widehat{x}_{0,k}}{t}. \tag{26}$$

For a generated parent satisfying $F_t(x_t, c) = x_0$, full-support Lipschitz continuity yields $\mathbb{E}\|\widehat{u}_k - u_k\| \le L_F\sqrt{D}/t$. For an inverted reference, the round-trip term gives Eq. (11). Because $D = t^2[\|(A-\mathbf{I})\epsilon_t\|^2 + \mathrm{tr}(B_\Gamma^2)]$, these bounds do not imply monotonicity in t alone. The constructed child need not have the Gaussian forward-kernel law conditioned on its scored endpoint.

Proof of Theorem 2. The two pools share their bases and unchanged children in distribution. Linearity of expectation gives $J_{\mathrm{RACE}^+} - J_{\mathrm{RACE}} = (K_r/K)(J_r - J_g)$, where J_g concerns only the replaced children. Under the full-support assumptions on F_T and $\mathcal{R}$,

$$\begin{aligned}\mathbb{E}[\mathcal{R}(F_T(x_T^{\mathrm{ref}} + \Delta, c), c) \mid I^{\mathrm{ref}}, c, T] \ge \mathcal{R}(x_0^{\mathrm{ref}}, c) \\ - L_R(e_{\mathrm{rt}} + L_F\sqrt{D_r}).\end{aligned} \tag{27}$$

Average over references and inversion levels, then substitute into the pool difference to prove Eq. (12). The reference score is that of its VAE reconstruction; a raw-image version needs an additional VAE reconstruction-error term. □

Reward-qualified discovery. Under the local assumptions of Proposition 3 and reward Lipschitzness on the reached endpoint region, the condition

$$\mathcal{R}(x_0^{\mathrm{ref}}, c) - L_R(e_{\mathrm{rt}} + L_F\sqrt{D_r/\alpha}) \ge b \tag{28}$$

implies a reference-child success probability $p_r \ge 1 - \alpha$ for raw reward at least b. This follows by applying the reward bound on the same probability-$1-\alpha$ event. For $M = K - B$ conditionally independent trials with common generated- and reference-child probabilities p_g and $p_r \ge p_g$, replacing K_r trials gives

$$1 - (1-p_g)^{M-K_r}(1-p_r)^{K_r} \ge 1 - (1-p_g)^M. \tag{29}$$

Effective exploration region and sample efficiency. Define

$$\mathcal{E}_r(\alpha) = \{y : \|y - x_0^{\mathrm{ref}}\| \le e_{\mathrm{rt}} + L_F\sqrt{D_r/\alpha}\}. \tag{30}$$

The locality result places at least $1-\alpha$ of the reference-child probability in this region. Reference guidance is useful when this region concentrates probability on task-relevant structure and its reward margin satisfies Eq. (28). This does not reduce the full support or dimension of a nondegenerate Gaussian, nor establish smaller covariance than generated-parent exploration. A mismatched reference or inaccurate inversion can remove the advantage.

For independent repeated child trials at fixed success probabilities $p_g, p_r > 0$, the expected counts to first success are $1/p_g$ and $1/p_r$. Thus $p_r > p_g$ gives fewer expected scored children. If P_+ and P_0 denote the batch success probabilities on the two sides of Eq. (29), independent complete batches with K scored candidates have expected first-success costs K/P_+ and K/P_0. These are fixed-distribution discovery costs; training curves additionally involve evolving models, and wall-clock cost includes inversion.

A.4 Shared-objective derivatives

For fixed (x, t, c), write $d = v - v_{\mathrm{old}}$ and $a = u - v_{\mathrm{old}}$. The per-sample objective is

$$\ell(v; u, r) = r\|\beta d - a\|^2 + (1-r)\|-\beta d - a\|^2 = \beta^2\|d\|^2 - 2\beta(2r-1)d^\top a + \|a\|^2. \tag{31}$$

For $\beta > 0$ and $\mathbb{E}_\mu[\|a\|^2 \mid x, t, c] < \infty$, the conditional objective is strictly convex in d. Its stationarity condition is

$$0 = 2\beta^2 d^* - 2\beta\mathbb{E}_\mu[(2r-1)a \mid x, t, c].$$

Solving for d^* gives

$$v^*(x, t, c) = v_{\mathrm{old}}(x, t, c) + \frac{1}{\beta}\mathbb{E}_\mu[(2r-1)(u - v_{\mathrm{old}}(x, t, c)) \mid x, t, c]. \tag{32}$$

The law μ includes the actual construction and group normalization; an additional sample-dependent loss multiplier would require its corresponding weighted conditional moment. The velocity derivative is $\nabla_v \ell = 2\beta^2 d - 2\beta(2r-1)a$. At $r = 1/2$, only the reward-direction term vanishes.

Predictor gradient. For a stored child context, $D_\phi u_i = t_i^{-1} D_\phi x_i$. Define $M_i = D_x(v_\theta - v_{\text{old}})$ and $N_i = t_i^{-1}\mathbf{I} - D_x v_{\text{old}}$ at the live child state. With the recorded r_i fixed,

$$\nabla_\phi \ell_i = 2(D_\phi x_i)^\top \left[\beta^2 M_i^\top d_i - \beta(2r_i - 1)(M_i^\top a_i + N_i^\top d_i) + N_i^\top a_i\right], \tag{33}$$

where the explicit construction Jacobian is

$$D_\phi x_i = t_i \operatorname{diag}\left(\frac{\tilde{\epsilon}_i - \Gamma_i \odot \epsilon_{t_i}}{(\mathbf{1} + \Gamma_i^{\odot 2})^{3/2}}\right) D_\phi \Gamma_i. \tag{34}$$

To see the chain rule explicitly, hold the parent and auxiliary direction fixed. Then $D_\phi d_i = M_i D_\phi x_i$ and $D_\phi a_i = N_i D_\phi x_i$. Differentiating the three terms in Eq. (31) gives, respectively, $2\beta^2 (D_\phi d_i)^\top d_i$, $-2\beta(2r_i - 1)[(D_\phi d_i)^\top a_i + (D_\phi a_i)^\top d_i]$, and $2(D_\phi a_i)^\top a_i$. Their sum is Eq. (33); differentiating the normalized mixture gives Eq. (34). Both vector fields retain input Jacobians, while the old field's parameters stay frozen. Bases have $D_\phi x_i = D_\phi u_i = 0$. If the two fields coincide as functions with identical input-gradient paths, $d_i = D_\phi d_i = 0$ and

$$\nabla_\phi \ell_i = \nabla_\phi \|u_i(\phi) - v_{\text{old}}(x_i(\phi), t_i, c)\|^2. \tag{35}$$

The predictor gradient is then reward-independent; the cross term becomes reward-dependent after the fields separate.

