RB-Modulation: Training-Free Personalization of Diffusion Models using Stochastic Optimal Control

Litu Rout[1,2,*] **Yujia Chen**[1] **Nataniel Ruiz**[1] **Abhishek Kumar**[3]

Constantine Caramanis[2] **Sanjay Shakkottai**[2] **Wen-Sheng Chu**[1]

[1] Google [2] UT Austin [3] Google DeepMind

{litu.rout,constantine,sanjay.shakkottai}@utexas.edu

{yujiachen,natanielruiz,abhishk,wschu}@google.com

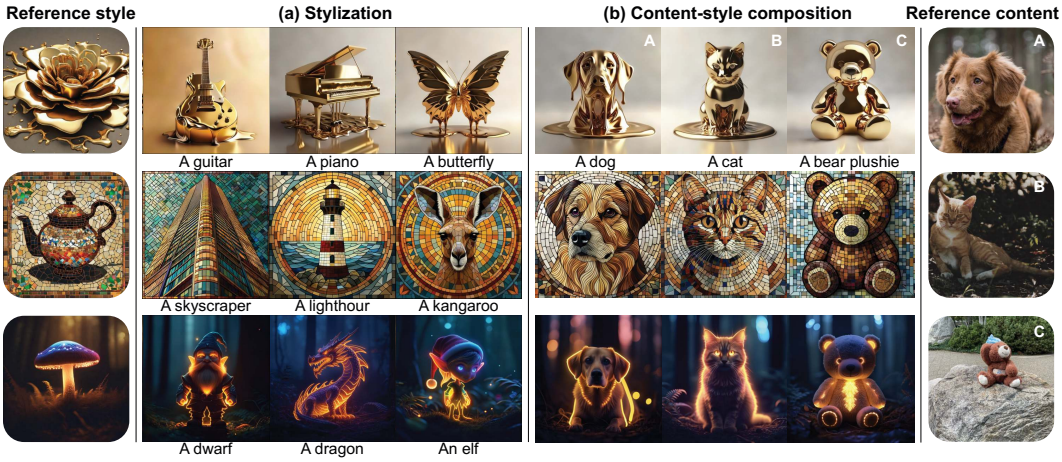

Figure 1: Given a single reference image (rounded rectangle), our method **RB-Modulation** offers a plug-and-play solution for (a) stylization, and (b) content-style composition with various prompts while maintaining sample diversity and prompt alignment. For instance, given a reference style image (*e.g.* "melting golden 3d rendering style") and content image (*e.g.* (A) "dog"), our method adheres to the desired prompts without leaking contents from the reference style image and without being restricted to the pose of the reference content image.

Abstract

We propose Reference-Based Modulation (RB-Modulation), a new plug-and-play solution for training-free personalization of diffusion models. Existing training-free approaches exhibit difficulties in (a) style extraction from reference images in the absence of additional style or content text descriptions, (b) unwanted content leakage from reference style images, and (c) effective composition of style and content. RB-Modulation is built on a novel stochastic optimal controller where a style descriptor encodes the desired attributes through a terminal cost. The resulting drift not only overcomes the difficulties above, but also ensures high fidelity to the reference style and adheres to the given text prompt. We also introduce a cross-attention-based feature aggregation scheme that allows RB-Modulation to decouple content and style from the reference image. With theoretical justification and empirical evidence, our framework demonstrates precise extraction and control of *content* and *style* in a training-free manner. Further, our method allows a seamless composition of content and style, which marks a departure from the dependency on external adapters or ControlNets.

[*] This work was done during an internship at Google.

Preprint. Under review.

1 Introduction

Text-to-image (T2I) generative models [1, 2, 3, 4] have excelled in crafting visually appealing images from text prompts. These T2I models are increasingly employed in creative endeavors such as visual arts [5], gaming [6], personalized image synthesis [7, 8, 9, 10], stylized rendering [11, 12, 13, 14], and image inversion or editing [15, 16, 17, 18]. Content creators often need precise control over both the *content* and the *style* of generated images to match their vision. While the content of an image can be conveyed through text, articulating an artist's unique style – characterized by distinct brushstrokes, color palette, material, and texture – is substantially more nuanced and complex. This has led to research on personalization through visual prompting [11, 12, 13].

Recent studies have focused on finetuning pre-trained T2I models to learn style from a set of reference images [19, 7, 11, 9]. This involves optimizing the model's text embeddings, model weights, or both, using the denoising diffusion loss. However, these methods demand substantial computational resources for training or finetuning large-scale foundation models, thus making them expensive to adapt to new, unseen styles. Furthermore, these methods often depend on human-curated images of the same style, which is less practical and can compromise quality when only a single reference image is available.

In training-free **stylization**, recent methods [12, 13, 14] manipulate keys and values within the attention layers using just one reference style image. These methods face challenges in both extracting the style from the reference style image and accurately transferring the style to a target content image. For instance, during the DDIM inversion step [20] utilized by StyleAligned [12], fine-grained details tend to be compromised. To mitigate this issue, InstantStyle [13] incorporates features from the reference style image into specific layers of a previously trained IP-Adapter [21]. However, identifying the exact layer for feature injection in a model is complex and not universally applicable across models. Also, feature injection can cause content leakage from the style image into the generated content. Moving on to content-style **composition**, InstantStyle [13] employs a ControlNet [22] (an additionally trained network) to preserve image layout, which inadvertently limits its diversity.

We introduce Reference-Based Modulation (RB-Modulation), a novel approach for content and style personalization that eliminates the need for training or finetuning diffusion models (*e.g.* ControlNet [22] or adapters [21, 9]). Our work reveals that the reverse dynamics in diffusion models can be formulated as stochastic optimal control with a terminal cost. By incorporating style features into the controller's terminal cost, we modulate the drift field in diffusion models' reverse dynamics, enabling training-free personalization. Unlike conventional attention processors that often leak content from the reference style image, we propose to enhance the image fidelity via an Attention Feature Aggregation (AFA) module that decouples content from reference style image. We demonstrate the effectiveness of our method in stylization [12, 13, 14] and style+content composition, as illustrated in Fig. 1(a) and (b), respectively. Our experiments show that RB-Modulation outperforms current SoTA methods [12, 13] in terms of human preference and prompt-alignment metrics.

Our contributions are summarized as follows:

- We present reference-based modulation (RB-Modulation), a novel stochastic optimal control framework that enables training-free, personalized style and content control, with a new Attention Feature Aggregation (AFA) module to maintain high fidelity to the reference image while adhering to the given prompt (§4).
- We provide theoretical justifications connecting optimal control and reverse diffusion dynamics. We leverage this connection to incorporate desired attributes (*e.g.*, style) in our controller's terminal cost and personalize T2I models in a training-free manner (§5).
- We perform extensive experiments covering stylization and content-style composition, demonstrating superior performance over SoTA methods in human preference metrics (§6).

2 Related Work

Personalization of T2I models: T2I generative models [3, 23, 24] are pushing the boundaries in generating aesthetically pleasing images by precisely interpreting text prompts. Their ability to follow a desired text has unlocked new avenues in *personalized* content creation, including text-guided image editing [17, 18], solving inverse problems [16, 17], concept-driven generation [7, 25, 26, 27], personalized outpainting [28], identity-preservation [29, 8, 30], and stylized synthesis [11, 13,

12, 10]. To tailor T2I models for a specific style (*e.g.*, painting) or content (*e.g.*, object), existing methods follow one of two recipes: (1) full finetuning (FT) [7, 31] or parameter efficient finetuning (PEFT) [26, 21, 9, 11, 10] and (2) training(finetuning)-free [12, 13, 14], which we discuss below.

Finetuning T2I models for personalization: FT [7, 31] and PEFT [26, 21, 9, 11, 10] methods, such as IP-Adapter [21], LoRA [9], ZipLoRA [10], and StyleDrop [11], excel at capturing style or object details when the underlying T2I model is finetuned on a few (typically 4) reference images for few thousand iterations. With the increasing size of T2I models, PEFT is preferred over FT due to fewer trainable parameters. However, the challenge of curating a set of reference images and resource-intensive finetuning for every style or content remains largely unexplored.

Training(finetuning)-free T2I models for personalization: The need to improve T2I model finetuning has sparked interests in training-free personalization methods. In **StyleAligned** [12], a reference style image and a text prompt describing the style are used to extract style features via DDIM inversion [20]. Target queries and keys are then normalized using adaptive instance normalization [32] based on reference counterparts. Finally, reference image keys and values are merged with DDIM-inverted latents in self-attention layers, which tends to leak content information from the reference style image (Figure 3). Moreover, the need for textual description in the DDIM inversion step can degrade its performance. **Swapping Self-Attention (SSA)** [14] addresses these limitations by replacing the target keys and values in self-attention layers with those from a reference style image. Yet, it still relies on DDIM inversion to cache keys and values of the reference style, which tends to compromise fine-grained details [13]. Both StyleAligned [12] and SSA [14] require two reverse processes to share their attention layer features and thus demand significant memory. Recently, **InstantStyle** [13] injects reference style features into specific cross-attention layers of IP-Adapter [21], addressing two key limitations: DDIM inversion and memory-intensive reverse processes. However, pinpointing the exact layers for feature injection is complex, and they may not generalize to other models. In addition, when composing style and content, InstantStyle [13] relies on ControlNet [22], which can limit the diversity of generated images to fixed layouts and deviate from the prompt.

Optimal Control: Stochastic optimal control finds wide applications in diverse fields such as molecular dynamics [33], economics [34], non-convex optimization [35], robotics [36], and mean-field games [37] Despite its extensive use, it has been less explored in personalizing diffusion models. In this paper, we introduce a novel framework leveraging the main concepts from optimal control to achieve training-free personalization. A key aspect of optimal control is designing a controller to guide a stochastic process towards a desired terminal condition [34]. This aligns with our goal of training-free personalization, as we target a specific style or content at the end of the reverse diffusion process, which can be incorporated in the controller's terminal condition.

RB-Modulation overcomes several challenges encountered by SoTA methods [12, 14, 13]. Since RB-Modulation does not require DDIM inversion, it retains fine-grained details unlike StyleAligned [12]. Using a stochastic controller to refine the trajectory of a single reverse process, it overcomes the limitation of coupled reverse processes [12]. By incorporating a style descriptor in our controller's terminal cost, it eliminates the dependency on Adapters [21, 9] or ControlNets [22] by InstantStyle [13].

3 Preliminaries

Diffusion models consist of two stochastic processes: (a) *noising process*, modeled by a Stochastic Differential Equation (SDE) known as forward-SDE: $\mathrm{d}X_t = f(X_t, t)\,\mathrm{d}t + g(X_t, t)\,\mathrm{d}W_t, X_0 \sim p_0$, and (b) *denoising process*, modeled by the time-reversal of forward-SDE under mild regularity conditions [38], also known as reverse-SDE:

$$\mathrm{d}X_t = \left[f(X_t, t) - g^2(X_t, t)\nabla \log p(X_t, t)\right]\,\mathrm{d}t + g(X_t, t)\,\mathrm{d}W_t, \qquad X_1 \sim \mathcal{N}\left(0, \mathrm{I}_d\right). \quad (1)$$

Here, $W = (W_t)_{t\geq 0}$ is standard Brownian motion in a filtered probability space, $(\Omega, \mathcal{F}, (\mathcal{F}_t)_{t\geq 0}, \mathcal{P})$, $p(\cdot, t)$ denotes the marginal density of p at time t, and $\nabla \log p_t(\cdot)$ the corresponding score function. $f(X_t, t)$ and $g(X_t, t)$ are called drift and volatility, respectively. A popular choice of $f(X_t, t) = -X_t$ and $g(X_t, t) = \sqrt{2}$ corresponds to the well-known forward Ornstein-Uhlenbeck (OU) process.

For T2I generation, the reverse-SDE (1) is simulated using a neural network $s\left(\mathbf{x}_t, t; \theta\right)$ [39, 40] to approximate $\nabla_{\mathbf{x}} \log p(\mathbf{x}_t, t)$. Importantly, to accelerate the sampling process in practice [20, 41, 42], the reverse-SDE (1) shares the same path measure with a probability flow ODE: $\mathrm{d}X_t = \left[f(X_t, t) - \frac{1}{2}g^2(X_t, t)\nabla \log p(X_t, t)\right]\,\mathrm{d}t$, where $X_1 \sim \mathcal{N}\left(0, \mathrm{I}_d\right)$.

Personalized diffusion models either fully finetune θ of $s(\mathbf{x}_t, t; \theta)$ [7, 31], or train a parameter-efficient adapter $\Delta\theta$ for $s(\mathbf{x}_t, t; \theta + \Delta\theta)$ on reference style images [9, 11, 10]. Our method does not finetune θ or train $\Delta\theta$. Instead, we derive a new drift field through a stochastic optimal controller that *modulates the drift* of the standard reverse-SDE (1).

