Dense Latent Reward Alignment for Text-to-Image Flow Matching Models

Zeeshan Khan **Xin Yu** **Shizhe Chen** **Cordelia Schmid**
Inria, École normale supérieure, CNRS, PSL Research University
https://zeeshank95.github.io/deflowcritic

Abstract

Online reinforcement learning (RL) has emerged as a powerful paradigm for aligning text-to-image flow matching models with complex user intent. However, existing methods typically rely on sparse rewards computed only from final decoded images. This delays credit assignment across the denoising trajectory and makes training computationally expensive. Moreover, simply combining multiple rewards does not reliably produce a Pareto trade-off, where gains in compositional accuracy often degrade aesthetic quality. In this work, we propose **DeFlowCritic**, an efficient online RL framework based on Dense Latent Reward Alignment. Unlike prior work, our method provides dense supervision on intermediate latents, enabling early-stage guidance for the denoising trajectory. To achieve this, we introduce an internal critic that evaluates noisy latents using features extracted directly from the diffusion model, a representation inherently better suited for latent-space reward estimation than standard image encoders. We construct a high-quality reward-modeling dataset with continuous scores for critic training, capturing compositional correctness on high-quality reference images. The resulting dense reward enables efficient online RL training and inference-time scaling. Experiments show that our method achieves a superior balance between prompt faithfulness and visual quality on GenEval2 and T2I-CompBench++ benchmarks. The approach also generalizes across text-to-image flow matching models, including SD3.5-Medium, SD3.5-Large, and FLUX.1-dev.

1 Introduction

While state-of-the-art text-to-image (T2I) models based on diffusion and flow matching [9, 22, 26] achieve impressive photorealism, they consistently struggle with *compositional prompts* that require precise numeracy, attribute binding, and spatial coordination. Closing this gap is essential for moving beyond visually appealing generation toward reliable, intent-aligned image synthesis.

Online reinforcement learning (RL) has recently become one of the most effective approaches for improving compositional alignment in text-to-image generation. Methods such as DDPO [2], DPOK [10], and Flow-GRPO [24] directly optimize generative models using rewards computed from final decoded images, yielding strong gains on compositional benchmarks such as GenEval [11] and T2I-CompBench [16].

However, existing online RL methods remain expensive and suffer from sparse credit assignment. Each rollout must complete an entire denoising trajectory before receiving a terminal reward, and group-relative methods such as GRPO [34] further multiply this cost by sampling multiple trajectories per prompt. Efficiency-oriented variants [12, 44, 25] reduce overhead by simplifying the training procedure, for example by removing classifier-free guidance [13] or KL regularization [30], but these changes can degrade image quality and sample diversity. Moreover, a single terminal reward provides no direct indication of which denoising steps caused a failure. DenseGRPO [8] addresses

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

this credit-assignment issue with step-level rewards, but its reward construction remains costly: each intermediate latent is further denoised into a clean image before being evaluated by an image-space reward model.

Designing the reward itself is equally challenging. Optimizing only for compositional accuracy can produce images that satisfy the prompt while sacrificing physical plausibility or visual appeal. A natural remedy is to combine a compositional reward with human-preference models such as PickScore [19] or HPSv2 [40]; however, these rewards may differ in scale and optimization direction, making clean Pareto trade-offs difficult. GDPO [25] mitigates this issue through reward-specific normalization, but still requires careful reward-weight tuning and relies on sparse terminal feedback. Recent work has explored latent reward models that score noisy denoising states directly [41, 48]. For example, SLRM [41] trains a latent reward model from human-preference data such as Pick-a-Pic [19]. While such supervision is useful for aesthetic or preference-based quality, it does not explicitly encode compositional correctness. Learning a dense latent reward that jointly captures prompt faithfulness and visual quality therefore remains an open problem.

In this work, we propose **DeFlowCritic**, a latent critic for flow matching text-to-image models that improves compositional alignment while retaining visual quality. We first construct a new reward-modeling dataset, where each training example consists of a high-quality reference image, one correct prompt, and several systematically perturbed negative prompts that introduce controlled compositional errors. The critic model is based on the flow matching model itself, which contains stronger representations between noisy latents and texts to score noisy intermediate states. We train it with a soft classification loss to calibrate absolute reward scores, a graded ranking loss to enforce the ordering between correct and corrupted prompts, and a positive-only flow-matching auxiliary to keep the adapted backbone close to the diffusion manifold. Together, this dataset and objective yield a single latent reward that captures prompt faithfulness and favors images close to the distribution of the high-quality reference images.

We apply the resulting critic in two settings. First, we use it as a dense per-step reward in online GRPO training for flow-matching T2I models. This improves credit assignment, especially at early denoising steps, while avoiding the need to decode intermediate latents into clean images. Second, we use the same critic for inference-time scaling, where multiple latent trajectories are explored and selected according to their latent rewards, again without repeated image decoding. Experiments on GenEval2 and T2I-CompBench++ benchmarks show that DeFlowCritic improves the balance of compositional accuracy, visual quality, and efficiency in both online RL and inference-time scaling. We demonstrate its applicability to different T2I backbones, including SD3.5-Medium, SD3.5-Large and FLUX.1-dev.

Our contributions are as follows:

- We introduce DeFlowCritic, a dense latent reward model trained on a high-quality reward-modeling dataset to score intermediate denoising latents.
- We leverage the critic model to improve two orthogonal settings: as a dense reward for online GRPO fine-tuning and as a selection signal for inference-time scaling.
- We demonstrate a better balance of compositional alignment, visual quality, and efficiency than state-of-the-art methods across compositional T2I benchmarks, and validate our approach across multiple T2I backbones.

2 Related Work

Compositional text-to-image generation. Current text-to-image models based on diffusion and flow matching [36, 22, 26, 9] can synthesize highly realistic images but remain unreliable on complex, out-of-distribution compositional prompts [11, 17, 16, 15]. Training-free methods improve compositionality at inference time through layout guidance [20], attention manipulation [5], or inference-time scaling [18, 35, 27] which allocate additional test-time compute to explore and select better generations. In contrast, training-based methods directly adapt the model using supervised fine-tuning [29, 6, 39], preference optimization [32, 10, 37], or reinforcement learning (RL) [8, 24, 12], but typically rely on rewards computed from final decoded images.

Alignment for text-to-image flow-matching models. In text-to-image generation, existing alignment methods can be broadly grouped into reward backpropagation, offline preference optimization,

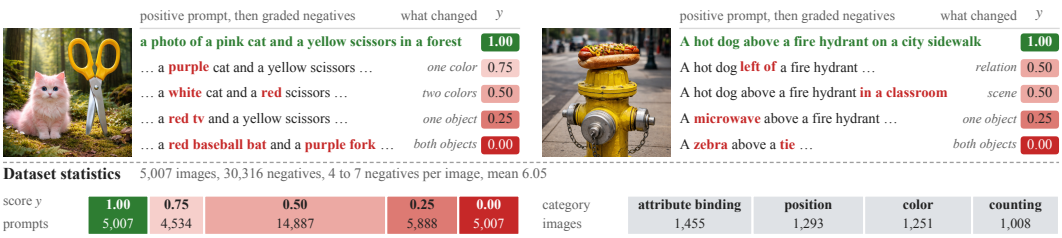

Figure 1: Samples from our GPT-IMAGE-COMPOSITION dataset and its training-set statistics. Four negatives are shown per image, with corrupted words in red and "…" marking unchanged words.

and online reinforcement learning. Reward backpropagation methods [31, 7] directly optimize differentiable reward signals, but require gradients through the sampler. Offline preference methods, including Diffusion-DPO [37] and related DPO-style objectives [32, 45, 21], avoid online rollouts but are constrained by the coverage and quality of static preference data. Online policy-gradient methods such as DDPO [2], DPOK [10], and Flow-GRPO [24] instead cast denoising as a sequential decision process and achieve strong alignment gains, but incur high computational cost and suffer from sparse rewards. Recent methods address these limitations through fewer sampling steps, modified stochastic windows, or improved advantage estimation [44, 25]. DenseGRPO [8] further tackles sparse credit assignment with step-level rewards, but is more expensive by transforming intermediate states into clean images for reward calculation. In contrast, DeFlowCritic targets this reward bottleneck by learning a critic that directly scores noisy latents, enabling efficient dense reward computation.