The construction and scored endpoints differ, so the algebra above does not by itself identify the update with the endpoint-distribution improvement direction of DiffusionNFT (Zheng et al., 2026). Replacing a child by a fresh forward pair from $\widehat{x}_0$ would change both the method and the predictor's gradient path. No such replacement is assumed here.

A.5 Conditional mean child-reward improvement

For the fixed context law ρ, let $q_\phi(x \mid z)$ be the induced child-state density and

$$\mathcal{J}(\phi) = \mathbb{E}_{z\sim\rho} \int S_t(x, c) q_\phi(x \mid z)\, dx. \tag{36}$$

For RACE$^+$, ρ mixes generated- and reference-parent contexts with weights $K_g/(K_g + K_r)$ and $K_r/(K_g + K_r)$. This is the mean raw reward of all children.

Reward sensitivity of the mixture. At fixed z, abbreviate $\Gamma = \Gamma_\phi(x_t, t)$. Each coordinate has mean $\mu_j = (1-t)x_{0,j} + t\epsilon_{t,j}/\sqrt{1+\Gamma_j^2}$ and variance $\sigma_j^2 = t^2\Gamma_j^2/(1+\Gamma_j^2)$. Since $\Gamma_j > \gamma_- > 0$, differentiation of the Gaussian density at fixed x, followed by evaluation at $x_j = \mu_j + \sigma_j \tilde{\epsilon}_j$, gives

$$\frac{\partial \log q_\phi(x \mid z)}{\partial \Gamma_j} = \frac{(\tilde{\epsilon}_j^2 - 1)/\Gamma_j - \epsilon_{t,j}\tilde{\epsilon}_j}{1 + \Gamma_j^2}. \tag{37}$$

Indeed, $\mu_j' = -t\Gamma_j \epsilon_{t,j}(1+\Gamma_j^2)^{-3/2}$ and $(\log \sigma_j)' = [\Gamma_j(1+\Gamma_j^2)]^{-1}$. Define

$$s_\phi(z, \tilde{\epsilon}) = (D_\phi \Gamma)^\top \left[\frac{(\tilde{\epsilon}^{\odot 2} - \mathbf{1})/\Gamma - \epsilon_t \odot \tilde{\epsilon}}{\mathbf{1} + \Gamma^{\odot 2}}\right]. \tag{38}$$

With justified differentiation–integration interchange and integrable moments, $\mathbb{E}[s_\phi \mid z] = 0$. For any baseline $b(z)$ independent of the auxiliary direction,

$$\nabla_\phi \mathcal{J}(\phi) = \mathbb{E}[(S_t(x_{t,k}, c) - b(z)) s_\phi(z, \tilde{\epsilon})]. \tag{39}$$

These are analysis identities; they add no score-function term to the training objective.

Proposition 4 (Drift–curvature sensitivity of mean child reward)**.** *Fix z with $t > 0$. Suppose S_t is twice continuously differentiable, its displayed derivatives are integrable, and differentiation and Gaussian integration by parts are valid throughout a compact positive coefficient interval. Then*

$$\begin{aligned}\frac{\partial}{\partial \Gamma_j}\mathbb{E}S_t(X_\Gamma, c) = &-\frac{t\Gamma_j\epsilon_{t,j}}{(1+\Gamma_j^2)^{3/2}}\mathbb{E}\,\partial_j S_t(X_\Gamma, c) \\ &+ \frac{t^2\Gamma_j}{(1+\Gamma_j^2)^2}\mathbb{E}\,\partial_{jj} S_t(X_\Gamma, c).\end{aligned} \tag{40}$$

Proof. Write $X_{\Gamma,j} = \mu_j + \sigma_j \tilde{\epsilon}_j$. Pathwise differentiation gives $\mu_j' \mathbb{E}\partial_j S_t + \sigma_j' \mathbb{E}[\tilde{\epsilon}_j \partial_j S_t]$. Gaussian integration by parts gives $\mathbb{E}[\tilde{\epsilon}_j \partial_j S_t] = \sigma_j \mathbb{E}\partial_{jj} S_t$. Substituting μ_j' above and $\sigma_j \sigma_j' = t^2 \Gamma_j (1+\Gamma_j^2)^{-2}$ proves the result, consistent with the Gaussian differentiation identities of Price's theorem (Voigtlaender, 2021). □

The terms describe coupled mean shift and curvature-weighted variance change. For a linear reward, the second term is zero. For a nonlinear reward, variance can help or hurt according to the expected curvature. Neither term establishes that the shared-loss optimizer chooses a favorable direction.

From coordinate sensitivity to a finite map change. Let $G_j(z, \Gamma)$ denote the right-hand side of Eq. (40). With an integrable context envelope, the chain rule gives $\nabla_\phi \mathcal{J} = \mathbb{E}_z[(D_\phi \Gamma)^\top G]$. For two admissible maps Γ^{e} and Γ^{l}, define $\Gamma^\lambda = \Gamma^{\mathrm{e}} + \lambda(\Gamma^{\mathrm{l}} - \Gamma^{\mathrm{e}})$. The fundamental theorem of calculus then yields

$$\mathcal{J}(\Gamma^{\mathrm{l}}) - \mathcal{J}(\Gamma^{\mathrm{e}}) = \mathbb{E}_z \int_0^1 \sum_j (\Gamma_j^{\mathrm{l}} - \Gamma_j^{\mathrm{e}}) G_j(z, \Gamma^\lambda)\, d\lambda. \tag{41}$$

Here $\mathcal{J}(\Gamma)$ denotes the child mean induced by a coefficient map. A positive integral certifies improvement under the same parents and continuation field. The segment is an analysis device, not necessarily the network's optimization path. This connects the two schematic reward distributions to a change in learned geometry without treating the schematic as measured training data.