4 Method

Personalization using optimal control: Normalize time t by the total number of diffusion steps T such that $0 \leq t \leq 1$. Let us denote by $u : \mathbb{R}^d \times [0,1] \to \mathbb{R}^d$ a controller from the admissible set of controls $\mathcal{U} \subseteq \mathbb{R}^d$, $X_t^u \in \mathbb{R}^d$ a state variable, $\ell : \mathbb{R}^d \times \mathbb{R}^d \times [0,1] \to \mathbb{R}$ the transient cost, and $h : \mathbb{R}^d \to \mathbb{R}$ the terminal cost of the reverse process $(X_t^u)_{t=1}^0$. We show in §5 that training-free personalization can be formulated as a control problem where the drift of the standard reverse-SDE (1) is modified via RB-modulation:

$$\min_{u \in \mathcal{U}} \mathbb{E}[\int_1^0 \ell(X_t^u, u(X_t^u, t), t)\,\mathrm{d}t + \gamma h(X_0^u)], \quad \text{where} \tag{2}$$
$$\mathrm{d}X_t^u = \left[f(X_t^u, t) - g^2(X_t^u, t)\nabla \log p(X_t^u, t) + u(X_t^u, t)\right]\mathrm{d}t + g(X_t^u, t)\mathrm{d}W_t, X_1^u \sim \mathcal{N}(0, \mathrm{I}_d).$$

Importantly, the terminal cost $h(\cdot)$, weighted by γ, captures the discrepancy in feature space between the styles of the reference image and the generated image. The resulting controller $u(\cdot, t)$ modulates the drift over time to satisfy this terminal cost. We derive the solution to this optimal control problem through the Hamilton-Jacobi-Bellman (HJB) equation [34]; refer to Appendix A for details. Our proposed RB-Modulation **Algorithm 1** has two key components: (a) stochastic optimal controller and (b) attention feature aggregation. Below, we discuss each in turn.

(a) Stochastic Optimal Controller (SOC): We show that the reverse dynamics in diffusion models can be framed as a stochastic optimal control problem with a quadratic terminal cost (theoretical analysis in §5). For personalization using a reference style image $X_0^f = \mathbf{z}_0$, we use a Consistent Style Descriptor (CSD) [43] to extract style features $\Psi(X_0^f)$. Since the score functions $s(\mathbf{x}_t, t; \theta) \approx \nabla \log p(X_t, t)$ are available from pre-trained diffusion models [23, 24], our goal is to add a correction term $u(\cdot, t)$ to modulate the reverse-SDE and minimize the overall cost (2). We approximate X_0^u with its conditional expectation using Tweedie's formula [16, 17]. Finally, we incorporate the style features into our controller's terminal cost as: $h(X_0^u) = \|\Psi(X_0^f) - \Psi(\mathbb{E}[X_0^u | X_t^u])\|_2^2$.

Our theoretical results (§5) suggest that the optimal controller can be obtained by solving the HJB equation and letting $\gamma \to \infty$. In practice, this translates to dropping the transient cost $\ell(X_t^u, u(X_t^u, t), t)$ and solving (2) with only the terminal constraint, *i.e.*,

$$\min_{u \in \mathcal{U}} \|\Psi(X_0^f) - \Psi(\mathbb{E}[X_0^u | X_t^u])\|_2^2. \tag{3}$$

Thus, we solve (3) to find the optimal control u and use this controller in the reverse dynamics (2) to update the current state from X_t^u to $X_{t-\Delta t}^u$ (recall that time flows backwards in the reverse-SDE (1)). Our implementation of (3) is given in **Algorithm 1**, which follows from our theoretical insights.

Implementation challenge: For smaller generative models [3], we can directly solve our control problem (3). However, for larger models [23, 24], optimizing our control objective (3) requires back propagation through the score network $s(\mathbf{x}_t, t; \theta)$ with tentatively billions of parameters. This significantly increases time and memory complexity [16, 17].

We propose a proximal gradient descent approach to address this challenge. Recall that the key ingredient of our **Algorithm 1** is to find the previous state $X_{t-\Delta t}$ by modulating the current state X_t based on an optimal controller u^*. The optimal controller u^* is obtained by minimizing the discrepancy in style between $\bar{X}_0^u := \mathbb{E}[X_0^u | X_t^u = \mathbf{x}_t]$, obtained using our controlled reverse-SDE (3), and the reference style image $\mathbf{z}_0$. Motivated by this interpretation, an alternate **Algorithm 2** (see Appendix B.2) avoids back propagation through $s(\mathbf{x}_t, t; \theta)$ by introducing a dummy variable $\mathbf{x}_0$, which serves as a proxy for $\bar{X}_0^u$ in the terminal cost. Instead of forcing $\mathbf{x}_0$ to be decided by the dynamics of the reverse-SDE as in **Algorithm 1**, we allow it to be only approximately faithful to the dynamics. This is implemented by adding a proximal penalty, *i.e.* $\mathbf{x}_0^* = \arg\min_{\mathbf{x}_0 \in \mathbb{R}^d} \|\Psi(X_0^f) - \Psi(\mathbf{x}_0)\|_2^2 + \lambda \|\mathbf{x}_0 - \mathbb{E}[X_0^u | X_t^u]\|_2^2$, where the hyper-parameter λ controls the faithfulness of the reverse dynamics. This penalty assumes that with a small step-size in the reverse-SDE dynamics (3), $\mathbf{x}_0^*$

Algorithm 1: Reference-Based Modulation

Input: Diffusion steps T, reference prompt $\mathbf{p}$, reference image $\mathbf{z}_0$, style descriptor $\Psi(\cdot)$, score network $s(\cdot,\cdot,\cdot;\theta)$
Tunable parameter: Stepsize η, optimization steps M
Output: Personalized latent X_0^u

Initialize $\mathbf{x}_T \leftarrow \mathcal{N}(0, \mathrm{I}_d)$
for $t = T$ **to** 1 **do**
  Initialize controller $u = 0$
  for $m = 1$ **to** M **do**
    $\hat{\mathbf{x}}_t = \mathbf{x}_t + u$ ▷ controlled state
    $\bar{X}_0^u = \frac{\hat{\mathbf{x}}_t}{\sqrt{\bar{\alpha}_t}} + \frac{(1-\bar{\alpha}_t)}{\sqrt{\bar{\alpha}_t}} s(\hat{\mathbf{x}}_t, t, \mathbf{p}; \theta)$
    $h(\bar{X}_0^u) = \|\Psi(\mathbf{z}_0) - \Psi(\bar{X}_0^u)\|_2^2$ using Eq. (3)
    $u = u - \eta \nabla_u h(\bar{X}_0^u)$ ▷ update controller
  end
  $\mathbf{x}_t^* = \mathbf{x}_t + u$ ▷ optimally controlled state
  $\bar{X}_0^u = \frac{\mathbf{x}_t^*}{\sqrt{\bar{\alpha}_t}} + \frac{(1-\bar{\alpha}_t)}{\sqrt{\bar{\alpha}_t}} s(\mathbf{x}_t^*, t, \mathbf{p}; \theta)$ ▷ terminal state
  $\mathbf{x}_{t-1} \leftarrow \mathrm{DDIM}(\bar{X}_0^u, \mathbf{x}_t^*)$ ▷ one step reverse-SDE [20]
end
return X_0^u

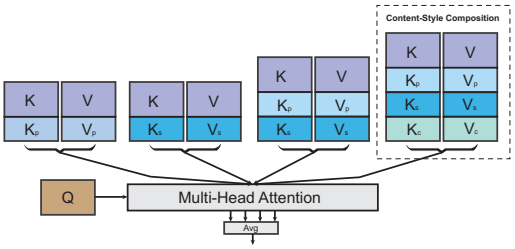

Figure 2: **Attention Feature Aggregation (AFA):** Within the cross-attention layers, the keys and values from the previous layers (K,V), text embedding (K_p,V_p), reference style image (K_s,V_s) and reference content image (K_c,V_c) are concatenated and processed separately to disentangle the information, which is followed by an averaging layer for the output. K_c,V_c and only used for content-style composition.

and $\mathbb{E}[X_0^u | X_t^u = \mathbf{x}_t]$ will be close. Therefore, **Algorithm 2** enables personalization of large-scale foundation models without significantly increasing time and memory complexity.

(b) Attention Feature Aggregation (AFA): Transformer-based diffusion models [3, 23, 24] consist of self-attention and cross-attention layers operating on latent embedding $\mathbf{x}_t \in \mathbb{R}^{d \times n_h}$. Within the attention module $\mathrm{Attention}(Q, K, V)$, $\mathbf{x}_t$ is projected into queries $Q \in \mathbb{R}^{d \times n_q}$, keys $K \in \mathbb{R}^{d \times n_q}$, and values $V \in \mathbb{R}^{d \times n_h}$ using linear projections. Through Q, K, and V, attention layers capture global context and improve long-range dependencies within $\mathbf{x}_t$.

To accommodate a reference image (*e.g.*, style or content) while retaining prompt-alignment, we propose an Attention Feature Aggregation (AFA) module, as illustrated in Figure 2. For a given prompt $\mathbf{p}$, a reference style image I_s, and a reference content image I_c, we first extract the embeddings using CLIP [44] text and image encoder, respectively. Then, we project these embeddings into keys and values using linear projection layers. We denote by K_p and V_p the keys and values from $\mathbf{p}$, K_s and V_s from I_s, K_c and V_c from I_c (used only in content-style composition). The query Q is obtained from a linear projection of $\mathbf{x}_t$, and remains the same in the AFA module. By processing the keys and values separately, we disentangle their relative importance with respect to the state variable. This ensures that the attention maps from text are not contaminated with attention maps from style. To make the text consistent with the style, we also compose the keys and values of both text and style in our attention processor. The final output of our AFA module is given by

$$AFA = \mathrm{Avg}\left(A_{text}, A_{style}, A_{text+style}\right), A_{text} = \mathrm{Attention}(Q, [K; K_p], [V; V_p]),$$
$$A_{style} = \mathrm{Attention}(Q, [K; K_s], [V; V_s]), A_{text+style} = \mathrm{Attention}(Q, [K; K_p; K_s], [V; V_p; V_s]),$$

where $[K; K_p] \in \mathbb{R}^{2d \times n_q}$ indicates concatenation of K with K_p along the number of tokens dimension. For style-content composition, we process the content image I_c in the same way as the reference style image I_s, and obtain another set of attention outputs:

$$AFA = \mathrm{Avg}\left(A_{text}, A_{style}, A_{content}, A_{content+style}\right),$$
$$A_{content} = \mathrm{Attention}(Q, [K; K_c], [V; V_c]), A_{content+style} = \mathrm{Attention}(Q, [K; K_s; K_c], [V; V_s; V_c]).$$

Importantly, the AFA module is computationally tractable as it only requires the computation of a multi-head attention, which is widely used in practice [23].

5 Theoretical Justifications

Problem setup: We outline an approach to derive the optimal controller for a special case of our control problem (2). We substitute $t \leftarrow 1 - t$ to account for the time reversal in the reverse-SDE (1). Here, $X_0^u \sim \mathcal{N}(0, \mathrm{I}_d)$ and $X_1^u \sim p_{data}$. We consider the dynamic without the Brownian motion: $\mathrm{d}X_t^u = v(X_t^u, u, t)\mathrm{d}t, \quad X_{t_0}^u = \mathbf{x}_0$, where $0 \le t_0 \le t \le t_N \le 1$ and $v : \mathbb{R}^d \times \mathbb{R}^d \times [t_0, t_N] \to \mathbb{R}^d$ denotes the drift field. The optimal controller u^* can be derived by solving the Hamilton-Jacobi-Bellman (HJB) equation [34, 45], see Appendix A for details.

Incorporating optimal control in diffusion: Following recent works [46, 47], we consider a dynamical system whose drift field minimizes a transient trajectory cost and a terminal cost (weighted by γ) to ensure "closeness" to reference content x_1 (Appendix A.1). **Proposition A.2** [47] outlines the optimal control in the limiting setting where $\gamma \to \infty$. Furthermore, suppose we replace x_1 with its conditional expectation (discussed in Remark A.3), *the resulting dynamic is the standard reverse-SDE for the Orstein-Uhlenbeck (OU) diffusion process for a particular noise schedule.* This connection between classic linear quadratic control and the standard reverse-SDE allows us to study other diffusion problems (*e.g.*, personalization) through the lens of stochastic optimal control. For instance, we derive the optimal controller given reference *style features* y_1 at the terminal time.

Proposition 5.1. *Suppose $A \in \mathbb{R}^{k\times d}$ be a linear style extractor that operates on the terminal state $X_1^u \in \mathbb{R}^d$. Given reference style features y_1, consider the control problem:*

$$\min_{u\in\mathcal{U}} \int_{t_0}^{1} \frac{1}{2} \left\|u(X_t^u, t)\right\|^2 dt + \frac{\gamma}{2} \left\|AX_1^u - y_1\right\|_2^2, \text{ where } \mathrm{d}X_t^u = u(X_t^u, t)\,\mathrm{d}t,\ X_{t_0}^u = x_0.$$

Then, in the limit when $\gamma \to \infty$, the optimal controller $u^* = \frac{\left(A^TA\right)^{-1}A^T(y_1 - A\mathbf{x}_t)}{1-t}$, which yields the following controlled dynamic: $\mathrm{d}X_t^u = \frac{\left(A^TA\right)^{-1}A^T(y_1 - A\mathbf{x}_t)}{1-t}\mathrm{d}t$.

Implication. The optimal controller depends on the reference *style features* y_1 at the terminal time, instead of the image content encoded in x_1. To simulate the controlled dynamic in practice, we use CSD [43] as a style feature extractor and replace y_1 with the style features extracted from the expected terminal state $\mathbb{E}[X_1^u|X_t^u]$, as discussed in **Appendix A.2**.