Reward modeling. Most reward models for image generation operate on final decoded images. Human-preference models such as PickScore [19], HPSv2/v3 [40, 28], ImageReward [43], MPS [47], VisionReward [42], and MLLM-based unified reward models [38] provide useful terminal feedback, but are typically trained on large-scale image-level comparisons and are costly to query during online RL. More importantly, terminal rewards do not indicate which denoising steps caused a failure. Step-level preference methods[21, 46] densify supervision, but perform scoring in decoded pixel space with image encoders, which is less efficient and less reliable for noisy intermediate states. Recent latent reward models, including SLRM [41] and noise-aware latent reward modeling [48], show that a diffusion backbone scores noisy latents more reliably than an external image encoder. They are trained on large-scale human preference pairs, and therefore mainly capture overall quality preference rather than fine-grained prompt faithfulness. We build on the SLRM architecture and adapt it to compositional alignment. We automatically construct a dataset of 5K reference images with lightweight human verification and improve the training objective to learn a more effective compositional critic.

3 Method

We train a latent reward model DeFlowCritic that provides dense alignment signals for the denoising trajectory of flow-matching text-to-image models, balancing compositional fidelity and visual quality. We first describe the curated reward-modeling dataset used for training and evaluation (Section 3.1). We then present the proposed model for predicting calibrated latent rewards (Section 3.2). Finally, we show how the critic supports text-to-image generation in two settings: online Flow-GRPO training with per-step rewards and reward-guided search at inference time (Section 3.3).

3.1 Reward Modeling Dataset

Existing text-to-image reward datasets, such as Pick-a-Pic [19], primarily capture image-level human preference with positive and negative images, rather than fine-grained compositional faithfulness. We therefore construct GPT-IMAGE-COMPOSITION, a reward-modeling dataset that captures both visual quality and prompt-level compositional correctness.

Positive image-text pairs. We first generate prompts from compositional templates spanning four categories: counting, color, spatial position, and attribute binding, following the compositional structure of GenEval [11]. For each prompt c^+, we use GPT-Image-1.5 to generate a high-quality reference image I. We manually check the generated data, removing approximately $25-30\%$ of the images that contain incorrect compositions, visible artifacts, or undesired styles such as poor lighting,

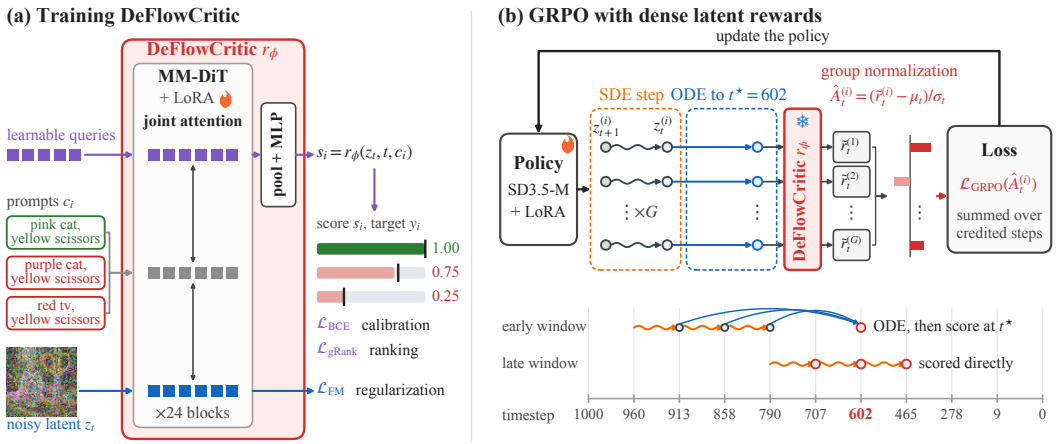

Figure 2: (a) DeFlowCritic scores a noisy latent z_t against a positive prompt and several negatives. Each latent-prompt pair (z_t, c_i) is fed to the critic as a separate input. Bars show s_i (predicted) and ticks show y_i (target). $\mathcal{L}_{\text{FM}}$ is applied to the positive prompt only. (b) In GRPO, the frozen critic provides dense rewards by scoring the latent at every SDE step, after an ODE rollout for early steps.

overly dark scenes, or cartoon-like rendering. The positive pair (c^+, I) is assigned the ground-truth score $y = 1$.

Negative image-text pairs. Based on the positive image-text pair (c^+, I), we construct up to seven negative prompts by systematically perturbing one or more attribute slots in c^+, including object identity, counts, attributes, and spatial relations. We assign each negative pair (c_*^-, I) a continuous compositional score $y \in [0, 1)$ measuring agreement with the fixed reference image. Lower scores reflect increasingly severe compositional mismatches. For example, for $c^+ =$ "a photo of a pink cat and a yellow scissors in a forest" (Figure 1, left), changing one color ("a purple cat") gives $y = 0.75$, changing both colors ("a white cat and a red scissors") gives $y = 0.5$, replacing one object and its color ("a red tv") gives $y = 0.25$, and replacing both objects gives $y = 0$.

Dataset summary. As shown in Fig. 1, the resulting dataset contains 5K visually clean reference images, each paired with one positive prompt and 4 to 7 controlled negative prompts. These negative prompts introduce different levels of compositional violations, enabling the critic to learn fine-grained distinctions between well-aligned and partially mismatched image-prompt pairs. We split the dataset into training and test sets with 5,007 and 240 examples, respectively. The evaluation split is balanced across the query categories (counting, color, attribute binding, and spatial position).

3.2 DeFlowCritic: Dense Latent Reward Model

Let z_t denote the noisy latent at denoising timestep t, and c the text prompt. The latent reward model r_ϕ predicts a scalar reward $s_i = r_\phi(z_t, t, c_i)$ that measures the quality of the latent-prompt pair.

3.2.1 Model Architecture

We build the reward model on top of the SD3.5-Medium MM-DiT backbone [9] following SLRM [41], which jointly represents noisy image latents and text tokens and therefore provides features well suited for latent-space alignment. We augment the frozen backbone with two trainable components: (i) LoRA [14] adapters applied to the attention projection and feed-forward layers in each MM-DiT block, and (ii) N_q learnable query tokens concatenated with the text stream inside the MM-DiT joint-attention blocks, where they attend to both text and image-latent tokens to summarize cross-modal alignment information.

After the final transformer block, the query tokens are mean-pooled and passed through a lightweight MLP head with sigmoid activation:

$$r_\phi(z_t, t, c) = \sigma\left(\text{MLP}\left(\frac{1}{N_q}\sum_{k=1}^{N_q} q_k^{(L)}(z_t, t, c)\right)\right), \tag{1}$$

where $q_k^{(L)}$ denotes the k-th query token after the last block.