Proof of Theorem 3. Condition on the training history before update j. The smoothness inequality (Bottou et al., 2018) gives

$$\mathcal{J}(\phi_j - \eta_j h_j) \geq \mathcal{J}(\phi_j) - \eta_j \langle \nabla \mathcal{J}(\phi_j), h_j \rangle - \tfrac{1}{2} L_J \eta_j^2 \|h_j\|^2.$$

Taking the conditional expectation yields Eq. (18) with $\chi_j = -\langle \nabla \mathcal{J}(\phi_j), \mathbb{E}_j h_j \rangle$ and $V_j = \mathbb{E}_j \|h_j\|^2$. For $\chi_j > 0$, $L_J > 0$, and $0 < \eta_j < 2\chi_j/(L_J V_j)$, the lower bound is positive. When $L_J = 0$, every positive step has positive expected increment if $\chi_j > 0$; when $V_j = 0$, the update vanishes. The effective direction h_j includes momentum, preconditioning, and weight decay when present. □

The condition $\chi_j > 0$ compares endpoint-reward sensitivity in Eq. (39) with the actual surrogate direction in Eq. (33); it does not follow from a decrease in the surrogate. Under fixed contexts and continuation, summing the conditional bound and applying the tower property gives

$$\mathbb{E}[\mathcal{J}(\phi_M)] - \mathcal{J}(\phi_0) \geq \sum_{j=0}^{M-1} \mathbb{E}\left[\eta_j \chi_j - \tfrac{1}{2} L_J \eta_j^2 V_j\right]. \tag{42}$$

Nonnegative expected increments imply a nondecreasing expected mean; individual children and finite-sample estimates can still fluctuate.

Why alignment is necessary. Consider the one-dimensional context $x_0 = x_t = 0$, $v_\theta = v_{\mathrm{old}} = 0$, and $F_t(x) = x$. With $b_\gamma = \gamma/\sqrt{1+\gamma^2}$, the construction gives $X = t b_\gamma \tilde{\epsilon}$ and $u = b_\gamma \tilde{\epsilon}$. For any recorded reward weight, $\mathbb{E}\ell = b_\gamma^2$ increases with $\gamma > 0$. For the bounded reward $S_t(x) = 1 - e^{-x^2}$,

$$\mathcal{J}(\gamma) = 1 - (1 + 2t^2 b_\gamma^2)^{-1/2}$$

also increases with γ. A small descent step in γ therefore decreases both loss and child reward. This analytic example concerns the fixed-continuation surrogate, not observed benchmark performance.

Changing models and collection copies. Let $\mathcal{J}_j$ include round j's contexts and continuation field, and suppose $\mathbb{E}_j[\mathcal{J}_{j+1}(\phi_{j+1}) - \mathcal{J}_j(\phi_{j+1})] \geq -d_j$. If $\mathcal{J}_j$ is G_j-Lipschitz in the relevant parameter region and $\ell_j = \|\phi_j^{\mathrm{col}} - \phi_j\|$, then

$$\begin{aligned} \mathbb{E}_j[\mathcal{J}_{j+1}(\phi_{j+1}^{\mathrm{col}})] - \mathcal{J}_j(\phi_j^{\mathrm{col}}) \geq \eta_j \chi_j - \tfrac{1}{2} L_j \eta_j^2 V_j - d_j \\ - G_j \ell_j - \mathbb{E}_j[G_{j+1} \ell_{j+1}]. \end{aligned} \tag{43}$$

To obtain this bound, add and subtract the two live-predictor rewards, apply the one-step inequality to $\mathcal{J}_j$, and bound the context change and copy discrepancies as stated. With fixed models, zero copy lag, and integrable terms, summing the one-step bounds gives the corresponding cumulative improvement condition. Online training trends can therefore reflect generator progress and changing contexts as well as predictor updates.

B ADDITIONAL QUALITATIVE EXAMPLES

The comparisons serve two different purposes. Noise-initialized generation shows the behavior of the trained models under matched conditions; explicit reconstruction checks whether the reference route preserves structure while allowing variation. Neither a few displayed successes nor reference similarity alone establishes an overall success rate.

B.1 BRAIN MRI GENERATION

Figure 6 shows CN, MCI, and AD conditioning examples. The paired image serves as an evaluation reference and is not a generator input under the main inference protocol. RACE+ may use paired images during training, whereas held-out generation for all methods starts from fresh Gaussian noise. Category labels denote requested conditioning categories rather than independently verified diagnoses of generated outputs.

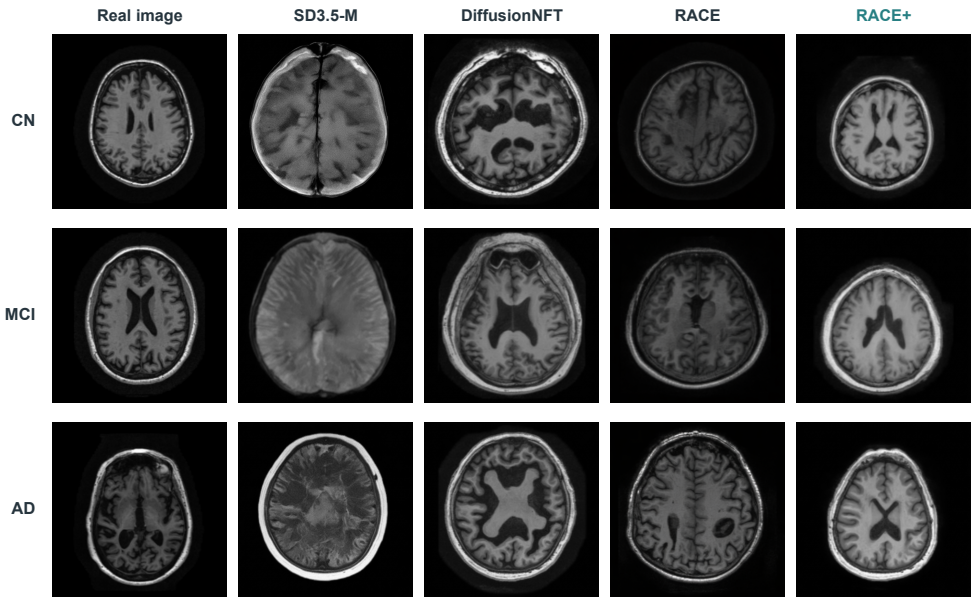

Figure 6: **Brain MRI examples.** Rows denote CN, MCI, and AD conditioning categories. Columns show the evaluation reference and outputs from SD3.5-M, DiffusionNFT, RACE, and RACE^{+}. The reference is not provided to the generator during held-out inference.