Drift modulation through optimal controller: We then study a control problem where the velocity field is a linear combination of the state and the control variable. This problem is interesting to study because the reverse-SDE dynamic of the standard OU process has a drift field of the form: $v\left(X_t, t\right) = -X_t - 2\nabla \log p(X_t, t)$. For a Gaussian prior $X_0 \sim \mathcal{N}\left(0, \mathrm{I}\right)$, the law of the OU process satisfies $\nabla \log p\left(X_t, t\right) = -X_t$, and the corresponding drift field becomes $v\left(X_t, t\right) = X_t$. Our goal is to modulate this drift field using a controller $u\left(X_t^u, t\right)$. The result below provides the structure of the optimal control (again in the setting where the terminal objective is known; see Appendix A1).

Proposition 5.2. *Suppose $A \in \mathbb{R}^{k\times d}$ be a linear style extractor that operates on the terminal state $X_1^u \in \mathbb{R}^d$. Let $\mathbf{p}_t$ denote $\nabla_{\mathbf{x}} V^*(\mathbf{x}, t)$ in HJB equation (A.1). Given reference style features y_1, consider the control problem:*

$$\min_{u\in\mathcal{U}} \int_{t_0}^{1} \frac{1}{2} \left\|u(X_t^u, t)\right\|^2 dt + \frac{\gamma}{2} \left\|AX_1^u - y_1\right\|_2^2, \text{ where } \mathrm{d}X_t^u = \left[X_t^u + u(X_t^u, t)\right]\mathrm{d}t,\ X_{t_0}^u = x_0,$$

Then, the optimal controller becomes $u^*(t) = -\mathbf{p}_t$, where the instantaneous state $X_t^u = \mathbf{x}_t$ and $\mathbf{p}_t$ satisfy the following coupled transitions:

$$\begin{bmatrix}\mathbf{x}_t \\ \mathbf{p}_t\end{bmatrix} = \begin{bmatrix} x_0 e^t - \frac{\gamma}{2} A^T\left(A\mathbf{x}_1 - y_1\right) e^{1+t} + \frac{\gamma}{2} A^T\left(A\mathbf{x}_1 - y_1\right) e^{1-t} \\ \gamma A^T\left(A\mathbf{x}_1 - y_1\right) e^{1-t} \end{bmatrix}.$$

Summary. We build on the connection between optimal control and reverse diffusion (see Appendices A.1-A.3 for details). The general strategy is to derive the optimal controller with known terminal state, and then replace the terminal state in the controller with its estimate using Tweedie's formula. For stylized models and Gaussian prior, the controllers have an explicit form. However in practice, the data distribution may not be Gaussian, and thus, we do not aim for a closed-form expression to modulate the drift. This line of analysis, however, points to our method RB-Modulation. As discussed in §4, we incorporate a style descriptor in our controller's terminal cost and numerically evaluate the resulting drift at each reverse time step either through back propagating through the score network (**Algorithm 1**), or an approximation based on proximal gradient updates (**Algorithm 2**).

6 Experiments

Metrics: Evaluating stylized synthesis is challenging due to the subjective nature of style, making simple metrics inadequate. We follow a two step approach: first using metrics from prior works and then conducting human evaluation. To evaluate prompt-image alignment, we use CLIP-T

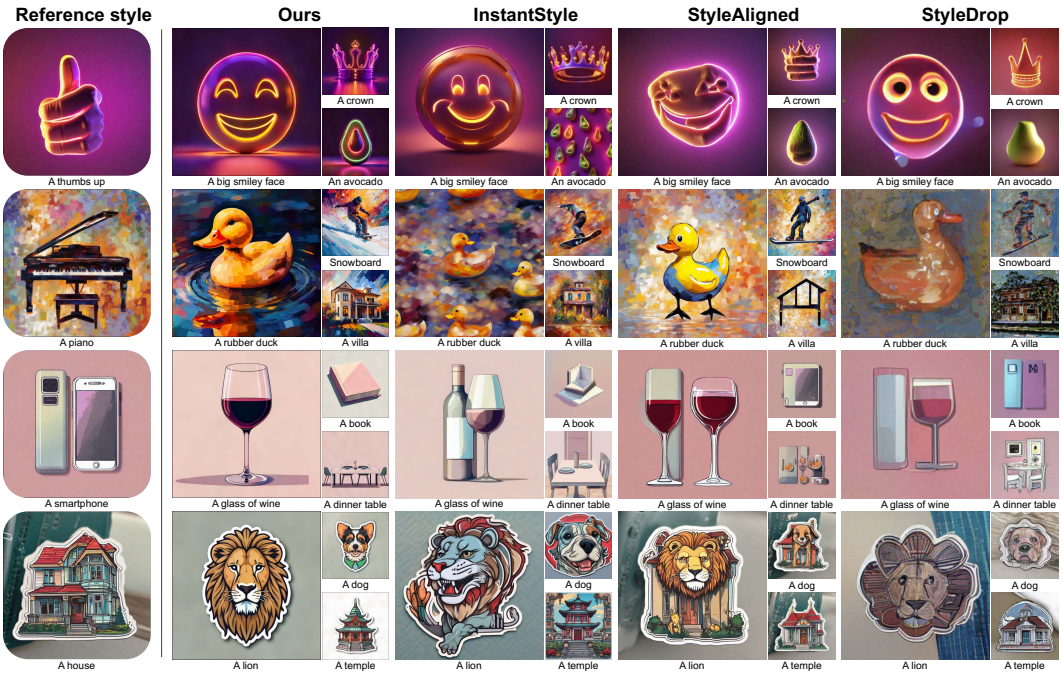

Figure 3: **Qualitative results for stylization:** A comparison with state-of-the-art methods (InstantStyle [13], StyleAligned [12], StyleDrop [11]) highlights our advantages in preventing information leakage from the reference style and adhering more closely to desired prompts.

score [12, 11, 13] and ImageReward [5], which also consider human aesthetics, distortions, and object completeness. When a style description is provided, CLIP-T and ImageReward also capture style alignment. We assess style similarity using DINO [48] and content similarity using CLIP-I [44] as in prior work [12, 7, 11], and highlight their limitations in disentangling style and content performance in evaluation. Given the importance of human evaluation in T2I personalization [12, 11, 7, 10, 14], we also conduct a user study though Amazon Mechanical Turk to measure both style and text alignment.

Datasets and baselines: We use style images from StyleAligned benchmark [12] for stylization and content images from DreamBooth [7] for content-style composition. We base RB-Modulation on the recently released StableCascade [24]. We compare our approach with three training-free methods: InstantStyle [13] (state-of-the-art), IP-Adapter [21], and StyleAligned [12]. For completeness, we also compare with training-based methods StyleDrop [11] and ZipLoRA [10].

Implementation details: All experiments run on a single A100 NVIDIA GPU. We use the same hyper-parameters for our method across tasks, and default settings for alternative methods as per their original papers. Details are provided in Appendix B.1.

6.1 Image Stylization

Qualitative analysis: This section describes image stylization experiments using a text prompt and a reference style image. Figure 3 compares our method with SoTA **training-free** InstantStyle [13] and StyleAligned [12], and **training-based** StyleDrop [11]. Except for StyleDrop, which requires $\sim$5 minutes of training per style, all methods, including ours, are training-free and complete inference in $<$1 minute. While all methods produce reasonable outputs, alternative methods encounter issues with information leakage. For instance, in the third row of Figure 3, StyleAligned and StyleDrop generate a wine bottle and book resembling the smartphone in the reference style image. In the last row, StyleAligned leaks the house and the background of the reference image; InstantStyle exhibits color leakage from the house, resulting in similar-colored images. Our method accurately adheres to the prompt in the desired style. As illustrated in the second and the third row, our method generates only one glass of wine and a high-fidelity rubber duck, compared to baselines where extra items appear (wine bottles styled like the left smartphone) or incorrect styles (cartoon-style rubber duck).

User study: To validate the qualitative analysis, we conduct a user study on Amazon Mechanical Turk with 155 participants using 100 styles from the StyleAligned dataset [12], collecting a total

Human Preference (%)	**Ours *vs*. InstantStyle** [13]			**Ours *vs*. StyleAligned** [12]			**Ours *vs*. IP-Adapter** [21]		
	OQ ↑	SA ↑	PA ↑	OQ ↑	SA ↑	PA ↑	OQ ↑	SA ↑	PA ↑
Alternative	39.8	38.5	39.5	24.4	27.8	29.4	8.1	20.1	8.3
Tie	9.3	6.4	7.3	8.8	7.1	5.8	6.9	4.8	4.5
RB-Modulation (ours)	**51.0**	**55.1**	**53.3**	**66.9**	**65.1**	**64.9**	**85.0**	**75.1**	**87.2**

Table 1: **User study:** We report the % of human preference on ours *vs*. alternatives for overall quality (OQ), style alignment (SA), and prompt alignment (PA), including ties where users couldn't decide. Our method consistently outperforms alternatives, achieving higher scores in all metrics.

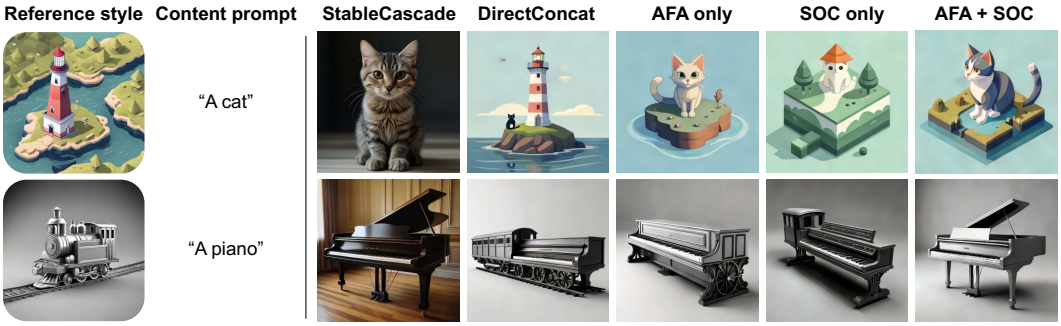

Figure 4: **Ablation study:** Our method builds on any transformer-based diffusion model. In this case, we use StableCascade [24] as the foundation, and sequentially add each module to show their effectiveness. DirectConcat involves concatenating reference image embeddings with prompt embeddings. Style descriptions are excluded in this ablation study.

of 7,200 answers (8 responses for each question). Each user answers 3 questions comparing our method with an alternative method regarding (1) overall quality, (2) style alignment, and (3) prompt alignment (details in the Appendix B.7). Table 1 summarizes the percentage of human preferences for our method, the alternative method, or a tie. Our method consistently outperforms the alternatives, including the most competitive method InstantStyle [13] in style alignment. The preference rates over all three metrics highlight the effectiveness of our method RB-Modulation.

Quantitative analysis: Table 2 evaluates 300 prompts and 100 styles on the StyleAligned dataset [12] using three metrics, with and without style descriptions in the prompts. Our method outperforms others notably in the ImageReward metric, closely matching human aesthetics assessment from the user study in Table 1. In addition, the CLIP-T score indicates our effective alignment between generated images and text prompts. While IP-Adapter and StyleAligned have higher DINO scores, their lower rating in ImageReward, CLIP-T and user preference expose information leakage from the reference style images. Nevertheless, our DINO score remains competitive with the leading method InstantStyle. Notably, all metrics show improvement with style descriptions, particularly in ImageReward, where leveraging style descriptions enhances prompt alignment. Our method achieves high ImageReward and CLIP-T score even without style descriptions, suggesting robustness in prompt alignment without explicit style information in the prompt.

Ablation Study: Figure 4 shows an ablation study of the AFA and SOC modules adding new capabilities to StableCascade [24]. We include a baseline, "DirectConcat", which concatenates reference style embeddings with text embeddings in the cross-attention modules. DirectConcat mixes both embeddings, making it less effective in disentangling style from prompts (*e.g.*, cat *vs*. lighthouse). While AFA or SOC alone mitigates this by modulating the reverse drift and attention modules (§4), each has drawbacks. AFA alone fails to capture the cat's style accurately, and SOC alone misplaces elements, like "a lighthouse hat on the cat" and "a railroad trunk on a piano". We observe consistent improvements with each module, with the best results when combined. Quantitative analysis is omitted due to the lack of suitable metrics for information leakage, as detailed in Appendix B.5.

6.2 Content-Style Composition

Qualitative analysis: Content-style composition aims to preserve the essence of both content and style depicted in the reference images, while ensuring the resulting image aligns with a given text prompt. Figure 5 compares our method against **training-free** InstantStyle [13], IP-Adapter [21],

With style description?	ImageReward ↑ No	ImageReward ↑ Yes	CLIP-T score ↑ No	CLIP-T score ↑ Yes	DINO score No	DINO score Yes
IP-Adapter [21]	-1.99	-1.51	0.21	0.26	0.89	0.89
StyleAligned [12]	-0.68	0.01	0.26	0.31	0.80	0.85
InstantStyle [13]	0.09	0.72	0.29	0.33	0.68	0.72
RB-Modulation (ours)	0.91	1.18	0.30	0.34	0.68	0.73

Table 2: **Quantitative results for stylization:** We compare alternative methods on three metrics: ImageReward [5] and CLIP-T [44] for prompt alignment, DINO [48] for style alignment. Note that DINO score does not capture information leakage, so higher scores are not necessarily better (§B.5).