3.2.2 Training Objective

Let s_i denote the predicted reward and let $y_i \in [0,1]$ denote the continuous score defined in Section 3.1. We train the critic with a composite objective on the constructed dataset:

$$\mathcal{L}_{r_\phi} = \mathcal{L}_{\mathrm{BCE}} + \lambda_{\mathrm{rank}}\mathcal{L}_{\mathrm{gRank}} + \lambda_{\mathrm{FM}}\mathcal{L}_{\mathrm{FM}}. \tag{2}$$

The three terms play complementary roles: score calibration fixes the absolute reward scale, compositional ranking enforces the ordering among prompts paired with the same image, and flow-matching regularization keeps the adapted backbone aligned with the denoising manifold. Figure 2(a) shows how DeFlowCritic is trained with the three objectives.

Score prediction loss $\mathcal{L}_{\mathrm{BCE}}$. We use soft binary cross-entropy on the continuous targets:

$$\mathcal{L}_{\mathrm{BCE}} = -\mathbb{E}_i \left[y_i \log s_i + (1 - y_i) \log(1 - s_i) \right]. \tag{3}$$

This term allows the critic to assign more calibrated scores to latent-prompt pairs with different levels of compositional agreement.

Ranking loss $\mathcal{L}_{\mathrm{gRank}}$. For positive and negative prompts paired with the same reference image, relative ordering provides strong supervision to measure the quality. We therefore apply a graded RankNet-style pairwise loss [4]:

$$\mathcal{L}_{\mathrm{gRank}} = \mathbb{E}_{\mathrm{rows}} \frac{\sum_{(i,j):\, y_i > y_j} (y_i - y_j)\, \mathrm{softplus}\Big(-\frac{s_i - s_j}{\tau}\Big)}{\sum_{(i,j):\, y_i > y_j} (y_i - y_j)}, \qquad \tau = 0.15. \tag{4}$$

Flow-matching regularization $\mathcal{L}_{\mathrm{FM}}$. Since the reward model is trained on a small high-quality dataset, preference losses alone may shift the adapted backbone too far from the denoising features. We therefore add a flow-matching auxiliary loss on positive prompts as regularization:

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{t, z_t, c^+} \left[w(t) \left\| v_\phi(z_t, t, c^+) - u_t(z_t \mid z_0) \right\|_2^2 \right], \qquad w(t) = 1 + \alpha_{\mathrm{ramp}} \frac{t - t_{\min}}{t_{\max} - t_{\min}}. \tag{5}$$

Here u_t is the target conditional velocity for the clean reference latent z_0, $\alpha_{\mathrm{ramp}} = 2.0$, and timesteps are sampled from $[t_{\min}, t_{\max}] = [50, 950]$ to avoid the extreme ends of the noise schedule. The ramp up-weights noisier timesteps, where velocity prediction provides a stronger constraint on the latent representation. Applying this auxiliary only to the positive prompt c^+ avoids training the model to denoise a clean image under an intentionally mismatched text condition.

3.3 Improving T2I Generation with DeFlowCritic

Dense latent rewards for Flow-GRPO. We use the same rollout and policy-update machinery as Flow-GRPO [24]: for each prompt, the policy samples a group of stochastic trajectories within the SDE window, recomputes per-step log-probabilities, and optimizes the clipped GRPO surrogate over the credited denoising steps.

Our modification is restricted to advantage construction. Standard Flow-GRPO assigns each trajectory a single terminal reward, normalizes these rewards across the group, and broadcasts the same advantage to every credited timestep. In contrast, we query the frozen latent critic at each credited timestep and normalize rewards across the group separately for each t, yielding timestep-specific advantages, as illustrated in Fig. 2(b).

Because the critic is best calibrated near a trusted low-noise timestep $t^\star$ (we use $t^\star = 602$), we do not score high-noise latents directly. Steps that land on $t \in \{707, 602, 465\}$ are close to this range and are scored directly. Each earlier credited latent $z_t^{(i)}$ is first mapped to $t^\star$ by a short deterministic Euler rollout and then evaluated by the critic and we compute the group-relative advantage for each timestep t:

$$\tilde{r}_t^{(i)} = r_\phi\Big(\mathrm{ODE}_{t \to t^\star}(z_t^{(i)}),\, t^\star,\, c\Big), \quad \hat{A}_t^{(i)} = \frac{\tilde{r}_t^{(i)} - \mu_t}{\sigma_t + \epsilon_{\mathrm{std}}}, \quad \mu_t, \sigma_t = \mathrm{mean}, \mathrm{std}_{i=1}^{G}\left(\tilde{r}_t^{(i)}\right). \tag{6}$$

We plug $\hat{A}_t^{(i)}$ into the unchanged Flow-GRPO surrogate. This replaces a terminal image-level reward broadcast with step-specific latent feedback, while avoiding VAE decoding and external pixel-space reward-model calls inside the dense reward loop.

Table 1: Comparison of Flow-GRPO variants on GenEval2 and T2I-CompBench++ benchmarks. All models are built upon the SD3.5-M [9] baseline. We report compositional fidelity, image quality measured by LAION Aesthetic [33] and HPSv3 [28], and relative training time.

Models	GenEval2			T2I-CompBench++			Time ↓
	Comp. ↑	Aesth. ↑	HPSv3 ↑	Comp. ↑	Aesth. ↑	HPSv3 ↑	
SD3.5-M [9]	18.10	5.51	6.66	52.37	5.46	7.76	–
Flow-GRPO (GenEval) [24]	23.40	5.41	7.13	59.44	5.27	9.26	1×
fast Flow-GRPO (GenEval)	22.70	5.17	4.45	58.64	5.18	5.71	0.1×
fast Flow-GRPO (GenEval + Aesth.)	12.50	6.91	6.89	55.24	6.96	8.72	0.12×
fast Flow-GRPO (DeFlowCritic)	21.44	5.66	8.07	59.37	5.69	10.34	0.1×

Guidance for inference-time scaling. We use the frozen critic for test-time trajectory selection: instead of sampling a single denoising trajectory, we denoise K independent candidates only up to the scoring timestep $t^\star$, rank them with r_ϕ, and complete only the best candidate with deterministic ODE steps. Losing candidates are thus pruned before being fully denoised or decoded, whereas an image-space reward model must finish and decode all K samples before it can rank them.

4 Experiments

4.1 Implementation Details

DeFlowCritic training. We train DeFlowCritic on our reward-modeling dataset with a frozen SD3.5-Medium MM-DiT backbone. The trainable parameters are LoRA adapters, a reward head, and $N_q = 16$ query tokens injected into every transformer block. We use LoRA rank 32 with $\alpha = 64$, train for up to 20 epochs, and sample timesteps from $[50, 950]$. The reward head and LoRA adapters use learning rates 1×10^{-4} and 3×10^{-5}, respectively. We set $\lambda_{\text{rank}} = 0.2$ and $\lambda_{\text{FM}} = 1$ in Equation (2). We stop training once ranking accuracy on a held-out split plateaus, which takes around 8 H100-hours (Section C).

GRPO training. We fine-tune SD3.5-Medium with Flow-GRPO using DeFlowCritic as the dense latent reward. Training uses GenEval prompts, group size $G = 22$, rollout length $T = 10$, and resolution 512×512; evaluation uses $T_{\text{eval}} = 40$ denoising steps. Dense rewards are computed with ODE-lookahead in Equation (6), using trusted timestep $t^\star = 602$ and scoring steps $\{465, 602, 707\}$. We disable classifier-free guidance and KL regularization during training, and use AdamW with learning rate 3×10^{-4}. GRPO training takes approx. 90 H100 hours.