B.2 REFERENCE RECONSTRUCTION AND CHILD VARIATION

Figure 7 separates the raw reference, VAE reconstruction, no-mix inversion–continuation output, and two augmented children. This comparison addresses the route analyzed in Appendix A.3: autoencoder distortion, numerical round-trip error, and perturbation effects are distinct. The displayed MRI and CT rows permit comparison of major structures; the AID row is less informative at this display scale and requires inspection at the original resolution. No numerical reconstruction or child-diversity measurement is inferred from the montage.

B.3 COCO PROMPT-CONDITIONED GENERATION

Figure 8 compares model outputs under the same prompt in each row. The three examples concern street scenes with light trails, downhill skiing, and horned mountain goats beside grass and snow. They illustrate differences in motion appearance, subject pose, object composition, and scene context. The associated COCO image shows the image–caption pair rather than a pixel-level reconstruction target; these qualitative examples complement the aggregate scores in Table 1.

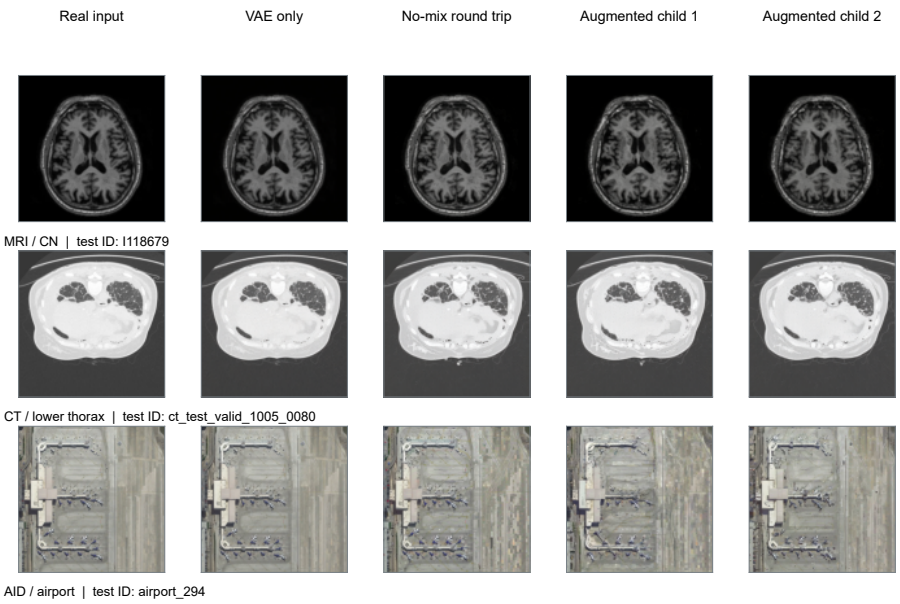

Figure 7: **Reference reconstruction and child variation.** Columns show the raw reference, its VAE reconstruction, inversion–reverse output with mixing disabled, and two augmented children for MRI, CT, and AID.

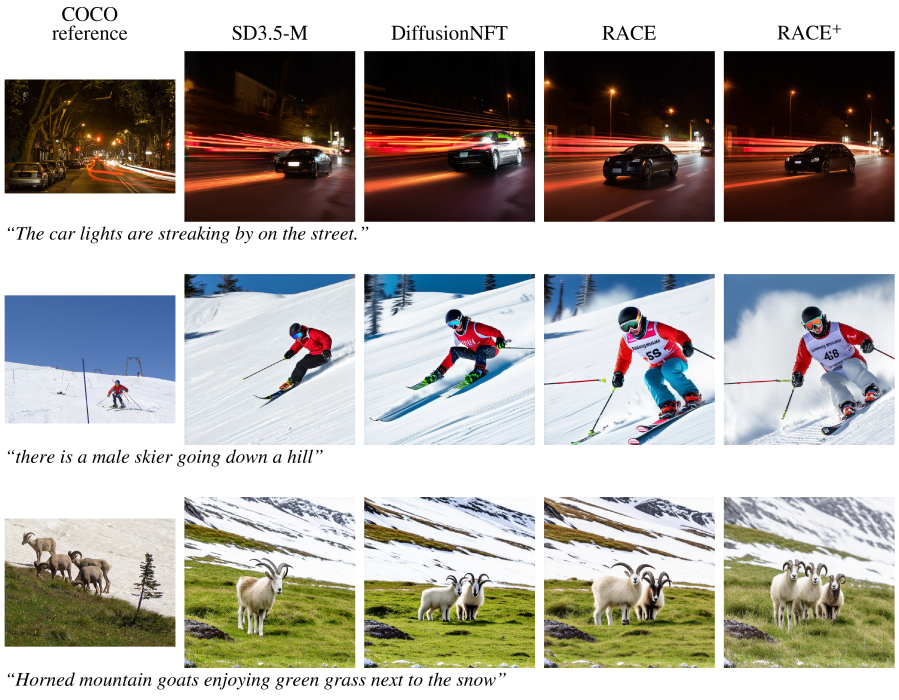

Figure 8: **COCO prompt-conditioned generation.** Each row shows the associated COCO reference and outputs from SD3.5-M, DiffusionNFT, RACE, and RACE$^+$ under the displayed prompt. The examples cover street-light trails, downhill skiing, and horned mountain goats beside grass and snow. References illustrate the image–caption pairs and are not test-time generator inputs.

Additional prompt-only comparisons. Figure 9 provides enlarged comparisons without reference-image columns. In the first example, both RACE and RACE$^+$ include the second burger requested by

the prompt, whereas the displayed DiffusionNFT output contains only one. In the second example, both variants exhibit more coherent motorcycle wheel and body structures. These selected examples complement the aggregate evaluation in Table 1.