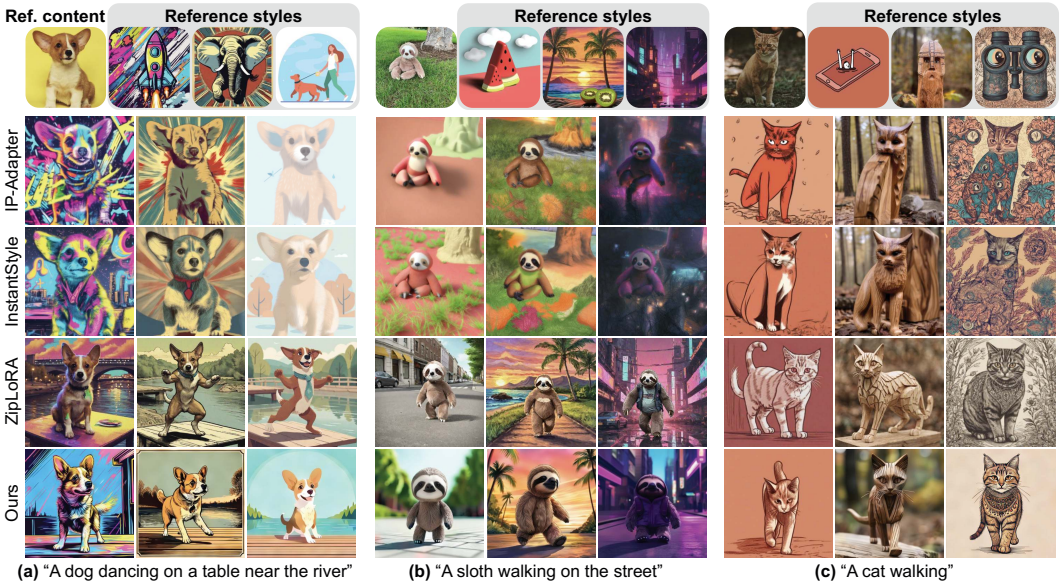

Figure 5: **Qualitative results for content-style composition:** Our method shows better prompt alignment and greater diversity than training-free methods IP-Adapter [21] and InstantStyle [13], and have competitive performance with training-based ZipLoRA [10] .

	ImageReward ↑	CLIP-T score ↑	DINO score	CLIP-I score
IP-Adapter [21]	-0.78	0.22	0.73	0.68
InstantStyle [13]	-0.54	0.21	0.71	0.71
RB-Modulation (ours)	0.74	0.26	0.74	0.71

Table 3: **Quantitative results for composition:** In addition to the metrics used for stylization, we use CLIP-T score [44] to evaluate content alignment with the reference content. Similar to DINO, CLIP-I could inflate test score [11, 10] by capturing content leakage, but not necessarily preferred by users; higher DINO and CLIP-I scores do not mean better human preference.

and **training-based** ZipLoRA [10]. Notably, the training-free InstantStyle and IP-Adapter rely on ControlNet [22], which often constrains their ability to accurately follow prompts for changing the pose of the generated content, such as illustrating "dancing" in Figure 5(b), or "walking" in (c). In contrast, our method avoids the need for ControlNet or adapters, and can effectively capture the distinctive attributes of both style and content images while adhering to the prompt to generate diverse images. In Figure 5(a), our method accurately captures elements like "table" and "river" that are overlooked in InstantStyle and IP-Adapter. In addition, our method mitigates information leakage, as evidenced in Figure 5(b), where the trunk of the tree behind the sloth is erroneously captured by InstantStyle and IP-Adapter but not by ours. Compared to ZipLoRA [10] that requires training of 12 LoRAs [9] and additional merge layers for each composition, our method requires no training at all while yielding competitive or better results. For instance, our method effectively captures the 2D cartoon and 3D rendering styles as illustrated in Figures 5(a) and (b).

Quantitative analysis: Table 3 shows quantitative evaluation using 50 styles from StyleAligned dataset [12] and 5 contents from DreamBooth dataset [7]. Unlike prior works [12, 11, 10, 7, 14] reporting either DINO and CLIP-I scores, we present both metrics and demonstrate comparable

performance across them. Additionally, we obtain notably higher ImageReward score, which aligns closely with human aesthetics assessment as evidenced in §6.1 and [5]. Consequently, we omitted a user study in this section. For more details, please refer to Appendix B.1.

7 Conclusion

We introduced Reference-Based modulation (RB-Modulation), a training-free method for personalizing transformer-based diffusion models. RB-Modulation builds on concepts from stochastic optimal control to modulate the drift field of reverse diffusion dynamics, incorporating desired attributes (*e.g.*, style or content) via a terminal cost. Our Attention Feature Aggregation (AFA) module decouples content and style in the cross-attention layers and enables precise control over both. In addition, we derived theoretical connections between linear quadratic control and the denoising diffusion process, which led to the creation of RB-Modulation. Empirically, our method outperformed current state-of-the-art methods in stylization and content+style composition. To our best knowledge, this is the first training-free personalization framework using stochastic optimal control, which marks the departure from external adapters or ControlNets.

Limitation: We proposed a framework and demonstrated its efficacy by incorporating a style descriptor [43] in a pre-trained diffusion model [24]. The inherent limitations of the style descriptor or diffusion model might propagate into our framework. We believe these limitations can be addressed by appropriate replacements of the descriptor or generative prior in a plug-and-play manner.

Acknowledgements: This research has been supported by NSF Grant 2019844, a Google research collaboration award, and the UT Austin Machine Learning Lab. Litu Rout has been supported by Ju-Nam and Pearl Chew Presidential Fellowship and George J. Heuer Graduate Fellowship.

References

[1] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. "Zero-shot text-to-image generation". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 8821–8831.

[2] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. "Hierarchical text-conditional image generation with clip latents". In: *arXiv preprint arXiv:2204.06125* (2022).

[3] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. "High-resolution image synthesis with latent diffusion models". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 10684–10695.

[4] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. "Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding". In: *arXiv preprint arXiv:2205.11487* (2022).

[5] 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* 36 (2024).

[6] Tim Pearce, Tabish Rashid, Anssi Kanervisto, Dave Bignell, Mingfei Sun, Raluca Georgescu, Sergio Valcarcel Macua, Shan Zheng Tan, Ida Momennejad, Katja Hofmann, and Sam Devlin. "Imitating Human Behaviour with Diffusion Models". In: *The Eleventh International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=Pv1GPQzRrC8.

[7] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. "Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 22500–22510.

[8] Jiehui Huang, Xiao Dong, Wenhui Song, Hanhui Li, Jun Zhou, Yuhao Cheng, Shutao Liao, Long Chen, Yiqiang Yan, Shengcai Liao, et al. "ConsistentID: Portrait Generation with Multimodal Fine-Grained Identity Preserving". In: *arXiv preprint arXiv:2404.16771* (2024).

[9] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. "LoRA: Low-Rank Adaptation of Large Language Models". In: *International Conference on Learning Representations*. 2021.

[10] Viraj Shah, Nataniel Ruiz, Forrester Cole, Erika Lu, Svetlana Lazebnik, Yuanzhen Li, and Varun Jampani. "ZipLoRA: Any subject in any style by effectively merging loras". In: *arXiv preprint arXiv:2311.13600* (2023).

[11] Kihyuk Sohn, Nataniel Ruiz, Kimin Lee, Daniel Castro Chin, Irina Blok, Huiwen Chang, Jarred Barber, Lu Jiang, Glenn Entis, Yuanzhen Li, et al. "StyleDrop: Text-to-Image Generation in Any Style". In: *37th Conference on Neural Information Processing Systems (NeurIPS)*. Neural Information Processing Systems Foundation. 2023.

[12] Amir Hertz, Andrey Voynov, Shlomi Fruchter, and Daniel Cohen-Or. "Style aligned image generation via shared attention". In: *arXiv preprint arXiv:2312.02133* (2023).

[13] Haofan Wang, Qixun Wang, Xu Bai, Zekui Qin, and Anthony Chen. "InstantStyle: Free Lunch towards Style-Preserving in Text-to-Image Generation". In: *arXiv preprint arXiv:2404.02733* (2024).

[14] Jaeseok Jeong, Junho Kim, Yunjey Choi, Gayoung Lee, and Youngjung Uh. "Visual Style Prompting with Swapping Self-Attention". In: *arXiv preprint arXiv:2402.12974* (2024).

[15] Mauricio Delbracio and Peyman Milanfar. "Inversion by Direct Iteration: An Alternative to Denoising Diffusion for Image Restoration". In: *Transactions on Machine Learning Research* (2023).

[16] Litu Rout, Negin Raoof, Giannis Daras, Constantine Caramanis, Alexandros G Dimakis, and Sanjay Shakkottai. "Solving Inverse Problems Provably via Posterior Sampling with Latent Diffusion Models". In: *Thirty-seventh Conference on Neural Information Processing Systems*. 2023. URL: https://openreview.net/forum?id=XKBFdYwfRo.

[17] Litu Rout, Yujia Chen, Abhishek Kumar, Constantine Caramanis, Sanjay Shakkottai, and Wen-Sheng Chu. "Beyond First-Order Tweedie: Solving Inverse Problems using Latent Diffusion". In: *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024.

[18] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. "Null-text inversion for editing real images using guided diffusion models". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 6038–6047.

[19] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. "An image is worth one word: Personalizing text-to-image generation using textual inversion". In: *arXiv preprint arXiv:2208.01618* (2022).

[20] Jiaming Song, Chenlin Meng, and Stefano Ermon. "Denoising Diffusion Implicit Models". In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=St1giarCHLP.

[21] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. "Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models". In: *arXiv preprint arXiv:2308.06721* (2023).

[22] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. "Adding conditional control to text-to-image diffusion models". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 3836–3847.

[23] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. "SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis". In: *The Twelfth International Conference on Learning Representations*. 2023.

[24] Pablo Pernias, Dominic Rampas, Mats Leon Richter, Christopher Pal, and Marc Aubreville. "Würstchen: An Efficient Architecture for Large-Scale Text-to-Image Diffusion Models". In: *The Twelfth International Conference on Learning Representations*. 2024. URL: https://openreview.net/forum?id=gU58d5QeGv.

[25] Yoad Tewel, Rinon Gal, Gal Chechik, and Yuval Atzmon. "Key-locked rank one editing for text-to-image personalization". In: *ACM SIGGRAPH 2023 Conference Proceedings*. 2023, pp. 1–11.

[26] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. "Multi-concept customization of text-to-image diffusion". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 1931–1941.

[27] Wenhu Chen, Hexiang Hu, Yandong Li, Nataniel Ruiz, Xuhui Jia, Ming-Wei Chang, and William W Cohen. "Subject-driven text-to-image generation via apprenticeship learning". In: *Advances in Neural Information Processing Systems* 36 (2024).

[28] Luming Tang, Nataniel Ruiz, Qinghao Chu, Yuanzhen Li, Aleksander Holynski, David E Jacobs, Bharath Hariharan, Yael Pritch, Neal Wadhwa, Kfir Aberman, et al. "Realfill: Reference-driven generation for authentic image completion". In: *arXiv preprint arXiv:2309.16668* (2023).

[29] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Wei Wei, Tingbo Hou, Yael Pritch, Neal Wadhwa, Michael Rubinstein, and Kfir Aberman. "Hyperdreambooth: Hypernetworks for fast personalization of text-to-image models". In: *arXiv preprint arXiv:2307.06949* (2023).

[30] Qixun Wang, Xu Bai, Haofan Wang, Zekui Qin, and Anthony Chen. "Instantid: Zero-shot identity-preserving generation in seconds". In: *arXiv preprint arXiv:2401.07519* (2024).

[31] Martin Nicolas Everaert, Marco Bocchio, Sami Arpa, Sabine Süsstrunk, and Radhakrishna Achanta. "Diffusion in style". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 2251–2261.

[32] Xun Huang and Serge Belongie. "Arbitrary style transfer in real-time with adaptive instance normalization". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 1501–1510.

[33] Lars Holdijk, Yuanqi Du, Ferry Hooft, Priyank Jaini, Berend Ensing, and Max Welling. "Stochastic Optimal Control for Collective Variable Free Sampling of Molecular Transition Paths". In: *Advances in Neural Information Processing Systems* 36 (2024).

[34] Wendell H Fleming and Raymond W Rishel. *Deterministic and stochastic optimal control*. Vol. 1. Springer Science & Business Media, 2012.

[35] Pratik Chaudhari, Adam Oberman, Stanley Osher, Stefano Soatto, and Guillaume Carlier. "Deep relaxation: partial differential equations for optimizing deep neural networks". In: *Research in the Mathematical Sciences* 5 (2018), pp. 1–30.

[36] Evangelos Theodorou, Freek Stulp, Jonas Buchli, and Stefan Schaal. "An iterative path integral stochastic optimal control approach for learning robotic tasks". In: *IFAC Proceedings Volumes* 44.1 (2011), pp. 11594–11601.

[37] René Carmona, François Delarue, et al. *Probabilistic theory of mean field games with applications I-II*. Springer, 2018.

[38] Brian D.O. Anderson. "Reverse-time diffusion equation models". In: *Stochastic Processes and their Applications* 12.3 (1982), pp. 313–326.

[39] Aapo Hyvärinen and Peter Dayan. "Estimation of non-normalized statistical models by score matching." In: *Journal of Machine Learning Research* 6.4 (2005).

[40] Pascal Vincent. "A connection between score matching and denoising autoencoders". In: *Neural computation* 23.7 (2011), pp. 1661–1674.

[41] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. "Elucidating the design space of diffusion-based generative models". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 26565–26577.

[42] Qinsheng Zhang and Yongxin Chen. "Fast Sampling of Diffusion Models with Exponential Integrator". In: *The Eleventh International Conference on Learning Representations*. 2022.