Evaluation. Our policy is trained on GenEval prompts, so GenEval-1 is in-domain and close to saturated. We use it only to compare with the published results from other latent reward models (Table 3), and report our main results zero-shot on GenEval2 [17] and T2I-CompBench++ [15], which neither the critic nor the policy sees during training. For compositional fidelity, we report the aggregate compositional score from each benchmark. For image quality, we report LAION Aesthetic Score [33] and HPSv3 [28].

4.2 Comparison with State-of-the-Art Methods

Baselines. We compare SD3.5-Medium variants that share the same backbone but differ in post-training reward and regularization. *SD3.5-M* [9] is the pretrained model. *Flow-GRPO (GenEval)* [24] uses terminal GenEval rewards with CFG and KL regularization. *Fast Flow-GRPO (GenEval)* removes CFG and KL for faster training. *Fast Flow-GRPO (GenEval + Aesth.)* adds a terminal LAION aesthetic reward to the fast setting. Our method uses the same fast GRPO setting but replaces terminal rewards with dense latent rewards from DeFlowCritic.

Comparison with GRPO methods. Table 1 reports results on GenEval2 and T2I-CompBench++. DeFlowCritic matches standard Flow-GRPO on T2I-CompBench++ compositional fidelity (59.37 vs. 59.44) and remains competitive on GenEval2 (21.44 vs. 23.40), while improving image-quality metrics on both benchmarks. Compared with Flow-GRPO, DeFlowCritic improves HPSv3 from

Table 2: Category-level comparison on T2I-CompBench++ [15]. Compositional fidelity is reported.

Models	T2I-CompBench++ Categories						
	Color	Shape	Texture	2D-Spatial	3D-Spatial	Numeracy	Non-Spatial
SD3.5-M [9]	79.94	56.69	73.38	28.50	37.39	59.27	31.46
Flow-GRPO (GenEval) [24]	83.79	61.30	72.36	54.47	44.71	67.52	31.95
fast Flow-GRPO (GenEval)	82.70	64.90	72.60	51.10	41.00	67.10	31.00
fast Flow-GRPO (GenEval + Aesth.)	81.50	57.40	67.40	42.10	43.90	64.10	30.30
fast Flow-GRPO (DeFlowCritic)	85.38	62.66	74.13	50.27	45.38	66.46	31.37

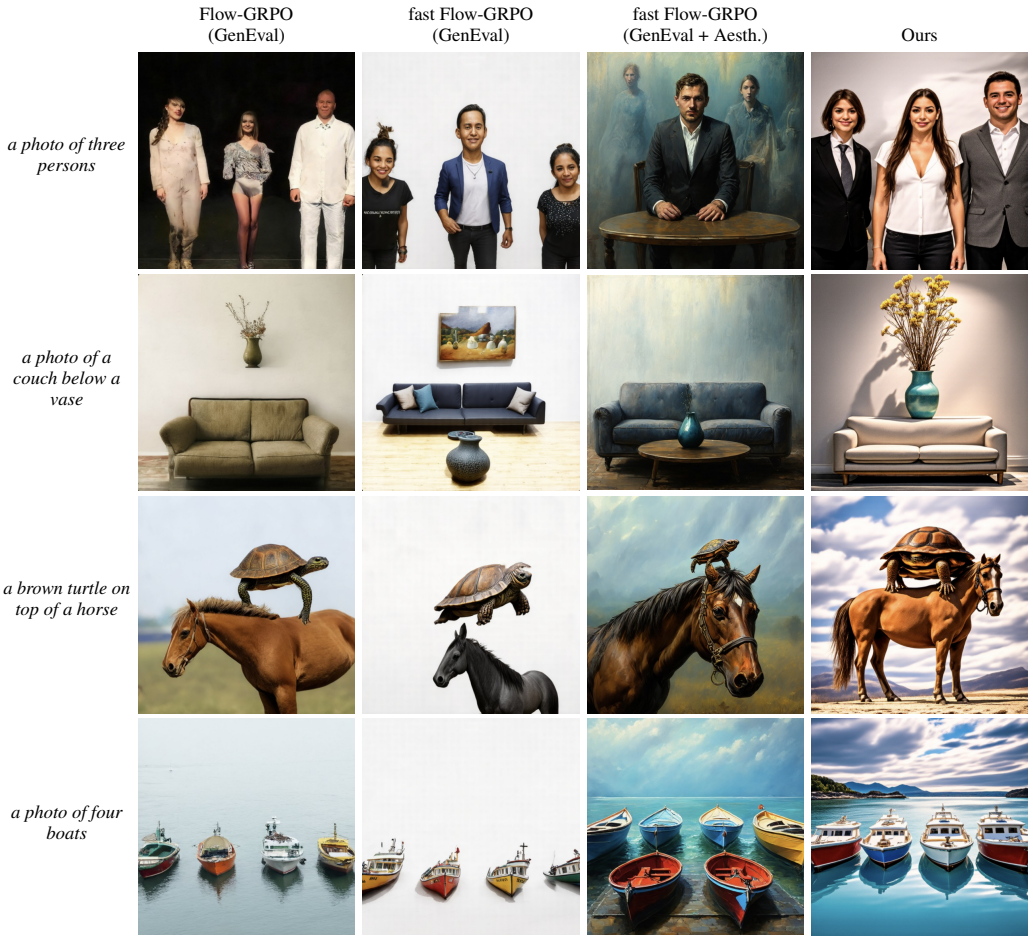

Figure 3: **Qualitative comparisons of Flow-GRPO variants on GenEval and GenEval2 prompts.** Our model achieves a superior balance between visual quality and compositional accuracy.

7.13 to 8.07 on GenEval2 and from 9.26 to 10.34 on T2I-CompBench++. Compared with Fast Flow-GRPO, the gains are larger, indicating that dense latent rewards help stabilize fast training when CFG and KL are removed. The aesthetic-reward baseline improves LAION Aesthetic but significantly reduces GenEval2 compositional accuracy, suggesting that naive terminal multi-reward mixing does not yield a clean quality-composition trade-off. DeFlowCritic obtains a stronger trade-off while achieving a $10\times$ training speed-up over standard Flow-GRPO at matched group size and inference schedule.

Table 2 breaks down T2I-CompBench++ performance by category. DeFlowCritic achieves the best scores on color, texture, and 3D-spatial relations, and remains close to Flow-GRPO on numeracy, 2D-spatial, and non-spatial categories. These gains are consistent with our reward data construction, where negatives are created by controlled perturbations of object attributes, bindings, and spatial relations.

Qualitative results. Figure 3 compares generations from Flow-GRPO, Fast Flow-GRPO, Fast Flow-GRPO (multi-reward), and DeFlowCritic. Flow-GRPO often captures the requested composition but can produce less realistic images such as floating objects. Fast Flow-GRPO improves efficiency but further degrades lighting, texture, and overall visual quality. Multi-reward Fast Flow-GRPO improves visual quality by incorporating aesthetic rewards, but often sacrifices compositional accuracy. In contrast, DeFlowCritic better preserves prompt-relevant details, while producing cleaner and more visually appealing images.