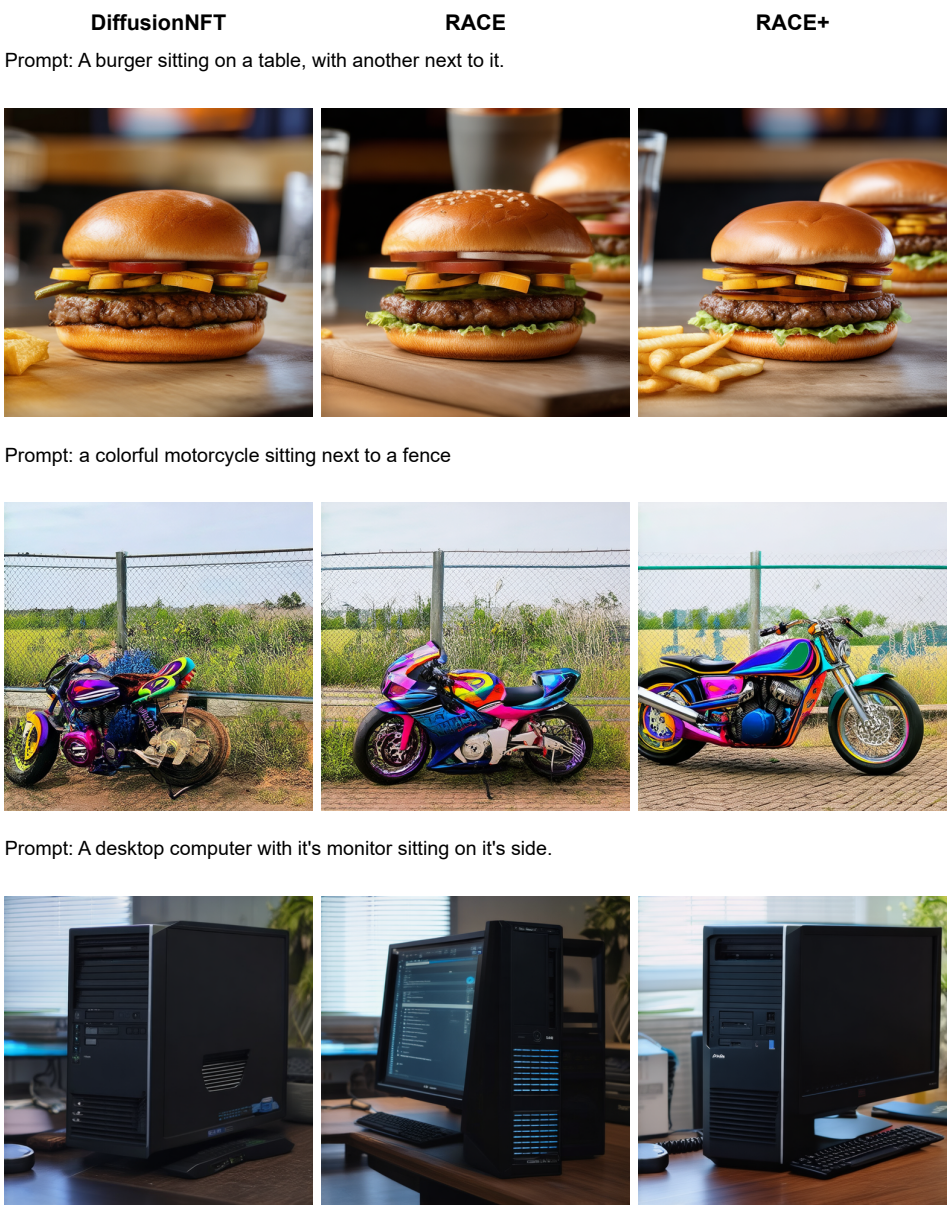

Figure 9: **Additional qualitative comparisons on COCO.** All methods use the same prompt and inference seed within each row, without reference images at inference. Examples are selected to illustrate improvements in object composition and structural coherence.

C Method and Implementation Details

C.1 Candidate Collection and Scoring Budgets

The main-text Algorithm 1 summarizes the method. Figure 10 gives a concrete $K = 8$ example, Table 7 records both benchmark budgets, and Algorithm 2 specifies collection and differentiable reuse. These views distinguish the candidate sources from their final reward weights.

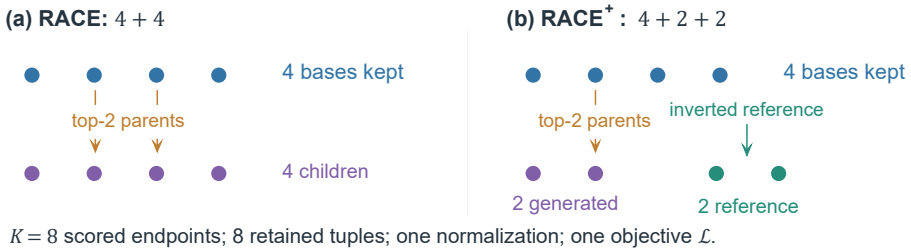

Figure 10: **Candidate allocation at $K = 8$.** All four base samples are retained. In the illustrated $P = 2$ case, RACE allocates four children to the selected generated parents, whereas RACE$^+$ uses two generated children and two reference children. The eight raw scores are normalized jointly and all eight candidates enter one shared objective. The illustrated parent count is not a universal setting.

A worked collection round. Generate four independent bases and evaluate all four. In the illustrated RACE allocation, two selected parents each supply two children. In RACE$^+$, the same two selected generated parents each supply one child and the inverted reference supplies the remaining two. The reference therefore replaces two child slots; it is not a ninth training sample. Both variants retain the four bases, including those not chosen as parents. Parent ranking chooses where to explore, not which bases are allowed to contribute to training. The $P = 2$ example specifies the allocation shown in the diagram.

Table 7: **Per-prompt candidate allocations.** Every row satisfies $B + K_g + K_r = K$. All scored candidates enter the objective. Small benchmarks are Architecture & Interior, MRI, CT, and AID.

Setting	Method	B	K_g	K_r	K
Small benchmarks	DiffusionNFT	8	0	0	8
	RACE	4	4	0	8
	RACE$^+$	4	2	2	8
COCO	DiffusionNFT	24	0	0	24
	RACE	20	4	0	24
	RACE$^+$	20	2	2	24

Scoring, normalization, and sample retention. Base scores are cached; each completed child is scored once. The illustrated round therefore requires $4 + 4 = 8$ endpoint reward evaluations, not eight extra evaluations after parent selection. Raw scores are pooled before computing the continuous weights in Eq. (13). Recomputing pool statistics does not require rescoring a base. A selected parent need not remain above the final pool mean after children are added, and neither child source has a predetermined positive or negative label. There is no final top-K filter and no fixed requirement for equal positive and negative counts.

Stored data and live reconstruction. For a base endpoint $y_{0,i}$, optimization draws a forward pair with state $(1 - t_i)y_{0,i} + t_i\epsilon_i$ and target $\epsilon_i - y_{0,i}$. A child retains its parent state, construction endpoint, time, auxiliary direction, and reward. Optimization rebuilds both its state and target with the live predictor. The clean construction endpoint remains the stored x_0, not the child's newly scored endpoint. These stored choices keep the construction differentiable while excluding continuation and the reward model from the optimization graph. Their actual sampling distributions and reuse count are run settings rather than consequences of K.