[43] Gowthami Somepalli, Anubhav Gupta, Kamal Gupta, Shramay Palta, Micah Goldblum, Jonas Geiping, Abhinav Shrivastava, and Tom Goldstein. "Measuring Style Similarity in Diffusion Models". In: *arXiv preprint arXiv:2404.01292* (2024).

[44] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. "Learning transferable visual models from natural language supervision". In: *International conference on machine learning*. PMLR. 2021, pp. 8748–8763.

[45] Tamer Basar, Sean Meyn, and William R Perkins. "Lecture notes on control system theory and design". In: *arXiv preprint arXiv:2007.01367* (2020).

[46] HJ Kappen. "Stochastic optimal control theory". In: *ICML, Helsinki, Radbound University, Nijmegen, Netherlands* (2008).

[47] Tianrong Chen, Jiatao Gu, Laurent Dinh, Evangelos Theodorou, Joshua M Susskind, and Shuangfei Zhai. "Generative Modeling with Phase Stochastic Bridge". In: *The Twelfth International Conference on Learning Representations*. 2023.

[48] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. "Emerging properties in self-supervised vision transformers". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 9650–9660.

[49] Litu Rout, Advait Parulekar, Constantine Caramanis, and Sanjay Shakkottai. "A Theoretical Justification for Image Inpainting using Denoising Diffusion Probabilistic Models". In: *arXiv preprint arXiv:2302.01217* (2023).

[50] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. "Flow matching for generative modeling". In: *arXiv preprint arXiv:2210.02747* (2022).

[51] Xingchao Liu, Chengyue Gong, et al. "Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow". In: *The Eleventh International Conference on Learning Representations*. 2022.

[52] Bradley Efron. "Tweedie's formula and selection bias". In: *Journal of the American Statistical Association* 106.496 (2011), pp. 1602–1614.

[53] Jonathan Ho, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 6840–6851.

[54] 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*. 2021. URL: https://openreview.net/forum?id=PxTIG12RRHS.

[55] Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, José Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. "Muse: Text-to-image generation via masked generative transformers". In: *Proceedings of the 40th International Conference on Machine Learning*. 2023, pp. 4055–4075.

A Additional Theoretical Results

In this section, we restate the propositions more precisely and provide their technical proofs. First, we recall standard terminologies from optimal control literature [34]. For $0 \leq t_0 \leq t \leq t_N \leq 1$, the cost function associated with the controller $u(\cdot)$ is defined by the integral:

$$V(u; \mathbf{x}_0, t_0) = \int_{t_0}^{t_N} \ell\left(X_t^u, u, t\right) dt + h\left(X_{t_N}^u\right), \quad X_{t_0}^u = \mathbf{x}_0, \tag{4}$$

where $\ell(\cdots)$ denotes a scalar valued function of the state X_t^u, controller $u(\cdot)$, and instantaneous time t. The value function $V^*(\mathbf{x}_0, t_0)$ is defined as the minimum value of $V(u; \mathbf{x}_0, t_0)$ over the set of admissible controllers $\mathcal{U}$, i.e.,

$$V^* = V^*(\mathbf{x}_0, t_0) = \min_{u \in \mathcal{U}} V(u; \mathbf{x}_0, t_0) = \min_{u \in \mathcal{U}} \int_{t_0}^{t_N} \ell\left(X_t^u, u, t\right) dt + h\left(X_{t_N}^u\right), \quad X_{t_0}^u = \mathbf{x}_0, \tag{5}$$

which satisfies a Partial Differential Equation (PDE) given below in **Theorem A.1**.

Theorem A.1 (HJB Equation, [34, 45])**.** *If V^* has continuous partial derivatives, then it must satisfy the following PDE, also known as Hamilton-Jacobi-Bellman (HJB) equation:*

$$-\frac{\partial V^*}{\partial t}\left(\mathbf{x}, t\right) = \min_{u \in \mathcal{U}} \left[H\left(\mathbf{x}, \nabla_{\mathbf{x}} V^*\left(\mathbf{x}, t\right), u, t\right) \coloneqq \ell\left(\mathbf{x}, u, t\right) + \left(\nabla_{\mathbf{x}} V^*\left(\mathbf{x}, t\right)\right)^T v\left(\mathbf{x}, u, t\right)\right].$$

Also, the Hamiltonian $H\left(\mathbf{x}, \nabla_{\mathbf{x}} V^*\left(\mathbf{x}, t\right), u, t\right)$, optimal controller $u^*(t)$ and the state trajectory $\mathbf{x}^*(t)$ must satisfy

$$\min_{u \in \mathcal{U}} H\left(\mathbf{x}^*(t), \nabla_{\mathbf{x}} V^*\left(\mathbf{x}^*(t), t\right), u, t\right) = H\left(\mathbf{x}^*(t), \nabla_{\mathbf{x}} V^*\left(\mathbf{x}^*(t), t\right), u^*(t), t\right).$$

A.1 Interpreting reverse-SDE as a solution to optimal control

For clarity, we restate the problem setup here and describe the main ideas from §4 in more details. **Problem setup:** We discuss a standard approach to derive the optimal controller in a special case of our control problem (2). We substitute $t \leftarrow 1 - t$ to account for the time reversal in the reverse-SDE (1). In this setup, $X_0^u \sim \mathcal{N}\left(0, \mathrm{I}_d\right)$ and $X_1^u \sim p_{data}$. We consider the following dynamic without the Brownian motion:

$$\mathrm{d}X_t^u = v(X_t^u, u, t)\mathrm{d}t, \quad X_{t_0}^u = \mathbf{x}_0, \tag{6}$$

where $0 \leq t_0 \leq t \leq t_N \leq 1$ and $v : \mathbb{R}^d \times \mathbb{R}^d \times [t_0, t_N] \to \mathbb{R}^d$ denotes the drift field. The optimal controller u^* can be derived by solving the Hamilton-Jacobi-Bellman (HJB) equation [34, 45], see Appendix A for details. By certainty equivalence, the same u^* applies to a more general case with the Brownian motion [47], where

$$\mathrm{d}X_t^u = v(X_t^u, u, t)\mathrm{d}t + \mathrm{d}W_t, \quad X_{t_0}^u = \mathbf{x}_0. \tag{7}$$

Therefore, without loss of generality, we analyze the reverse dynamic in the absence of the Brownian motion, and employ the same controller in more general cases with the Brownian motion.

Below, we consider a dynamical system whose drift field is chosen to minimize a transient trajectory cost and a terminal cost (weighted by γ) that enforces "closeness" to reference content x_1. **Proposition A.2** provides the structure of the optimal control in the limiting setting where $\gamma \to \infty$. Furthermore, suppose we replace x_1 with its conditional expectation (discussed in Remark A.3), the resulting dynamic, interestingly, is the standard reverse-SDE for the Orstein-Uhlenbeck (OU) diffusion process. This connection between optimal control (more precisely, classic Linear Quadratic Control) and the standard reverse-SDE provides us a path to study other diffusion problems (*e.g.* personalization [7, 12, 11, 13], image editing or inversion [18, 15, 16, 17, 49]) through the lens of stochastic optimal control.

Proposition A.2 (Linear optimal control with quadratic cost [47])**.** *Consider the control problem:*

$$\min_{u \in \mathcal{U}} \int_{t_0}^{1} \frac{1}{2} \left\|u(X_t^u, t)\right\|^2 dt + \frac{\gamma}{2} \left\|X_1^u - x_1\right\|_2^2,$$
$$\text{where } \mathrm{d}X_t^u = u(X_t^u, t)\, \mathrm{d}t, \quad X_{t_0}^u = x_0$$

Then, in the limit when $\gamma \to \infty$, the optimal controller is given by $u^* = \frac{x_1 - X_t^u}{1-t}$, which yields $\mathrm{d}X_t^u = \frac{x_1 - X_t^u}{1-t}\mathrm{d}t$ for the deterministic case and $\mathrm{d}X_t^u = \frac{x_1 - X_t^u}{1-t}\mathrm{d}t + \mathrm{d}W_t$ for the stochastic case.

The optimal controller for the problem presented in **Proposition A.2** can be derived using established techniques from control theory [34, 45, 46]; the specific form of the above result follows from [47] (but without their momentum term). The key steps in this derivation include: (1) computing the Hamiltonian, (2) applying the minimum principle theorem to derive a set of differential equations, and (3) taking the limit as $\gamma \to \infty$. These three steps are fundamental in deriving a closed-form solution. The final step is critical for satisfying hard terminal constraint and is essential for the practical implementation of **Algorithm 1** and **Algorithm 2**, as detailed in §4.

For generative modeling, the controlled dynamics described in **Proposition A.2** cannot be directly applied. This limitation arises because the optimal control u^* depends on the terminal state x_1, making it non-causal or reliant on future information. Inspired by recent advancements in flow-based generative models [50, 51], we make the optimal controller causal by replacing the terminal state with its conditional expectation given the current state, i.e., , *i.e.* $x_1 \leftarrow \mathbb{E}[X_1^u | X_t^u = \mathbf{x}_t]$. This modification results in a controlled dynamic that can be simulated to produce a generative model incorporating principles from optimal control, as elaborated in **Remark A.3**.

Remark A.3 (Connections between diffusion-based generative modeling and stochastic optimal control)**.** Following conditional diffusion models and optimal transport paths [50, 51], where $X_t^f = tX_0^f + (1-t)\epsilon$, the state variable X_t^u is equal in distribution to $X_{1-t}^f = (1-t)X_0^f + t\epsilon$, $\epsilon \sim \mathcal{N}(0, \mathrm{I}_d)$ after time reversal. Now, we use Tweedie's formula [52] to compute the posterior mean:

$$\mathbb{E}[X_1^u | X_t^u] = \frac{X_t^u}{1-t} + \frac{t^2}{1-t} \nabla \log p\left(X_t^u, 1-t\right). \tag{8}$$

Substituting the posterior mean in the controlled reverse dynamic of **Proposition A.2**, we arrive at

$$\begin{aligned} \mathrm{d}X_t^u &= \frac{(\mathbb{E}[X_1^u | X_t^u] - X_t^u)}{(1-t)} \mathrm{d}t + \mathrm{d}W_t \\ &= \Big[\frac{t}{(1-t)^2} X_t^u + \frac{t^2}{(1-t)^2} \nabla \log p(X_t^u, 1-t)\Big] \mathrm{d}t + \mathrm{d}W_t. \end{aligned}$$

We observe that the above equation is structurally the same as reverse-SDE associated with a forward Orstein-Uhlenbeck (OU) diffusion process. This relation between diffusion-based generative models and optimal control is further explored in the Appendices below.

Indeed, diffusion models [53, 54, 3, 23, 24] provide an effective approximation to the terminal state of a denoising process. This approximation has been used for a variety of generative modeling tasks. Also, the terminal state can be approximated using Tweedie's formula [52] with a learned score function [53] [1]. By utilizing these pre-trained diffusion models, we can employ the connection to optimal control as discussed above to develop practically implementable generative models that incorporates terminal objectives such as style and personalization. Consequently, the subsequent sections are dedicated to deriving the optimal controller assuming a known terminal state; we will approximate this in practice using Tweedie's formula as above.

A.2 Incorporating personalized style constraints through a terminal cost

In this section, we derive the optimal controller when we have access to the reference *style features* y_1 at the terminal time (instead of the content of the image encoded through x_1).

Proposition A.4. *Suppose $A \in \mathbb{R}^{k \times d}$ be a linear style extractor that operates on the terminal state $X_1^u \in \mathbb{R}^d$. Given reference style features y_1, consider the control problem:*

$$\min_{u \in \mathcal{U}} \int_{t_0}^{1} \frac{1}{2} \left\| u(X_t^u, t) \right\|^2 dt + \frac{\gamma}{2} \left\| A X_1^u - y_1 \right\|_2^2, \tag{9}$$

$$\textit{where } \mathrm{d}X_t^u = u(X_t^u, t)\, \mathrm{d}t, \quad X_{t_0}^u = x_0, \tag{10}$$

Then, in the limit when $\gamma \to \infty$, the optimal controller $u^* = \frac{\left(A^T A\right)^{-1} A^T (y_1 - A X_t^u)}{1-t}$, which yields the following controlled dynamic:

$$\mathrm{d}X_t^u = \frac{\left(A^T A\right)^{-1} A^T \left(y_1 - A X_t^u\right)}{1-t} \mathrm{d}t. \tag{11}$$

[1] Alternatively, when the reverse process is described by a probability flow ODE, a trained neural network can directly predict the terminal state [20].

Proof. We derive the closed-form solution of the optimal controller given a fixed terminal state condition. This is similar to [47], where the reverse process is accelerated using momentum (see also [46, 45] for further details on this approach). The distinction, however, lies in the treatment of the terminal constraint. For completeness, we provide full details of the proof below.