Comparison with latent reward models. We next compare DeFlowCritic with existing latent reward models by evaluating the policies obtained by fine-tuning/RL training with each of them. LPO [48], SLRM [41], and DINA-LRM [23] learn from 0.17M to 0.8M human-preference pairs, while DeFlowCritic learns from only $5{,}007$ curated images with rule-graded caption perturbations. Table 3 reports compositional accuracy on GenEval-1 and preference scores (HPSv2.1 and HPSv3) on Pick-a-Pic prompts. DeFlowCritic reaches 90.65 on GenEval-1, about 22 points above the prior methods. On Pick-a-Pic, the home domain of preference rewards, it stays comparable on HPSv3 (9.62 vs. 9.69 and 9.89), while DINA-LRM and SLRM lead on HPSv2.1. DeFlowCritic thus reaches much higher compositional accuracy, with comparable visual quality while using substantially less reward data and no human-preference annotation.

Table 3: Comparison with latent reward models. Reward data is the training set of each reward model: Pick-a-Pic (LPO, SLRM) or HPDv3 (DINA-LRM). Rows marked † are measured by us.

Method	Backbone	Reward data	GenEval-1	HPSv2.1	HPSv3
SD3-M	SD3-M	–	67.29	28.74	–
LPO [48]	SD3-M	0.17M pairs	68.63	30.44	–
LPO, from [41]	SDXL	0.17M pairs	59.85	31.89	9.10
SD3.5-M	SD3.5-M	–	55.34	27.15	8.29
SLRM [41]	SD3.5-M	0.58M pairs	68.93	32.01	9.89
DINA-LRM† [23]	SD3.5-M	0.8M pairs	68.15	32.41	9.69
DeFlowCritic†	SD3.5-M	5K images	90.65	30.71	9.62

Inference-time scaling. We further evaluate the frozen critic as a test-time selection signal on GenEval2, using best-of-K selection at $t^\star = 602$ (Section 3.3). Table 4 compares this with best-of-K selection by CLIPScore, which must denoise and decode all K candidates before it can rank them. With $K = 8$, DeFlowCritic reaches 24.09 versus 21.95 for CLIPScore, at 444 instead of 640 model calls and with higher HPSv3.

Table 4: Inference-time scaling on GenEval2. Calls are denoiser passes per image, on one H100.

Method	Calls	sec/im	GenEval2	HPSv3
SD3.5-M (base)	80	1.4	18.10	6.66
+ CLIPScore BO-8	640	7.3	21.95	6.58
+ DeFlowCritic BO-8 (ours)	444	5.1	24.09	7.32

4.3 Ablation Studies

We ablate the reward model r_ϕ design choices: the backbone, the trainable components, the training objective, and the scoring timestep. We report ranking accuracy ($\mathrm{acc}_{\mathrm{rank}}$), the percentage of evaluation rows where the positive prompt is scored above all negatives paired with the same reference image. We also report pairwise accuracy ($\mathrm{acc}_{\mathrm{pair}}$), the percentage of prompt pairs for the same image with different target scores that the critic puts in the right order. Both are measured on our held-out test split (Section 3.1): 240 reference images that the critic never sees in training, with $1{,}422$ negatives built by the same rules as in the training set.

Table 5: Ranking accuracy ($\mathrm{acc}_{\mathrm{rank}}$) (%) of reward models on noisy intermediate latents.

Method	Reward input	Fine-tuned	t=465	t=602	t=707	t=780
Qwen-3-VL (zero-shot)	Decoded image	×	58.41	22.66	12.73	5.23
Qwen-3-VL (fine-tuned, noisy images)	Decoded image	✓	92.16	85.07	77.64	67.22
DeFlowCritic	Latent	✓	97.08	96.67	95.83	91.03

Backbone choice: SD3.5 versus VLM. A natural alternative to DeFlowCritic is to decode noisy latents into noisy images and score them with a vision-language model. Table 5 compares zero-shot Qwen-3-VL [1], Qwen-3-VL fine-tuned on the same reward dataset, and our SD3.5-based critic

across four timesteps. Zero-shot Qwen-3-VL rapidly fails as noise increases, dropping from 58.41% at $t = 465$ to 5.23% at $t = 780$. Fine-tuning improves robustness, but still falls to 67.22% at high noise. In contrast, DeFlowCritic remains accurate across the trajectory, achieving 97.08%, 96.67%, 95.83%, and 91.03%. This supports the use of the generator backbone as the reward representation: its features are pretrained on the same noisy latent space visited during RL, whereas VLM features remain mismatched to intermediate denoising states.

LoRA and query tokens. Table 6 isolates the trainable components. LoRA is the primary contributor: without backbone adaptation, the query tokens and reward head alone reach less than 30% ranking accuracy and the downstream policy collapses. A critic that cannot rank also cannot guide training: without LoRA, the policy falls far below the base model on GenEval2 (3.33 and 2.38, against 18.10).

Table 6: LoRA and query tokens at $t = 465$, with downstream GRPO on GenEval2.

Trainable parts	Params	$\mathrm{acc}_{\mathrm{rank}}$	GenEval2	HPSv3
Head only	9.45M	29.17	3.33	-2.46
Queries + head	9.48M	27.50	2.38	-9.89
LoRA + head	51.3M	95.00	21.58	7.62
LoRA + queries + head	51.4M	97.08	21.44	8.07

Training objective. We next ablate the critic objective while keeping the architecture fixed. Table 7 compares combinations of the training objectives, including binary BCE, which collapses all negatives to label 0; soft BCE, which preserves the absolute continuous compositional scores; and the graded ranking loss which preserves the relative ranks. Binary BCE achieves high ranking accuracy, but much lower pairwise accuracy, indicating that binary labels discard severity information among negative prompts. Soft BCE improves pairwise accuracy by supervising the continuous score scale. Combining soft BCE, the graded ranking loss, and the flow-matching term gives both strong ranking and pairwise accuracy. The three losses therefore provide complementary supervision, and the full objective keeps all of them.

Table 7: Loss ablation at $t = 465$ on the test set. All variants use the same training step.

Loss	$\mathrm{acc}_{\mathrm{rank}}$	$\mathrm{acc}_{\mathrm{pair}}$
Binary BCE	95.85	73.71
Soft BCE only	94.17	98.93
Ranking only	93.33	99.06
Soft BCE + ranking	95.42	99.06
Soft BCE + ranking + FM	97.08	99.05

Scoring timestep and lookahead. Table 8 ablates the use of lookahead and the choice of $t^\star$. Without lookahead, the critic must evaluate highly noisy latents, which weakens the compositional supervision and reduces GenEval2 performance. Propagating farther to $t^\star = 278$ slightly improves GenEval2, but introduces more computational effort. We therefore choose $t^\star = 602$ empirically, as a compute-performance trade-off.

Table 8: ODE lookahead and the choice of $t^\star$.

Noisy steps	$t^\star$	GenEval2	Rel. time
No lookahead	–	20.41	$1.00\times$
Lookahead	278	22.06	$1.72\times$
Lookahead	602	21.44	$1.26\times$

Critic's preference on image quality. DeFlowCritic receives no explicit supervision for image quality. Each negative keeps the reference image and corrupts only its prompt, so the critic never sees one image labeled as worse than another. To understand what the critic rewards beyond composition, we take its held-out test prompts and collect one image per prompt from three sources: the GPT-Image-1.5 reference images, SD3.5-M, the base model of both the critic and the policy; and our GRPO policy. A VLM judge then keeps only the images it rates as fully correct, leaving 200 images per source that all match their prompts and differ only in how they look. We encode each image, noise it to $t = 465$, and score it against its prompt with the frozen critic (Table 9). Although every image is correct, the critic does not score them equally: the references receive 0.97 and our policy 0.96, while SD3.5-M receives only 0.82. The critic therefore favors images from a distribution close

Table 9: Critic scores at $t = 465$ on images that are all compositionally correct. 200 images per pool.