Budget scaling and compute. COCO enlarges the independent base pool while keeping four children. The child fraction changes from $1/2$ to $1/6$, and the reference fraction in RACE$^+$ from $1/4$ to $1/12$. These are separate benchmark configurations, not a controlled K ablation. Only children

Algorithm 2 Detailed collection and differentiable reuse for one round

Require: Live parameters (θ, ϕ); frozen copies $(\theta_{\text{old}}, \phi_{\text{old}})$; reward R.
Require: Condition c; $K = B + K_g + K_r$; $P \le B$; a training reference when $K_r > 0$.
Generate and score B bases using v_{old}; retain their endpoints and cached scores.
Select the top-P bases as parents; allocate exactly K_g generated-child slots.
for each allocated generated child **do**
    Store the parent state x_t, construction endpoint x_0, level t, and draw $\tilde{\epsilon}_k$.
    Construct $x_{t,k}$ and u_k with $\Gamma_{\phi_{\text{old}}}$; continue, decode, and score the child.
end for
if $K_r > 0$ **then**
    Encode and deterministically invert the training reference to x_T^{ref}.
    Construct and score K_r reference children; store their contexts and auxiliary directions.
end if
Merge all K candidates; compute their pooled reward weights using Eq. (13).
for each optimization batch from the stored pool **do**
    Build fresh base forward pairs and rebuild child states and targets with live Γ_ϕ.
    Evaluate v_θ and input-differentiable v_{old}; keep recorded rewards fixed.
    Update (θ, ϕ) using only the shared objective in Eq. (15).
end for
Refresh collection copies using the configured update rule.

contribute predictor derivatives:

$$\nabla_\phi \mathcal{L} = \frac{K_g + K_r}{K} \left[\frac{1}{K_g + K_r} \sum_{i \in \mathcal{C}_g \cup \mathcal{C}_r} \nabla_\phi \ell_i \right].$$

This factor describes the unpreconditioned objective gradient; learning rate and optimizer preconditioning also affect the applied update. Cumulative training cost counts $N_R = \sum_{g=1}^{G} K^{(g)}$ over all collected prompt groups. Equal scored-candidate budgets do not imply equal time: continuation, reference inversion, optimization reuse, and validation add different costs. Diagnostic reward evaluations are recorded separately from training collection and included when total compute is compared.

C.2 Predictor architecture and gradient paths

The 16-channel latent is concatenated with a spatially broadcast 16-dimensional sinusoidal time embedding. Two 3×3 convolutions operate at 32 channels with GELU activations and a residual connection inside the 32-channel block. A zero-initialized 1×1 projection maps to 16 channels. Including biases, the parameter count is $2(32 \cdot 32 \cdot 3 \cdot 3 + 32) + (32 \cdot 16 + 16) = 19{,}024$.

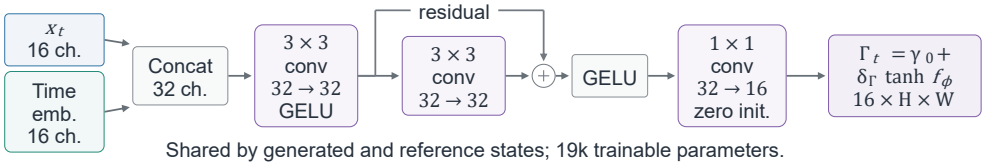

Figure 11: **Shared state-conditioned predictor.** The output has the latent's shape and parameterizes Eq. (4). The residual connection remains within the 32-channel block.

The last projection initializes $\Gamma_t = \gamma_0$. Its zero weights initially block derivatives to earlier layers; the output projection can update on the first backward pass. One predictor handles both generated and reference contexts.

During optimization, Γ_ϕ, $x_i(\phi)$, and $u_i(\phi)$ retain gradients. The old field has frozen parameters but retains its input Jacobian; the live field retains both parameter and input gradients. Parent contexts, times, random draws, reward labels, and normalization statistics are constants. Continuation, VAE decoding for scoring, and reward evaluation are outside this graph. The target derivative is $\nabla_{u_i} \ell_i = 2[u_i - r_i v_i^+ - (1 - r_i) v_i^-]$; no independent reference or reconstruction loss is added.

Table 8: **Gradient paths during optimization.** Freezing old-model parameters does not detach its input. The collection/scoring pass remains outside the graph.

Quantity	Parameter update	Input/construction derivative
Live generator v_θ	Yes, through θ	Retained
Old field v_{old}	No	Retained
Live predictor Γ_ϕ	Yes, through ϕ	Retained
Child state and target	Through ϕ	Retained for both
Parent context and random draws	No	Fixed
Reward weights and pool statistics	No	Fixed
Continuation and scoring pipeline	No	Outside the graph

The collection-copy rule is

$$\theta_{\text{old}} \leftarrow \lambda_\theta \theta_{\text{old}} + (1-\lambda_\theta)\theta, \qquad \phi_{\text{old}} \leftarrow \lambda_\phi \phi_{\text{old}} + (1-\lambda_\phi)\phi, \tag{44}$$

with $0 \le \lambda_\theta, \lambda_\phi < 1$. Zero gives a direct copy; positive values give an exponential moving average. Refresh frequency, coefficients, and optimization reuse count are separate run settings. If both fields are reset to identical functions before every simultaneous gradient evaluation, Eq. (35) applies repeatedly, eliminating direct reward dependence in that predictor gradient.

A recorded reward labels the collected state x_i^{col}. If the reward composition is L_S-Lipschitz along the segment to a rebuilt state, then $|S_{t_i}(x_i(\phi), c) - S_{t_i}(x_i^{\text{col}}, c)| \le L_S \|x_i(\phi) - x_i^{\text{col}}\|$. This bounds raw-label drift, not equality of freshly normalized weights. The COCO collection and evaluation copy settings are listed in Table 10.

C.3 Reference inversion and inference

For $0 = t_0 < t_1 < \cdots < t_{N_{\text{step}}} = T$, the stated Euler inversion is

$$x_{t_{j+1}} = x_{t_j} + (t_{j+1} - t_j) v_{\text{old}}(x_{t_j}, t_j, c). \tag{45}$$

Continuation uses decreasing levels. Here T is continuous noise level and N_{step} is solver-step count. Reference reconstruction in Figure 7 separately examines VAE error, no-mix round-trip error, and child variation.

In the stated validation/test protocol, generation starts from $x_1 \sim \mathcal{N}(0, \mathbf{I})$ with condition c; neither reference inversion nor the predictor is used. A paired held-out image is an evaluator input, not a generator input. The reference reconstruction diagnostic is a different experiment and does not substitute for noise-initialized test generation.