To derive the closed-form solution[2], recall from equation (5) that $\ell(\mathbf{x}_t, \mathbf{u}_t, t) = \frac{1}{2}\|\mathbf{u}_t\|^2$ and the terminal cost $h(\mathbf{x}_1) = \frac{\gamma}{2}\|A\mathbf{x}_1 - y_1\|^2$. Let $\mathbf{p}_t$ represent $\nabla_{\mathbf{x}} V^*(\mathbf{x}, t)$ in **Theorem A.1**. Then, the Hamiltonian of the control problem (9) is given by

$$\begin{aligned} H(\mathbf{x}_t, \mathbf{p}_t, \mathbf{u}_t, t) &= \ell(\mathbf{x}_t, \mathbf{u}_t, t) + \mathbf{p}_t^T \mathbf{u}_t \\ &= \frac{1}{2}\|\mathbf{u}_t\|^2 + \mathbf{p}_t^T \mathbf{u}_t. \end{aligned}$$

Since the minimizer of the Hamiltonian is $\mathbf{u}_t^* = -\mathbf{p}_t$, the value function becomes

$$V^* = \min_{\mathbf{u}_t} H(\mathbf{u}_t, \mathbf{p}_t, \mathbf{u}_t, t) = H(\mathbf{u}_t, \mathbf{p}_t, \mathbf{u}_t^*, t) = -\frac{1}{2}\|\mathbf{p}_t\|^2. \tag{12}$$

Now, we use minimum principle theorem [45] to obtain the following set of differential equations:

$$\frac{\mathrm{d}\mathbf{x}_t}{\mathrm{d}t} = \nabla_{\mathbf{p}} H(\mathbf{x}_t, \mathbf{p}_t, \mathbf{u}_t^*, t) = -\mathbf{p}_t; \tag{13}$$

$$\frac{\mathrm{d}\mathbf{p}_t}{\mathrm{d}t} = -\nabla_{\mathbf{x}} H(\mathbf{x}_t, \mathbf{p}_t, \mathbf{u}_t^*, t) = 0; \tag{14}$$

$$\mathbf{x}_{t_0} = x_0; \tag{15}$$

$$\mathbf{p}_{t_N} = \nabla_{\mathbf{x}} h(\mathbf{x}_{t_N}, t_N) = \gamma A^T (A\mathbf{x}_{t_N} - y_1). \tag{16}$$

Integrating both sides of (13), we have

$$\int_{t_0}^{1} \mathrm{d}\mathbf{x}_t = -\int_{t_0}^{1} \mathbf{p}_t \mathrm{d}t = -\mathbf{p}(1 - t_0), \tag{17}$$

where the last equality is due to (14), which states that $\mathbf{p}_t$ is a constant independent of time t. This implies $\mathbf{x}_1 = \mathbf{x}_{t_0} - \mathbf{p}(1 - t_0)$. From (16), we know for $t_N = 1$ that

$$\begin{aligned} \mathbf{p}_1 &= \gamma A^T (A\mathbf{x}_1 - y_1) \\ &= \gamma \left(A^T A (x_0 - \mathbf{p}(1 - t_0)) - A^T y_1\right) \\ &= \gamma A^T A x_0 - \gamma A^T A \mathbf{p}_1 (1 - t_0) - \gamma A^T y_1 \end{aligned} \tag{18}$$

Rearranging (18) and solving for $\mathbf{p}_1$, we get

$$\begin{aligned} \mathbf{p}_1 &= \gamma \left(I + \gamma A^T A (1 - t_0)\right)^{-1} \left(A^T A x_0 - A^T y_1\right) \\ &= \left(\frac{I}{\gamma} + A^T A (1 - t_0)\right)^{-1} \left(A^T A x_0 - A^T y_1\right) = \mathbf{p} \end{aligned} \tag{19}$$

Passing (19) through the limit $\gamma \to \infty$, we get

$$\lim_{\gamma \to \infty} \mathbf{p} = \frac{\left(A^T A\right)^{-1} \left(A^T A x_0 - A^T y_1\right)}{1 - t_0}. \tag{20}$$

Therefore, the optimal control becomes $\mathbf{u}_t^* = -\mathbf{p} = -\frac{\left(A^T A\right)^{-1}\left(A^T A \mathbf{x}_t - A^T y_1\right)}{1-t}$, and the resulting dynamical system is given by

$$\mathrm{d}\mathbf{x}_t = \frac{\left(A^T A\right)^{-1} A^T (y_1 - A\mathbf{x}_t)}{1 - t}\mathrm{d}t,$$

for the deterministic process and

$$\mathrm{d}\mathbf{x}_t = \frac{\left(A^T A\right)^{-1} A^T (y_1 - A\mathbf{x}_t)}{1 - t}\mathrm{d}t + \mathrm{d}W_t,$$

for the stochastic process with the Brownian motion. This completes the statement of the proof. □

[2]With slight abuse of notation, we use $\mathbf{x}_t$ to denote X_t^u and $\mathbf{u}_t$ to denote $u(X_t^u, t)$ in the deterministic case.

Implications: The optimal controller depends on the reference *style features* y_1 at the terminal time (instead of the image content x_1 as in Appendix A.1). The reverse dynamic can be simulated in practice by using CSD [43] as a style feature extractor and replacing y_1 with the extracted style features from the expected terminal state $\mathbb{E}[X_1^u|X_t^u]$, as discussed in **Remark A.3**. This makes the controller drift causal and non-anticipating future information.

A.3 Incorporating personalized style constraint through modulation and a terminal cost

In this section, we study a control problem where the velocity field is a linear combination of the state and the control variable. This problem is interesting to study because of the following reason. The reverse-SDE dynamic of the standard OU process has a drift field of the form:

$$v\left(X_t, t\right) = -X_t - 2\nabla \log p(X_t, t).$$

For a Gaussian prior $X_0 \sim \mathcal{N}\left(0, \mathrm{I}\right)$, the law of the OU process satisfies $\nabla \log p\left(X_t, t\right) = -X_t$, and the corresponding drift field becomes $v\left(X_t, t\right) = X_t$. Our goal is to modulate this drift field using a controller $u\left(X_t^u, t\right)$. The result below provides the structure of the optimal control (again in the setting where the terminal objective is known; see Appendix A1).

Proposition A.5. *Suppose $A \in \mathbb{R}^{k\times d}$ be a linear style extractor that operates on the terminal state $X_1^u \in \mathbb{R}^d$. Let $\mathbf{p}_t$ denote $\nabla_{\mathbf{x}} V^*(\mathbf{x}, t)$ in HJB equation (A.1). Given reference style features y_1, consider the control problem:*

$$\min_{u\in\mathcal{U}} \int_{t_0}^{1} \frac{1}{2}\left\|u(X_t^u, t)\right\|^2 dt + \frac{\gamma}{2}\left\|AX_1^u - y_1\right\|_2^2, \tag{21}$$

$$\text{where } \mathrm{d}X_t^u = \left[X_t^u + u(X_t^u, t)\right]\mathrm{d}t, \quad X_{t_0}^u = x_0, \tag{22}$$

Then, the optimal controller becomes $u^*(t) = -\mathbf{p}_t$, where the instantaneous state $X_t^u = \mathbf{x}_t$ and $\mathbf{p}_t$ satisfy the following:

$$\begin{bmatrix}\mathbf{x}_t \\ \mathbf{p}_t\end{bmatrix} = \begin{bmatrix} x_0 e^t - \frac{\gamma}{2}A^T\left(A\mathbf{x}_1 - y_1\right)e^{1+t} + \frac{\gamma}{2}A^T\left(A\mathbf{x}_1 - y_1\right)e^{1-t} \\ \gamma A^T\left(A\mathbf{x}_1 - y_1\right)e^{1-t}\end{bmatrix}.$$

Proof. The proof of **Proposition A.5** is similar to **Proposition A.4**. One key distinction is the set of differential equations obtained using minimum principle theorem [45]. We begin with the Hamiltonian:

$$\begin{aligned} H(\mathbf{x}_t, \mathbf{p}_t, \mathbf{u}_t, t) &= \ell(\mathbf{x}_t, \mathbf{u}_t, t) + \mathbf{p}_t^T\left(\mathbf{u}_t + \mathbf{x}_t\right) \\ &= \frac{1}{2}\left\|\mathbf{u}_t\right\|^2 + \mathbf{p}_t^T\mathbf{u}_t + \mathbf{p}_t^T\mathbf{x}_t, \end{aligned}$$

which gives us the minimizer of the Hamiltonian $\mathbf{u}_t^* = -\mathbf{p}_t$ and its value function becomes $V^* = \min_{\mathbf{u}_t} H(\mathbf{u}_t, \mathbf{p}_t, \mathbf{u}_t, t) = H(\mathbf{u}_t, \mathbf{p}_t, \mathbf{u}_t^*, t) = -\frac{1}{2}\|\mathbf{p}_t\|^2 + \mathbf{p}_t^T\mathbf{x}_t$. By the minimum principle theorem [45],

$$\dot{\mathbf{x}}_t := \frac{\mathrm{d}\mathbf{x}_t}{\mathrm{d}t} = \nabla_{\mathbf{p}} H\left(\mathbf{x}_t, \mathbf{p}_t, \mathbf{u}_t^*, t\right) = -\mathbf{p}_t + \mathbf{x}_t; \tag{23}$$

$$\dot{\mathbf{p}}_t := \frac{\mathrm{d}\mathbf{p}_t}{\mathrm{d}t} = -\nabla_{\mathbf{x}} H\left(\mathbf{x}_t, \mathbf{p}_t, \mathbf{u}_t^*, t\right) = -\mathbf{p}_t; \tag{24}$$

$$\mathbf{x}_{t_0} = x_0; \tag{25}$$

$$\mathbf{p}_{t_N} = \nabla_{\mathbf{x}} h\left(\mathbf{x}_{t_N}, t_N\right) = \gamma A^T\left(A\mathbf{x}_{t_N} - y_1\right). \tag{26}$$

This leads to a coupled system of differential equations with boundary conditions as given below:

$$\begin{aligned} \begin{bmatrix}\dot{\mathbf{x}}_t \\ \dot{\mathbf{p}}_t\end{bmatrix} &= \begin{bmatrix}1 & -1 \\ 0 & -1\end{bmatrix}\begin{bmatrix}\mathbf{x}_t \\ \mathbf{p}_t\end{bmatrix}; \\ \mathbf{x}_{t_0} &= x_0; \\ \mathbf{p}_1 &= \gamma A^T\left(A\mathbf{x}_1 - y_1\right). \end{aligned}$$

This can be solved numerically using ODE solvers, see [34, 45] for details. Denote $\dot{\mathbf{q}}_t = \begin{bmatrix} \dot{\mathbf{x}}_t \\ \dot{\mathbf{p}}_t \end{bmatrix}$ and $\mathrm{M} = \begin{bmatrix} 1 & -1 \\ 0 & -1 \end{bmatrix}$. We seek a solution of the form $\mathbf{q}(t) = \mathbf{q}e^{\lambda t}$. If $\mathbf{q}(t)$ is a solution of the above problem, then it must satisfy the following eigen value problem:

$$\mathbf{q}e^{\lambda t}\lambda = \mathrm{M}\mathbf{q}e^{\lambda t}. \tag{27}$$

Writing the characteristic polynomial of (27), we get $\det(\mathrm{M} - \lambda \mathrm{I}) = 0$, which gives the eigen values $\lambda = \{1, -1\}$. Substituting these eigen values, we have

$$\begin{bmatrix} 0 & -1 \\ 0 & -2 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \mathbf{0}, \quad \begin{bmatrix} 2 & -1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \mathbf{0},$$

which gives two fundamental solutions. By combining these two, we obtain the final solution

$$\begin{bmatrix} \mathbf{x}_t \\ \mathbf{p}_t \end{bmatrix} = \omega \begin{bmatrix} 1 \\ 0 \end{bmatrix} e^t + \xi \begin{bmatrix} 1 \\ 2 \end{bmatrix} e^{-t},$$

where ω and ξ can be found using the boundary conditions. Since $\mathbf{x}_0 = x_0$ and $\mathbf{p}_1 = \gamma A^T (A\mathbf{x}_1 - y_1)$, we get $\omega = x_0 - \frac{\gamma}{2} A^T (A\mathbf{x}_1 - y_1) e$ and $\xi = \frac{\gamma}{2} A^T (A\mathbf{x}_1 - y_1) e$. Substituting the values of ω and ξ, we arrive at

$$\begin{bmatrix} \mathbf{x}_t \\ \mathbf{p}_t \end{bmatrix} = \begin{bmatrix} x_0 e^t - \frac{\gamma}{2} A^T (A\mathbf{x}_1 - y_1) e^{1+t} + \frac{\gamma}{2} A^T (A\mathbf{x}_1 - y_1) e^{1-t} \\ \gamma A^T (A\mathbf{x}_1 - y_1) e^{1-t} \end{bmatrix}.$$

This completes the proof of the proposition.

□

Summary: Though Appendices A.1-A.3, we have seen the connection between optimal control and diffusion based generation with a personalized terminal constraint. The general strategy has been to derive the optimal controller with known terminal state, and then replace the terminal state in the controller with its estimate using Tweedie's formula. While the controllers so far have an explicit form, in practice, the data distribution is not Gaussian, and thus, we do not have a closed-form expression for the drift of the controller.

This line of analysis, however, points to our method RB-Modulation. As discussed in §4, we incorporate a consistent style descriptor in our controller's terminal cost and numerically evaluate the drift of the controller at each reverse time step either through back propagation through the score network, or an approximation based on proximal gradient updates.

B Additional Experimental Evaluation

B.1 Implementation details

Baselines: We demonstrate the applicability of our method RB-Modulation with StableCascade [24] (released before April 2024). To our best knowledge, RB-Modulation is the first framework that introduces new capabilities to StableCascade by incorporating SOC and AFA modules. Since there are no existing training-free personalization baselines designed for StableCascade, we seek alternatives built on other comparable state-of-the-art models such as SDXL [23] and Muse [55][3].