Image pool	Correct	Critic	HPSv3
GPT-Image-1.5 (reference)	100%	0.97	10.93
Our GRPO policy	100%	0.96	10.71
SD3.5-M (base model)	100%	0.82	9.66



Figure 4: FLUX.1-dev before (top) and after (bottom) 240 GRPO steps with DeFlowCritic, evaluated on GenEval2 prompts.

to its references, even over images from its own base model. GRPO moves the policy toward this distribution, and HPSv3 rises from 9.66 to 10.71 along the way.

4.4 Generalization Across T2I Models

Table 10: DeFlowCritic GRPO improves different T2I flow matching models.

Generator	GenEval2	HPSv3
SD3.5-L	19.14	3.93
+ GRPO, 440 steps	25.83	8.29
FLUX.1-dev	21.40	7.89
+ GRPO, 240 steps	28.49	9.02

We evaluate DeFlowCritic across T2I models in two settings. When generators share the same VAE latent space, the same critic can be reused without retraining. We demonstrate this by applying our SD3.5-Medium critic, built on a 2.5B backbone, to guide an 8B SD3.5-Large policy. When a generator uses a different latent space, a new critic is required, but the same training approach remains applicable. For FLUX.1-dev [3], we train a FLUX-based critic using the same 5K reference images and training objective. Training takes 837 steps, approximately 5 H100-hours (Section C). As shown in Table 10, DeFlowCritic-guided GRPO improves GenEval2 scores from 19.14 to 25.83 on SD3.5-Large and from 21.40 to 28.49 on FLUX.1-dev, and increases HPSv3 as well. Figure 4 shows examples where base FLUX misses an object, a count, or a relation, and GRPO fixes the compositions while retaining the visual quality.

5 Conclusion

We presented **DeFlowCritic**, a dense latent reward model for efficient compositional alignment of text-to-image flow-matching models. Instead of relying on sparse terminal rewards from decoded images, DeFlowCritic scores noisy intermediate latents directly using the generator's own representation space, enabling timestep-specific feedback during online GRPO training. Experiments on GenEval2 and T2I-CompBench++ show that the method preserves the compositional gains of Flow-GRPO, improves image-quality metrics such as HPSv3, and substantially reduces training cost; the same frozen critic also improves inference-time trajectory selection without additional training. These results suggest that generator-native latent critics are a promising direction for scalable, reward-efficient alignment of diffusion and flow-matching generators.

Limitations. Our experiments focus on compositional T2I; transfer to higher resolutions and video or image-to-video generation remains open. The method also depends on the empirical finding that early-to-mid denoising latents are effective scoring states, so other noise schedules may require adaptive selection of $t^\star$.

Broader impact. While stronger compositional control can benefit creative and design applications, it may also increase misuse risks for deceptive or harmful synthetic imagery; responsible deployment should pair latent reward alignment with content filtering, dataset auditing, and provenance mechanisms.

Acknowledgments and Disclosure of Funding

This work was granted access to the HPC resources of IDRIS under the allocation AD011014688R2 made by GENCI. It was funded in part by the French government under management of Agence Nationale de la Recherche as part of the "France 2030" program, reference ANR-23-IACL-0008 (PR[AI]RIE-PSAI project), and Paris Île-de-France Région in the frame of the DIM AI4IDF. Cordelia Schmid would like to acknowledge the support by the Körber European Science Prize.

References

[1] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[2] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2024.

[3] Black Forest Labs. FLUX.1-dev. `https://github.com/black-forest-labs/flux`, 2024.

[4] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *International Conference on Machine Learning (ICML)*, 2005.

[5] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-Excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics*, 42(4):1–10, 2023.

[6] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024.

[7] Kevin Clark, Paul Vicol, Kevin Swersky, and David J. Fleet. Directly fine-tuning diffusion models on differentiable rewards. In *International Conference on Learning Representations (ICLR)*, 2024.

[8] Haoyou Deng, Keyu Yan, Chaojie Mao, Xiang Wang, Yu Liu, Changxin Gao, and Nong Sang. DenseGRPO: From sparse to dense reward for flow matching model alignment. In *International Conference on Learning Representations (ICLR)*, 2026.

[9] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *International Conference on Machine Learning (ICML)*, 2024.

[10] 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. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[11] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. GenEval: An object-focused framework for evaluating text-to-image alignment. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023.

[12] Xiaoxuan He, Siming Fu, Yuke Zhao, Wanli Li, Jian Yang, Dacheng Yin, Fengyun Rao, and Bo Zhang. TempFlow-GRPO: When timing matters for GRPO in flow models. In *International Conference on Learning Representations (ICLR)*, 2026.

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

[14] 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 (ICLR)*, 2022.

[15] Kaiyi Huang, Chengqi Duan, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2I-CompBench++: An enhanced and comprehensive benchmark for compositional text-to-image generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(5):3563–3579, 2025.

[16] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2I-CompBench: A comprehensive benchmark for open-world compositional text-to-image generation. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023.

[17] Amita Kamath, Kai-Wei Chang, Ranjay Krishna, Luke Zettlemoyer, Yushi Hu, and Marjan Ghazvininejad. GenEval 2: Addressing benchmark drift in text-to-image evaluation. *arXiv preprint arXiv:2512.16853*, 2025.

[18] Jaihoon Kim, Taehoon Yoon, Jisung Hwang, and Minhyuk Sung. Inference-time scaling for flow models via stochastic generation and rollover budget forcing. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[19] 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. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[20] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. GLIGEN: Open-set grounded text-to-image generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[21] Zhanhao Liang, Yuhui Yuan, Shuyang Gu, Bohan Chen, Tiankai Hang, Mingxi Cheng, Ji Li, and Liang Zheng. Aesthetic post-training diffusion models from generic preferences with step-by-step preference optimization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[22] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.

[23] Gongye Liu, Bo Yang, Yida Zhi, Zhizhou Zhong, Lei Ke, Didan Deng, Han Gao, Yongxiang Huang, Kaihao Zhang, Hongbo Fu, and Wenhan Luo. Beyond VLM-based rewards: Diffusion-native latent reward modeling. In *International Conference on Machine Learning (ICML)*, 2026.

[24] 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. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[25] Shih-Yang Liu, Xin Dong, Ximing Lu, Shizhe Diao, Peter Belcak, Mingjie Liu, Min-Hung Chen, Hongxu Yin, Yu-Chiang Frank Wang, Kwang-Ting Cheng, Yejin Choi, Jan Kautz, and Pavlo Molchanov. GDPO: Group reward-decoupled normalization policy optimization for multi-reward RL optimization. In *International Conference on Machine Learning (ICML)*, 2026.

[26] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations (ICLR)*, 2023.

[27] Nanye Ma, Shangyuan Tong, Haolin Jia, Hexiang Hu, Yu-Chuan Su, Mingda Zhang, Xuan Yang, Yandong Li, Tommi Jaakkola, Xuhui Jia, and Saining Xie. Scaling inference time compute for diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[28] Yuhang Ma, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. HPSv3: Towards wide-spectrum human preference score. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

[29] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[30] Jan Peters, Katharina Mülling, and Yasemin Altun. Relative entropy policy search. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2010.

[31] Mihir Prabhudesai, Anirudh Goyal, Deepak Pathak, and Katerina Fragkiadaki. Aligning text-to-image diffusion models with reward backpropagation. *arXiv preprint arXiv:2310.03739v2*, 2023.