Reading the reconstruction diagnostic. The reference and VAE reconstruction distinguish source-image detail from autoencoder error. The no-mix round trip isolates the inversion–continuation route. Two children then show the effect of the learned perturbation around the same center. A no-mix round trip bypasses mixing rather than setting a positive bounded predictor to its minimum. Similarity is assessed together with actual variation: identical image files are not evidence of two sampled children. These stages must share the same reference and compatible solver settings.

C.4 Shared objective and child-reward measurements

At checkpoint j, with N prompt groups and $n = K_g + K_r$ children per group, the all-child statistic is

$$\overline{R}_{\text{child}}^{(j)} = \frac{1}{Nn} \sum_{p=1}^{N} \sum_{k=1}^{n} r_{p,k}^{\text{raw},(j)}. \tag{46}$$

Online collection uses the current generator, parents, and predictor, measuring the combined system. A fixed-context comparison holds the parent bank, continuation field, conditions, times, and auxiliary directions fixed while varying predictor checkpoints. Generated and reference children are reported separately and combined with their actual counts.

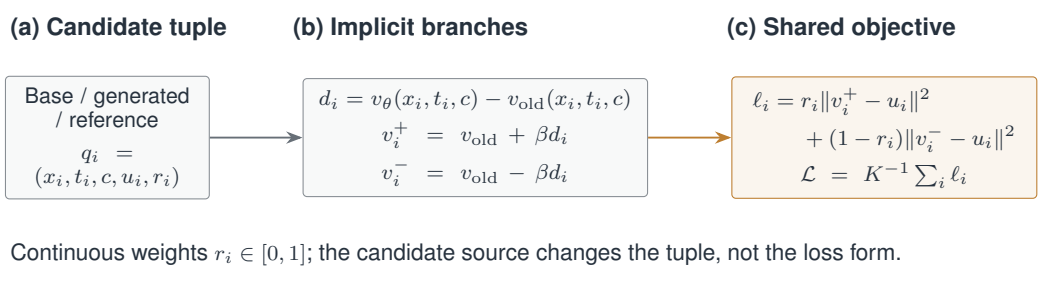

Figure 12: **One reward-weighted objective.** Candidate sources supply states, targets, and pooled reward weights. The generator and predictor optimize the same objective.

For a parent p with unperturbed-continuation reward b_p and $n_p > 0$ allocated children, define

$$\begin{aligned} \delta_p^{(j)} &= \frac{1}{n_p} \sum_{k=1}^{n_p} (r_{p,k}^{\text{raw},(j)} - b_p), \\ \delta_{p,\max}^{(j)} &= \max_{1 \le k \le n_p} r_{p,k}^{\text{raw},(j)} - b_p, \qquad \widehat{p}_p^{(j)} = \frac{1}{n_p} \sum_{k=1}^{n_p} \mathbf{1}\{r_{p,k}^{\text{raw},(j)} > b_p\}. \end{aligned} \tag{47}$$

The mean measures average benefit, the maximum measures the best discovered child, and the exceedance fraction measures how often children improve on their parent. A top-child improvement is not evidence of an all-child mean increase. For a reference parent, b_p is the no-mix continuation reward; it is distinct from the raw input-image score and from J_{in}, which uses the VAE reconstruction.

A fixed-context comparison uses the same parent bank, continuation model, collection levels, auxiliary directions, and child counts for all predictor checkpoints. Its paired differences isolate the predictor under that context law. Online statistics instead mix predictor change with generator and context change. Displacement, map variation/saturation, and round-trip error help interpret either measurement. Aggregate differences within prompt groups and training seeds; siblings are not independent runs. Accordingly, the schematic trend in Figure 3(a) illustrates the conditional regime analyzed in Theorem 3 rather than an empirically measured training curve.

D Experimental protocol and evaluation

This section specifies dataset access, collection budgets, training settings, score conventions, and aggregation.

D.1 Datasets and splits

Architecture & Interior is used as the controlled natural-image benchmark. The COCO subset contains 1,200/160/240 training, validation, and test prompts (Lin et al., 2014). Brain MRI contains 306 CN/MCI/AD images with subject-disjoint splits of 214/44/48 images. CT uses paired axial images with axial-position conditions. AID contains 10,000 aerial images from 30 scene categories (Xia et al., 2017).

D.2 Training configurations and checkpoint selection

Architecture uses different generator learning rates for NFT and the RACE family. CT uses different map initializations for RACE and RACE$^+$; at a common t, their factors $\Gamma_j^2/(1+\Gamma_j^2)$ lie approximately in $[0.059, 0.360]$ and $[0.586, 0.741]$. The CT result therefore compares full methods rather than isolating reference access alone.

Table 9: **Configurations for Architecture & Interior, brain MRI, and CT.**

Parameter	Architecture	Brain MRI	CT
LoRA learning rate (NFT)	3×10^{-4}	10^{-4}	10^{-4}
LoRA learning rate (RACE family)	10^{-4}	10^{-4}	10^{-4}
Predictor learning rate	10^{-4}	10^{-4}	10^{-4}
Optimizer-update budget	600	600	600
Groups per round	6	12	12
Sampling steps / CFG	24 / 4.5	28 / 4.5	28 / 4.5
γ_0 (RACE)	1.44	1.44	0.5
γ_0 (RACE$^+$)	1.44	1.44	1.44
δ_Γ	0.25	0.25	0.25
Configured training seeds	3	3	3

Table 10: **COCO training and evaluation configuration.**

Setting	Value
Training / validation / test prompts	1,200 / 160 / 240
Backbone	Stable Diffusion 3.5 Medium
Resolution / sampling steps / CFG	512×512 / 24 / 4.5
Optimizer / generator learning rate	AdamW / 2×10^{-5}
LoRA rank	32
Prompt groups per optimizer update	12
Optimizer updates per seed	200
Candidates per prompt	24: NFT 24; RACE $20 + 4$; RACE$^+$ $20 + 2 + 2$
Scored training candidates per full seed	$200 \times 12 \times 24 = 57{,}600$
Online reward	PickScore$/26$; per-prompt z-score normalization
Validation schedule / checkpoint	Every 10 updates; select by validation PickScore
Evaluation generation	One image per prompt; paired generation seeds across methods
Parent count / allocation rule	Top 2 of 20 bases; 2 children per parent (RACE: 4 generated; RACE$^+$: 1 generated + 1 reference child per parent)
Collection noise levels	Generated-parent children: fixed $t = 0.5$; RACE$^+$ reference children: $T = 0.25$ for RACE$^+$
Predictor learning rate	2×10^{-5}
Copy coefficients / replay count	Collection EMA: `outer_ema_decay` $= 0.5$ (official NFT schedule); evaluation EMA: 0.9; one pass per round (`inner_epochs` $= 1$)

Checkpoint selection. Validation selects checkpoints within the configured budget, and a disjoint test split supplies the final evaluation. For COCO, $\overline{s} = N_{\text{test}}^{-1} \sum_p s_p$ with $N_{\text{test}} = 240$. Equal-topic and equal-prompt averages coincide only for equal topic counts.