Among alternate training-free baselines, InstantStyle [13] does not directly apply to StableCascade because it requires feature injection into specific layers of an IP-Adapter, which is not available for StableCascade. Similarly, StyleAligned [12] relies on DDIM inversion, which is currently applicable only to single-stage diffusion models. In contrast, StableCascade utilizes a two-stage diffusion process, making the application of standard DDIM inversion [20] infeasible. We run the official source code for InstantStyle[4] and StyleAligned[5]. In the absence of a style description, we use "image

[3] Note that StableCascade and SDXL have comparable performance in prompt alignment whereas StableCascade is more efficient due to a highly compressed semantic latent space [24].

[4] https://github.com/InstantStyle/InstantStyle

[5] https://github.com/google/style-aligned

Algorithm 2: Reference-Based Modulation (RB-Modulation) for large-scale generative models

Input: Diffusion time steps T, reference style image $\mathbf{z}_0$,
style descriptor $\Psi(\cdot)$, score network $s(\cdot, \cdot; \theta)$
Tunable parameters: Stepsize η, optimization steps M, proximal strength λ
Output: Personalized latent X_0^u

Initialize $\mathbf{x}_T \leftarrow \mathcal{N}(0, \mathrm{I}_d)$
Initialize controller $u \in \mathbb{R}^d$
for $t = T$ **to** 1 **do**
  Compute posterior mean $\mathbb{E}[X_0^u | X_t^u = \mathbf{x}_t] = \frac{\mathbf{x}_t}{\sqrt{\bar{\alpha}_t}} + \frac{(1-\bar{\alpha}_t)}{\sqrt{\bar{\alpha}_t}} s(\mathbf{x}_t, t; \theta)$
  Initialize optimization variable $\mathbf{x}_0 = 0$
  for $m = 1$ **to** M **do**
    Compute controller's cost $\mathcal{L}(\mathbf{x}_0) \coloneqq \|\Psi(\mathbf{z}_0) - \Psi(\mathbf{x}_0)\|_2^2 + \lambda \|\mathbf{x}_0 - \mathbb{E}[X_0^u | X_t^u = \mathbf{x}_t]\|_2^2$
    Update optimization variable $\mathbf{x}_0 = \mathbf{x}_0 - \eta \nabla_{\mathbf{x}_0} \mathcal{L}(\mathbf{x}_0)$
  end
  Compute previous state $\mathbf{x}_{t-1} = \mathrm{DDIM}(\mathbf{x}_0, \mathbf{x}_t)$ [20]
end
return X_0^u

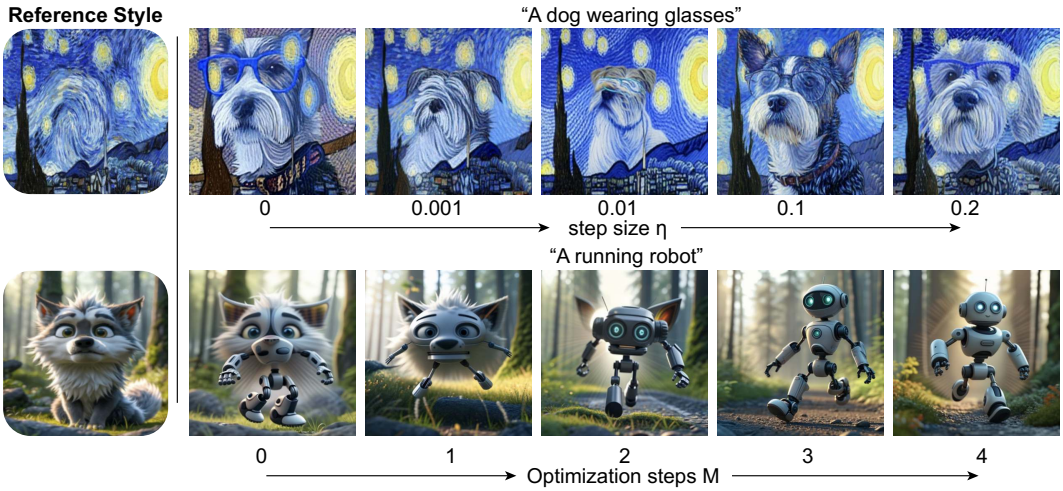

Figure 6: **Qualitative results of different tunable hyperparameters:** Improved style-prompt disentanglement are shown when increasing to our best configurations optimization step size $\eta = 0.1$ and optimization steps $M = 3$.

in style" for DDIM inversion in StyleAligned. Following InstantStyle [13], we also compare with IP-Adapter. We include the quantitative comparison in Table 2, and only compare qualitatively with stronger baselines in Figure 3.

For completeness, we also compare with training-based baselines: StyleDrop [11] and ZipLoRA [10]. Since the official codebase for StyleDrop[6] and ZipLoRA[7] are not publicly available, we use the third-party implementation and follow the training details in the corresponding papers. It takes 5 minutes for training StyleDrop for 1000 steps and 20 minutes for training each LoRA for ZipLoRA. We train each LoRA with only one reference image for both content and styles to make a fair comparison with other methods. Similarly, we train StyleDrop with only one reference image. When a style description is not provided, we follow the original paper [11] and use "in a [v*] style" instead.

Tunable parameters. Our method introduces only two hyper-parameters: stepsize η and optimization steps M in **Algorithm 1**. We use DDIM sampling with $\eta = 0.1$ and $M = 3$ for all the experiments.

[6] https://github.com/aim-uofa/StyleDrop-PyTorch
[7] https://github.com/mkshing/ziplora-pytorch

Figure 7: **Impact of style descriptions in the prompt:** (a) When style descriptions are provided, all methods yield better results. (b) Without style descriptions (*e.g.*, hard for users to describe in text), alternative methods could struggle to capture the intended style in the reference image. Our method offers consistent stylization even without explicit style descriptions.

Content-style composition. The prompt-guided content-style composition task introduces a new layer of complexity beyond stylization §6.1. This task necessitates the disentanglement of the text prompt, reference style image, and reference content image through additional conditioning. Such complexity poses significant challenges for DDIM inversion [20] and attention caching mechanisms [12] due to the inherent dependencies on multiple reverse paths.

Our AFA module effectively addresses these challenges. It manipulates transformer layers to easily incorporate these additional conditions. The content information is integrated in a manner similar to the style information. Specifically, we use a pre-trained ViT-L/14 model to extract content features in the SOC framework and update the latent embeddings concurrently via the AFA module, using an additional set of keys and values illustrated in Figure 2.

Furthermore, to better preserve the identity of the foreground content, we extract the desired content using LangSAM[8] based on the content prompt. This step is optional but offers more user control when multiple subjects are present in the reference image.

B.2 Implementation using large-scale diffusion models

The exact implementation of our control problem (3) is given in **Algorithm 1**, which follows from our theoretical insights. In practice, our controller encounters a challenge when the generative model contains billions of parameters as in StableCascade [24] due to back propagation through the score network, as discussed in §4. Our strategy to overcome this practical challenge involves a proximal gradient update, given in Line 7-8 of **Algorithm 2**. To accelerate the sampling process, we run a few steps ($M = 3$) of gradient descent after initializing $\mathbf{x}_0 = \mathbb{E}\left[X_0^u | X_t^u = \mathbf{x}_t\right]$, resulting in only two hyperparameters to tune: stepsize η and the number of optimization steps M. Further, since the CSD model expects a clean image to extract style features, we apply the previewer model available in StableCascade on the terminal state before extracting style features. After obtaining the final personalized latent using our **Algorithm 1** and **Algorithm 2**, we follow the decoding process as per the inference pipeline of the adopted generative model.

B.3 Impact of hyperparameters on controlling style and content features

As detailed in §4 and the ablation study in §6.1, SOC helps disentangle the style and the prompt information by updating the drift field in the standard reverse-SDE. We study the impact of the two hyperparameters present in **Algorithm 1** and **Algorithm 2** that enables this disentanglement, as shown in Figure 6. We found better disentanglement when the step size $\eta = 0.1$ and the number of optimization steps $M = 3$. However, increasing the step size further results in style image information leaking into the output (top row). Additionally, adding more optimization steps increases computational overhead without yielding much performance gain (bottom row).

B.4 Style description in text prompts for better assimilation of unique styles

In addition to the quantitative analysis in §6.1, Figure 7 demonstrates that our method generates consistent stylized results with and without the style description. In contrast, the alternatives fail to accurately follow the prompt when the style description is absent. Although all results show

[8] https://github.com/luca-medeiros/lang-segment-anything

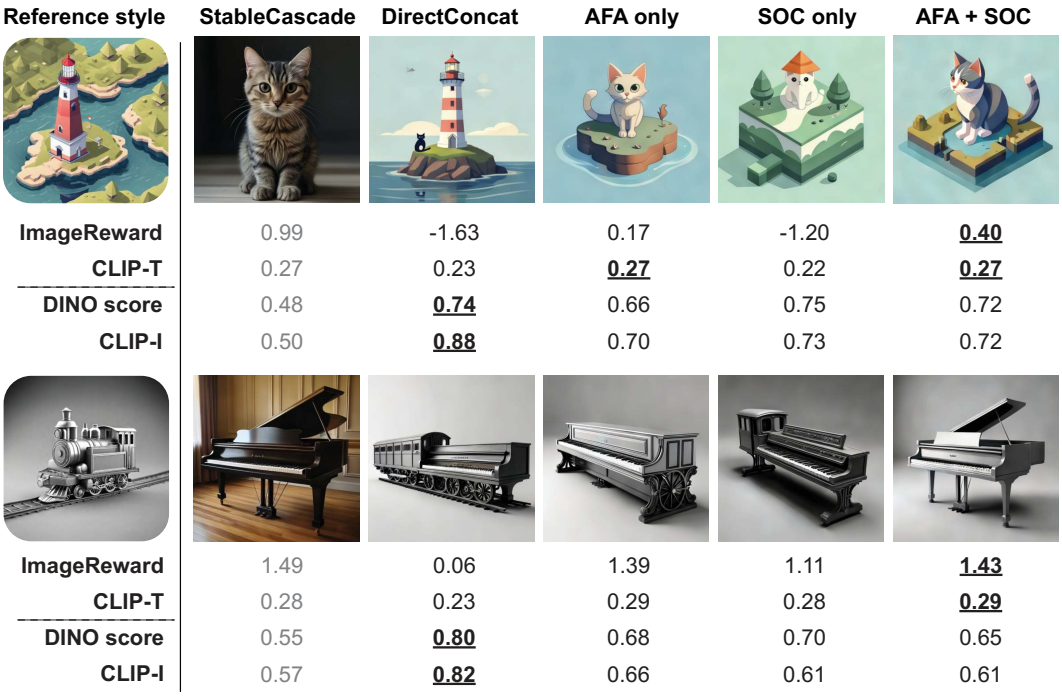

Figure 8: **Comparison of different evaluation metrics:** The StableCascade output is provided for reference because it doesn't use the reference style image. The highest score for each metric is marked **bold** with underscore. We compare four metrics: ImageReward and CLIP-T score for prompt alignment, DINO and CLIP-I score for style alignment. The prompt for the top row is "A cat" and for the bottom row is "A piano".

noticeable improvement when the style description is provided, it is often challenging for users to describe styles in many real-world scenarios. We believe our early results by RB-Modulation will pave the way for interesting future research along this direction.

We present additional qualitative results on stylization with (Figure 9) and without (Figure 10) style descriptions using StyleAligned dataset [12]. Our results consistently align with the reference style and the prompt, while other methods encounter several issues: (1) difficulty in following prompt guidance, (2) information leakage from the style reference image, and (3) failure to achieve reasonable prompt/style alignment in the absence of style descriptions.

B.5 Challenges of evaluation metrics in measuring style and content leakage

In §6, we discussed the limitations of metrics used in previous works [11, 12, 10], such as DINO [48] and CLIP-I score [44]. To quantify these limitations, we use results from our ablation study shown in Figure 4. As illustrated in Figure 8, DINO and CLIP-I scores are not well-suited for measuring style similarity in the presence of content leakage. This is because images with high semantic correlations to the reference style image consistently receive higher scores. For instance, in the top row, although the last two columns visually align more closely with the isometric illustration styles of the reference image, the DirectConcat output featuring a lighthouse receives higher scores. The margin is particularly pronounced for CLIP-I score.

A similar observation can be made in the bottom row, where images containing train-related objects receive higher scores regardless of their stylistic similarity. Conversely, images with less content leakage (as seen in the last column) are assigned lower scores. This indicates that DINO and CLIP-I scores prioritize semantic content over stylistic fidelity, thus failing to accurately measure style similarity in scenarios where content leakage prevails.

On the other hand, our final method (last column), which combines AFA and SOC, demonstrates high scores for both prompt alignment metrics: ImageReward [5] and CLIP-T [44]. This method also

shows higher user preference, as evidenced in Table 1. In contrast, the DirectConcat results suffer from information leakage and poor alignment with the prompt, resulting in significantly lower or even negative reward scores.

In the ablation study, our primary focus is on the disentanglement of prompts and reference styles. The conventional metrics fail to accurately reflect true performance due to information leakage. Consequently, we emphasize qualitative demonstrations and place greater importance on user study results, as shown in Table 1, similar to previous approaches [12, 11].

B.6 More qualitative results on stylization and content-style composition

We also showcase results on consistent style generation using user defined prompts in Figure 11. Our results with different prompts consistently align with the styles while introducing various scenarios following the prompts. The other methods face challenges like information leakage (*e.g.* hiking boots and the monocular) and monotonous scenes (*e.g.* InstantStyle). Note that the original StyleDrop paper [11] has mentioned its difficulty when training with one image without description. We keep the results for completeness even though they are less satisfying. Besides, we also demonstrate more qualitative results for content-style composition in Figure 13.