[32] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[33] Christoph Schuhmann and Romain Beaumont. LAION-Aesthetics. `https://laion.ai/blog/laion-aesthetics/`, 2022.

[34] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[35] 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. In *International Conference on Machine Learning (ICML)*, 2025.

[36] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations (ICLR)*, 2021.

[37] 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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[38] Yibin Wang, Yuhang Zang, Hao Li, Cheng Jin, and Jiaqi Wang. Unified reward model for multimodal understanding and generation. *arXiv preprint arXiv:2503.05236*, 2025.

[39] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations (ICLR)*, 2022.

[40] 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.

[41] Xiaole Xian, Xilin He, Wenting Chen, Wenshuang Liu, Wenqi Mu, Yancheng He, Liang Li, Yi Zhang, and Xiangyu Yue. Consistent noisy latent rewards for trajectory preference optimization in diffusion models. In *International Conference on Learning Representations (ICLR)*, 2026.

[42] Jiazheng Xu, Yu Huang, Jiale Cheng, Yuanming Yang, Jiajun Xu, Yuan Wang, Wenbo Duan, Shen Yang, Qunlin Jin, Shurun Li, Jiayan Teng, Zhuoyi Yang, Wendi Zheng, Xiao Liu, Dan Zhang, Ming Ding, Xiaohan Zhang, Shiyu Huang, Xiaotao Gu, Minlie Huang, Jie Tang, and Yuxiao Dong. VisionReward: Fine-grained multi-dimensional human preference learning for image and video generation. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2026.

[43] 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 (NeurIPS)*, 2023.

[44] Renye Yan, Jikang Cheng, Yaozhong Gan, Shikun Sun, You Wu, Yunfan Yang, Ling Liang, Jinlong Lin, Yeshuang Zhu, Jie Zhou, Jinchao Zhang, Junliang Xing, Yimao Cai, and Ru Huang. Entropy-adaptive diffusion policy optimization with dynamic step alignment. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

[45] Kai Yang, Jian Tao, Jiafei Lyu, Chunjiang Ge, Jiaxin Chen, Weihan Shen, Xiaolong Zhu, and Xiu Li. Using human feedback to fine-tune diffusion models without any reward model. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[46] Jiacheng Zhang, Jie Wu, Weifeng Chen, Yatai Ji, Xuefeng Xiao, Weilin Huang, and Kai Han. Align video diffusion model with online video-centric preference optimization. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2026.

[47] Sixian Zhang, Bohan Wang, Junqiang Wu, Yan Li, Tingting Gao, Di Zhang, and Zhongyuan Wang. Learning multi-dimensional human preference for text-to-image generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[48] Tao Zhang, Cheng Da, Kun Ding, Huan Yang, Kun Jin, Yan Li, Tingting Gao, Di Zhang, Shiming Xiang, and Chunhong Pan. Diffusion model as a noise-aware latent reward model for step-level preference optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

A Scoring Rules

Prompt specifications. Each prompt is generated from a structured specification of its objects, attributes, counts, relation, and scene based on GenEval-1 compositions. The four categories use different parts of this specification. A counting prompt fixes the number of instances of one object class, a color prompt gives one object a color, an attribute-binding prompt names two objects with one color each, and a position prompt places two objects in a spatial relation. Every prompt also carries a scene phrase, such as *at a train station*. During critic training, half of the batches use the prompts with their scene phrase and half without it, so the critic sees both object-specific short prompts and longer prompts that adds a background.

Negatives and their scores. A negative prompt applies one corruption to this specification and is created with the same template, so it differs from the positive only in the corrupted part. The corruption type alone determines its score: the more of the prompt that remains correct for the reference image, the higher the score. The labels are therefore fully rule-based, with no per-sample human or LLM judgment. Table 11 lists every corruption type with a real example. Three principles set the scores:

- **Count errors are graded by their number.** A count that is off by one, two, or three instances scores 0.75, 0.5, or 0.25.
- **Attribute errors keeping the objects correct.** A wrong color, relation, or scene scores 0.5, and 0.75 when only one of two colors in attribute binding is wrong.
- **Object errors are the most severe.** Replacing one of two objects scores 0.25, and replacing every object scores 0.

Table 11: Corruption types with real examples from the dataset.

Category	Corruption	Example negative	y	Negatives
Counting *Four trains at a train station*	Count off by one	Three trains at a train station	0.75	1,624
	Count off by two	Two trains at a train station	0.50	1,008
	Count off by three	One train at a train station	0.25	392
Color *an orange oven in a modern kitchen*	Wrong color	a blue oven in a modern kitchen	0.50	3,753
Attribute binding *A yellow broccoli and a red tennis racket*	One color wrong	A purple broccoli and a red tennis racket	0.75	2,910
	Both colors wrong	A black broccoli and a brown tennis racket	0.50	1,455
	One object	A brown giraffe and a red tennis racket	0.25	2,910
	All objects	A pink baseball bat and a brown tv remote	0.00	1,455
Position *A fork above a cow*	Wrong relation	A fork left of a cow	0.50	3,879
	One object replaced	A snowboard above a cow	0.25	2,586
	Both objects replaced	A hair drier above a potted plant	0.00	1,293
All categories *Three trains at a train station*	Scene replaced	Three trains in a kitchen	0.50	4,792
	Only object replaced	Three cats at a train station	0.00	2,259

Dataset statistics. Table 12 breaks the dataset down by category. The $5{,}007$ training rows carry $30{,}316$ negatives, between 4 and 7 per image and 6.05 on average. Counting and color rows have about five negatives each, and the two-object categories about seven, because they have more parts to corrupt. Every image has exactly one fully wrong negative with $y = 0$, so each row spans the whole score range. The test split has 60 rows per category, 240 in total, with $1{,}422$ negatives built by the same rules. Its images are never used for training.

Table 12: Rows and negatives per category, with the number of training negatives at each score y.

	Training							Test	
Category	Rows	Negatives	Per image	y=0.75	y=0.5	y=0.25	y=0	Rows	Negatives
Counting	1,008	5,018	4.98	1,624	1,994	392	1,008	60	297
Color	1,251	6,164	4.93	–	4,913	–	1,251	60	295
Attribute binding	1,455	10,114	6.95	2,910	2,839	2,910	1,455	60	414
Position	1,293	9,020	6.98	–	5,141	2,586	1,293	60	416
Total	5,007	30,316	6.05	4,534	14,887	5,888	5,007	240	1,422

B Unseen-Prompt Evaluation Set

Our critic is initialized from a large pretrained diffusion model with broad knowledge of visual concepts and their compositions, so LoRA fine-tuning mainly aligns the pretrained representations with compositional scoring rather than relearning or overwriting this knowledge. Although our corruption rules use controlled semantic edits, they require fine-grained comparisons involving objects, attributes, counts, spatial relations, and scenes. We therefore expect this compositional supervision to transfer to open-domain prompts and new perturbations.

To test this, we construct a new evaluation set from user-written prompts in Pick-a-Pic [19]. These prompts are free-form and include artistic styles, action verbs, surreal concepts, and world knowledge, none of which are present in our training prompts, for example *A fisherman fishing on a no man's lake in 1000 years*, *tshirt design with a zebra colorful rock and roll*, and *transylvania castle on hilltop and dusk bats and scifi*. For each prompt, an LLM extracts five semantic specifications (objects, actions, attributes, counts, and scenes) and generates negative captions by modifying one or more of them. We then generate images for the positive prompts with GPT-Image-1.5 and manually filter the resulting pairs for correctness, which leaves 200 positive prompts and 716 negatives. Table 13 shows an example.