D.3 Reward functions and reported scores

Each table reports evaluator scores before the within-pool normalization in Eq. (13). Pool statistics and reward weights are treated as constants during optimization, and constant-score groups receive $r_i = 1/2$.

For COCO, the online training reward is $R_{\text{COCO}}(I, c) = \mathcal{E}_{\text{Pick}}(I, c)/26$. Table 1 reports PickScore on its unscaled evaluator scale, rather than the divided training reward or the within-pool weight r_i. CLIP similarity, HPSv2.1, Aesthetic Score, and ImageReward are reported separately using their respective evaluation conventions. The training-time score transformation for Architecture & Interior is specified separately from the reporting scale.

Table 11: **Evaluator roles and score conventions.** Frozen checkpoints, preprocessing, and scores before within-pool normalization.

Evaluator	Input and role	Checkpoint / preprocessing / score
$\mathcal{E}_{\text{Pick}}$	Image and prompt; preference	PickScore-v1; LAION ViT-H/14 processor. L2-normalized features; $s = \exp(\ell)\cos(f_I, f_c)$. Training: $s/26$; reported PickScore: s.
$\mathcal{T}_{\text{Arch}}$	Architecture reward scaling	No learned checkpoint: $\mathcal{T}_{\text{Arch}}(s) = s/26$. The table reports $s = 26\mathcal{T}_{\text{Arch}}(s)$, before pool normalization.
$\mathcal{E}_{\text{CLIP}}$	Image–prompt agreement	OpenAI CLIP ViT-L/14; native 224×224 processor and tokenizer. Implemented score: image–text logit divided by 100.
HPSv2.1	Image and prompt; preference	ViT-H/14; `HPS_v2.1_compressed.pt`. Bicubic resize and square padding, then encoder normalization; image–text feature dot product, no extra score scaling.
Aesthetic Score	Image only; aesthetic quality	CLIP ViT-L/14 and `sac+logos+ava1-l14-linearMSE.pth` MLP. Native CLIP processor; L2-normalized image features; direct MLP output.
ImageReward	Image and prompt; preference	`ImageReward.pt` and `med_config.json` (BLIP). RGB bicubic resize/center crop to 224; 35-token text limit. Score: $(h - 0.1671736283)/1.0333394966$.
$\mathcal{A}_{\text{MRI}}$	Axial-depth alignment; not diagnosis	UniMed-CLIP ViT-B/16 visual tower (Khattak et al., 2024) and `depth_reward_v3_frozen.pt` ridge head. Gray-to-RGB, bilinear resize to 224 and CLIP normalization. $1 - \lvert\hat{d} - d^*\rvert$, with $d^* = k/15$.
$\mathcal{F}_{\text{MRI}}$	Generated and paired MRI; fidelity	Same UniMed-CLIP backbone: $0.6\mathcal{C}_{01} + 0.4\mathcal{S}_3$. $\mathcal{C}_{01} = (1 + \cos)/2$; $\mathcal{S}_3$ averages SSIM at 224, 112, and 56 pixels, clipped to $[0, 1]$.
$\mathcal{A}_{\text{CT}}$	Axial-position alignment	DALE-CT-2S (Damron et al., 2026) with the frozen `proxy_axial_position_ridge.joblib` axial-position head; unnormalized CLS features. $\exp(-\lvert\hat{z} - z^*\rvert/0.075)$.
$\mathcal{F}_{\text{CT}}$	Generated and paired CT; fidelity	No separate learned head: grayscale SSIM at 512×512, `data_range=1`, clipped to $[0, 1]$. DALE-cosine weight 0; SSIM weight 1.
$\mathcal{E}_{\text{AID}}$	Generated image and requested scene	RemoteCLIP ViT-B/32, V3; native 224-pixel preprocessing and L2-normalized features. Target-class probability $\text{softmax}_{30}(\cos(f_I, f_{c_j})/0.02)_y$. No paired-image fidelity term.

MRI and CT use domain-specific functions $R_{\text{align}}^{(d)} = \mathcal{A}_d$ and $R_{\text{fid}}^{(d)} = \mathcal{F}_d$ in Eq. (19). Equal averaging does not calibrate different evaluators onto a common scale. For the same evaluated images,

$$\overline{R}_{\text{joint}}^{(d)} = \tfrac{1}{2}\overline{R}_{\text{align}}^{(d)} + \tfrac{1}{2}\overline{R}_{\text{fid}}^{(d)}.$$

The reported joint score is computed from the alignment and fidelity components on the same generated images, and its standard deviation is computed from per-seed joint scores. AID's generation evaluator is distinct from the original AID classification benchmark: we use frozen RemoteCLIP (Liu et al., 2024) similarities to obtain a target-class probability over the 30 scene categories.

D.4 AGGREGATION AND UNCERTAINTY

For training seed a, average over its n_p generated images for prompt p, then over N prompts:

$$\overline{s}_a = \frac{1}{N}\sum_{p=1}^{N}\frac{1}{n_p}\sum_{k=1}^{n_p} s_{a,p,k}, \qquad \overline{s} = \frac{1}{S}\sum_{a=1}^{S}\overline{s}_a, \qquad s_{\text{seed}} = \sqrt{\frac{1}{S-1}\sum_{a=1}^{S}(\overline{s}_a - \overline{s})^2}. \tag{48}$$

The standard deviation requires $S \geq 2$ independent training runs. Generation seeds within a run and correlated slices within a subject are not independent training runs. Subject-level resampling uses subjects, and paired method comparisons preserve prompt or subject correspondence. Relative changes are $100(\overline{s}_{\text{method}} - \overline{s}_{\text{NFT}})/\overline{s}_{\text{NFT}}$, as in the main medical tables; they are relative changes in a metric, not accuracy percentages.