Table 13: An example from the unseen-prompt set.

Label	Kind	Caption
1.00	Positive	Crazy monkey stealing food from gorilla
0.75	Swap verb	Crazy monkey sharing food with gorilla
0.50	Change count	Three crazy monkeys stealing food from gorilla
0.50	Swap object	Crazy monkey stealing food from a bear
0.00	Swap verb and objects	Calm parrot sharing food with a bear

Table 14 reports the ranking accuracy on this set. The FLUX critic achieves 91.6% and the SD3.5-M critic 88.7% at $t = 465$, showing that compositional accuracy generalizes to unseen prompts and new perturbations.

Table 14: Ranking accuracy ($\mathrm{acc}_{\mathrm{rank}}$, %) on the unseen-prompt set.

t	SD3.5-M critic	FLUX.1-dev critic
465	88.7	91.6
602	89.2	89.8
707	89.2	91.0

C Critic Training

Recipe. Both critics follow the same recipe on the same $5{,}007$ training rows, and only the backbone changes. The backbone stays frozen in bf16. We train LoRA adapters on its attention and feed-forward layers, the 16 query tokens, and the reward head with AdamW and a cosine learning-rate schedule that decays to 0.1 of its initial value. Each micro-batch holds two reference images. Each image is paired with its positive prompt and all of its negatives, and all of these image–prompt pairs,

about 14, pass through the backbone together in one forward pass, sharing the image's latent and noise. One optimizer step accumulates gradients over nine micro-batches. For each micro-batch, we draw one timestep uniformly from $[50, 950]$ and noise all of its latents to it. We set $\lambda_{\text{rank}} = 0.2$ with temperature $\tau = 0.15$, and $\lambda_{\text{FM}} = 1$. FLUX.1-dev is guidance-distilled, so its critic receives a constant guidance embedding of 3.5. It also uses a higher LoRA learning rate, 10^{-4}, which converged fast. Table 15 summarizes both settings.

Stopping and cost. We use early stopping on the held-out split, and stop training once ranking accuracy plateaus. The SD3.5-M critic reaches 97% after $5{,}301$ steps, about 8 hours on one H100, and the FLUX critic reaches 95% after 837 steps, about 5.2 hours. Although the FLUX critic takes longer per step, it converges in far fewer steps, which we attribute to its higher LoRA learning rate and guidance distillation.

Table 15: Critic training settings and cost on one H100.

	SD3.5-M critic	FLUX.1-dev critic
Backbone (frozen)	SD3.5-M, 2.5B	FLUX.1-dev, 12B
LoRA	rank 32, $\alpha = 64$	rank 32, $\alpha = 64$
Query tokens	16	16
Learning rate (head / LoRA)	10^{-4} / $3{\times}10^{-5}$	10^{-4} / 10^{-4}
Images per optimizer step	2×9	2×9
Training timesteps	$[50, 950]$	$[50, 950]$
Guidance embedding	–	3.5
Time per optimizer step	5.3 s	22.4 s
Optimizer steps	5,301	837
Training time	$\sim$8 h	$\sim$5.2 h

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the paper's main contributions: DeFlow-Critic, a dense latent reward model for flow-matching T2I alignment; its use for online GRPO training and inference-time trajectory selection; and empirical evaluation on GenEval2 and T2I-CompBench++, with transfer to SD3.5-Large and FLUX.1-dev. These claims are supported by the method and experiments in Sections 3, 4.2, and 4.4.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses limitations and broader impact in Section 5, including transfer to higher resolutions, video or image-to-video generation, and the empirical choice of the trusted scoring timestep.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not present theoretical results or formal proofs; it proposes an empirical method for dense latent reward modeling and online RL alignment.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper, regardless of whether the code and data are provided or not?

   Answer: [Yes]

   Justification: The paper provides the information needed to reproduce the main experiments, including reward-model dataset construction and split, model architecture, training objective, key hyperparameters, GRPO training setup, evaluation benchmarks, and compute cost in Sections 3 and 4.1 and Appendices A and C. Code, data, and model checkpoints will also be released on the project page.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: We will release the code, the reward-modeling dataset, and the model checkpoints on the project page (`https://zeeshank95.github.io/deflowcritic`); they are not yet public at the time of this version. The paper nevertheless provides the main implementation details needed to understand and reproduce the experiments.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details, e.g., data splits, hyperparameters, how they were chosen, type of optimizer, necessary to understand the results?

   Answer: [Yes]

   Justification: The paper reports the main training and evaluation details, including data splits, model architecture, loss terms, hyperparameters, optimizer, timestep ranges, GRPO settings, and benchmark protocols in Sections 3.1 and 4.1, with the scoring rules and critic training details in Appendices A and C. Additional low-level implementation details will be included with the released code.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We do not report error bars or confidence intervals because the main online RL experiments are computationally expensive. We instead report comparisons across multiple held-out benchmarks and ablations. Repeated runs and statistical uncertainty estimates are left for future work.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources, type of compute workers, memory, time of execution, needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Section 4.1 reports the compute resources and runtime for the main experiments, including H100 GPU usage, DeFlowCritic training time, and GRPO training time. Appendix C gives the training cost of each critic, and Table 4 the inference cost of best-of-K selection.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conforms to the NeurIPS Code of Ethics. The work uses generated and benchmark data for text-to-image alignment research, and we discuss potential misuse risks and responsible deployment considerations in the limitations and broader impact discussion.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The limitations and broader impact discussion in Section 5 covers both positive impacts, such as improved prompt faithfulness for creative and design applications, and negative impacts, including potential misuse for deceptive or harmful synthetic imagery. We also discuss responsible deployment measures such as content filtering, dataset auditing, and provenance mechanisms.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse, e.g., pre-trained language models, image generators, or scraped datasets?

    Answer: [No]

    Justification: We discuss responsible deployment considerations, including content filtering, dataset auditing, and provenance mechanisms, but the current paper does not specify concrete release safeguards for the model or dataset. We will include usage guidelines and release documentation with the public artifacts.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets, e.g., code, data, models, used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the existing models, benchmarks, and evaluation assets used in the paper, including SD3.5-Medium, SD3.5-Large, FLUX.1-dev, Flow-GRPO, GenEval,

GenEval2, T2I-CompBench++, Pick-a-Pic, LAION Aesthetic, HPSv2.1, HPSv3, Qwen-3-VL, and the DINA-LRM checkpoint. We will include license and terms-of-use information for these assets in the supplementary material or public release, and will ensure that any released code, data, and models respect the corresponding licenses.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The paper introduces a new reward-modeling dataset and the DeFlowCritic model. Dataset construction, filtering, splits, label semantics, model architecture, and training objectives are documented in the paper, and additional documentation will be provided with the released assets.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation, if any?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments or human-subject research. The reward-modeling dataset is generated and manually filtered by the authors, without collecting responses from external participants.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals, or an equivalent approval/review based on the requirements of your country or institution, were obtained?

    Answer: [N/A]

    Justification: The paper does not involve human-subject experiments or external study participants, so IRB approval or equivalent review is not applicable.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: We use GPT-Image-1.5 to generate reference images for the reward-modeling dataset, as described in Section 3.1. This usage is part of the data construction pipeline and is disclosed in the paper. We also use an LLM to write the negative captions of the unseen-prompt evaluation set (Appendix B).