B.7 Human evaluation to discern highly subjective nature of style

We conduct a user study with 155 participants via Amazon Mechanical Turk using 100 styles from the StyleAligned dataset [12]. The study requires no personally identifiable information of the participants. There is no risk incurred and no vulnerable population. The standard guidelines have been followed while conducting the user study.

We first provide participants with instructions to familiarize them with the relevant terminologies. For each style, we randomly sample three outputs using three different prompts. Participants see two rows of model outputs in random order (3 images per row) and answer 3 questions, as illustrated in Figure 12.

1. In which row below, the images align better with the reference style image?
2. In which row below, the images align better with the reference text prompt above each image?
3. In which row below, the images overall align better with the reference style image AND the text prompt above each image AND with high quality?

For each question, participants choose one of three options. We collect 8 responses for each question, with each question comparing our method against one of the alternatives. In total, we gathered 7,200 responses.

B.8 Failure cases of training-free stylization using RB-Modulation

In Figure 14, we illustrate stylization of different letters using a single reference style image. Although our method captures the intended style and generates prompted letters, we notice that there is an inherent tendency to generate upper-case letters (Figure 14 (a)), even though it is prompted to generate lower-case letters. Upon further investigation, we observed that this issue stems from the underlying generative model StableCascade, as shown in Figure 14 (b). This highlights a crucial limitation of our method. As a training-free method, RB-Modulation shares a concern with other training-free methods [13, 12, 14] that the performance is influenced by the original generative prior.

C Broader impact statement

Social impact: Image stylization and content-style composition based on diffusion models potentially have both positive and negative social impact. This technology provides an easy-to-use tool to the general public for image generation which can help visualize their artistic ideas. On the other hand, our work on stylization and content-style composition poses a risk of generating arts that closely mimic or infringe upon existing copyrighted material, leading to legal and ethical issues. More

broadly, our method inherits the risks from T2I models which are capable of generating fake contents that can be misused by malicious users.

Safeguards: We build on StableCascade [24], which has a mechanism to filter offensive image generations. Since our method RB-Modulation builds on this pre-trained generative model, we inherit these safeguards.

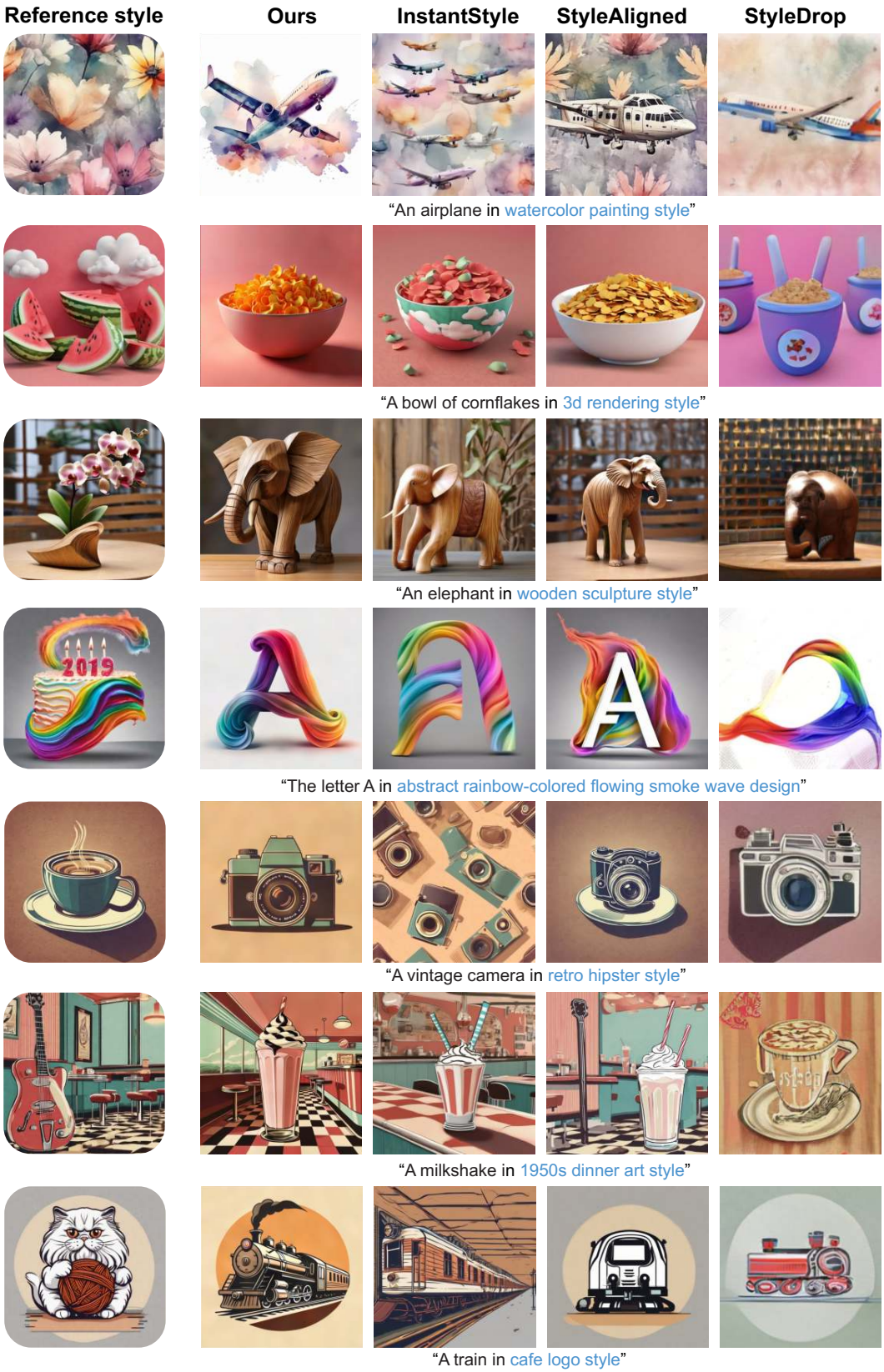

Figure 9: **Additional qualitative results for stylization with style description:** While the alternative methods face challenges like following the prompts (*e.g.*, multiple airplanes instead of an airplane) and information leakage (*e.g.*, the clouds on the cornflake bowl and the guitar in the milkshake image), our method demonstrates strong performance on both prompt and style alignment. Style description is in blue.

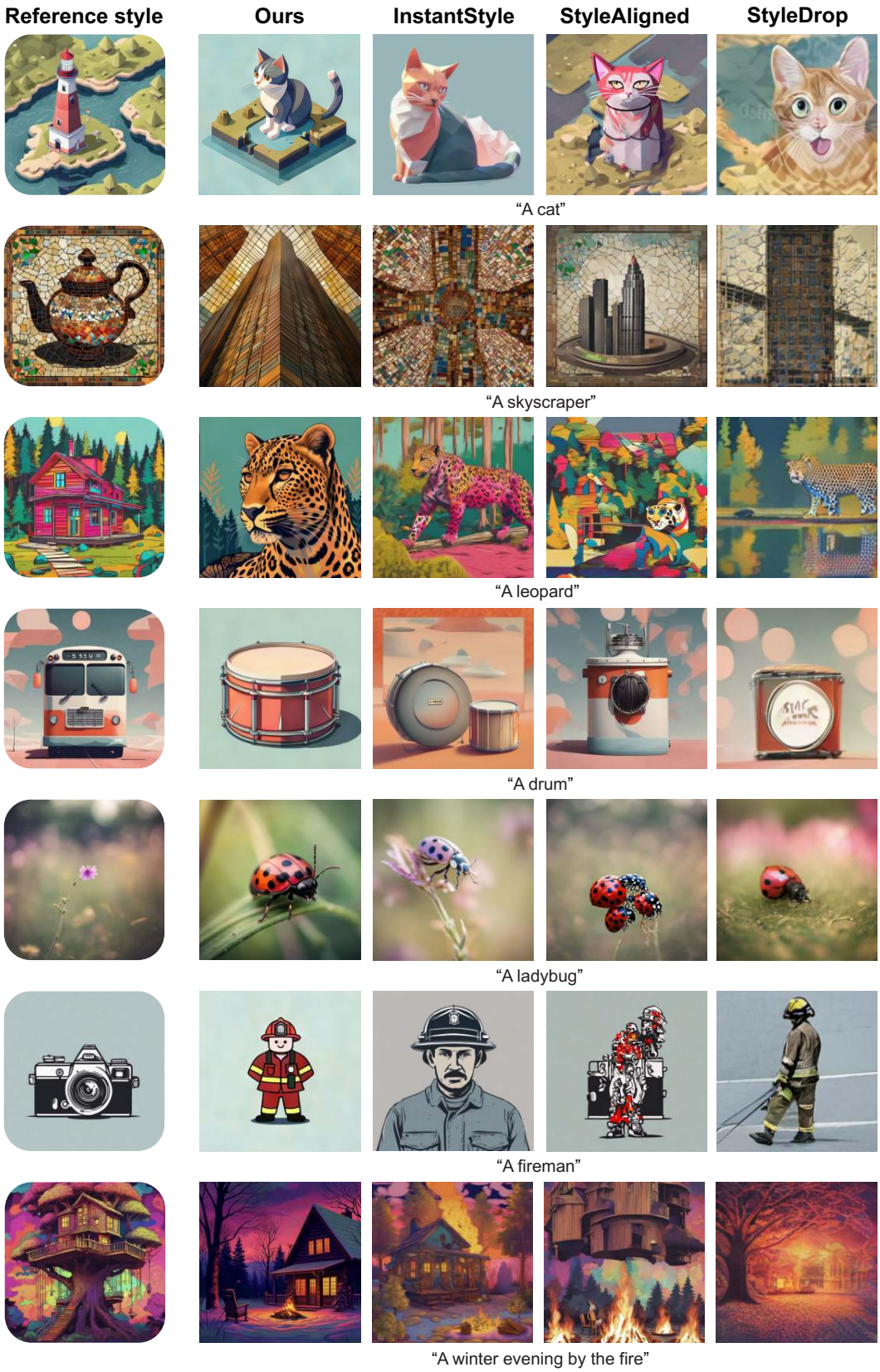

Figure 10: **Additional qualitative results for stylization without style description:** StyleAligned and StyleDrop show severe performance drop after removing the style descriptions (*e.g.*, see fireman and cat images). InstantStyle results show more information leakage (*e.g.*, the pink ladybug and leopard), whereas no obvious performance drop is observed in our results.

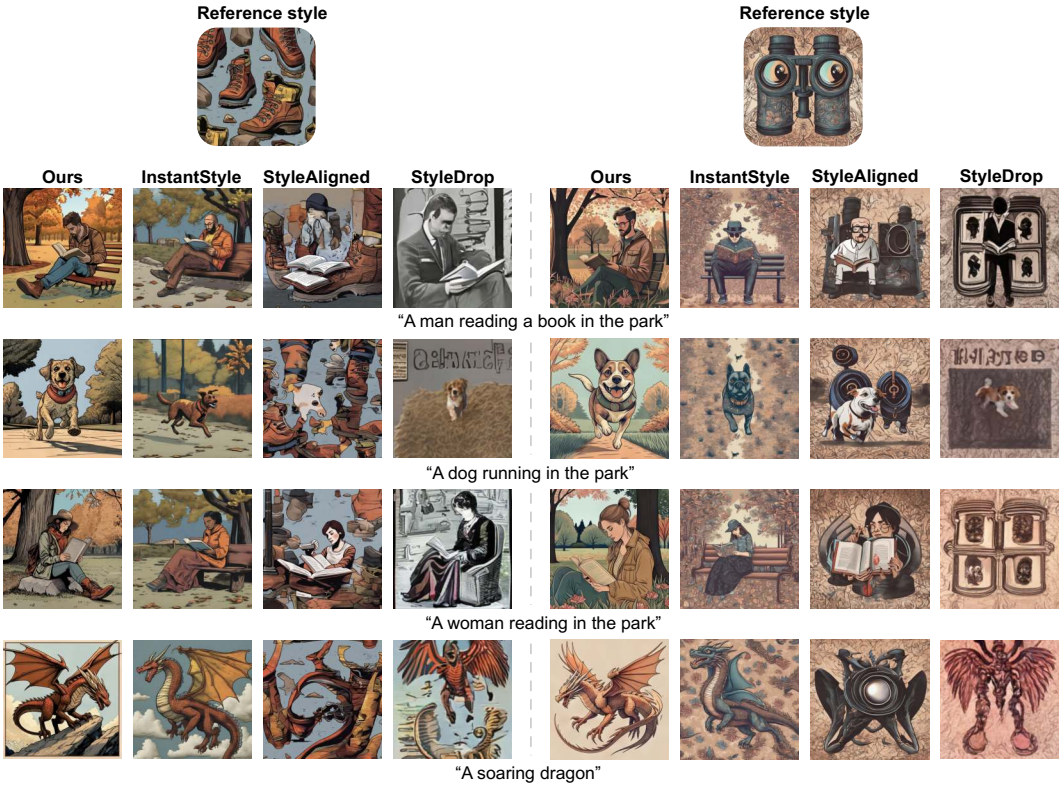

Figure 11: **Additional qualitative results for consistent stylization for user defined prompts:** With no style description, our results demonstrate more diversity while following the styles and prompts. InstantStyle results show monotonous scenes and StyleAligned results suffer from severe information leakage. We report StyleDrop results for completeness and it is known to perform worse with no style description and single training image [11].

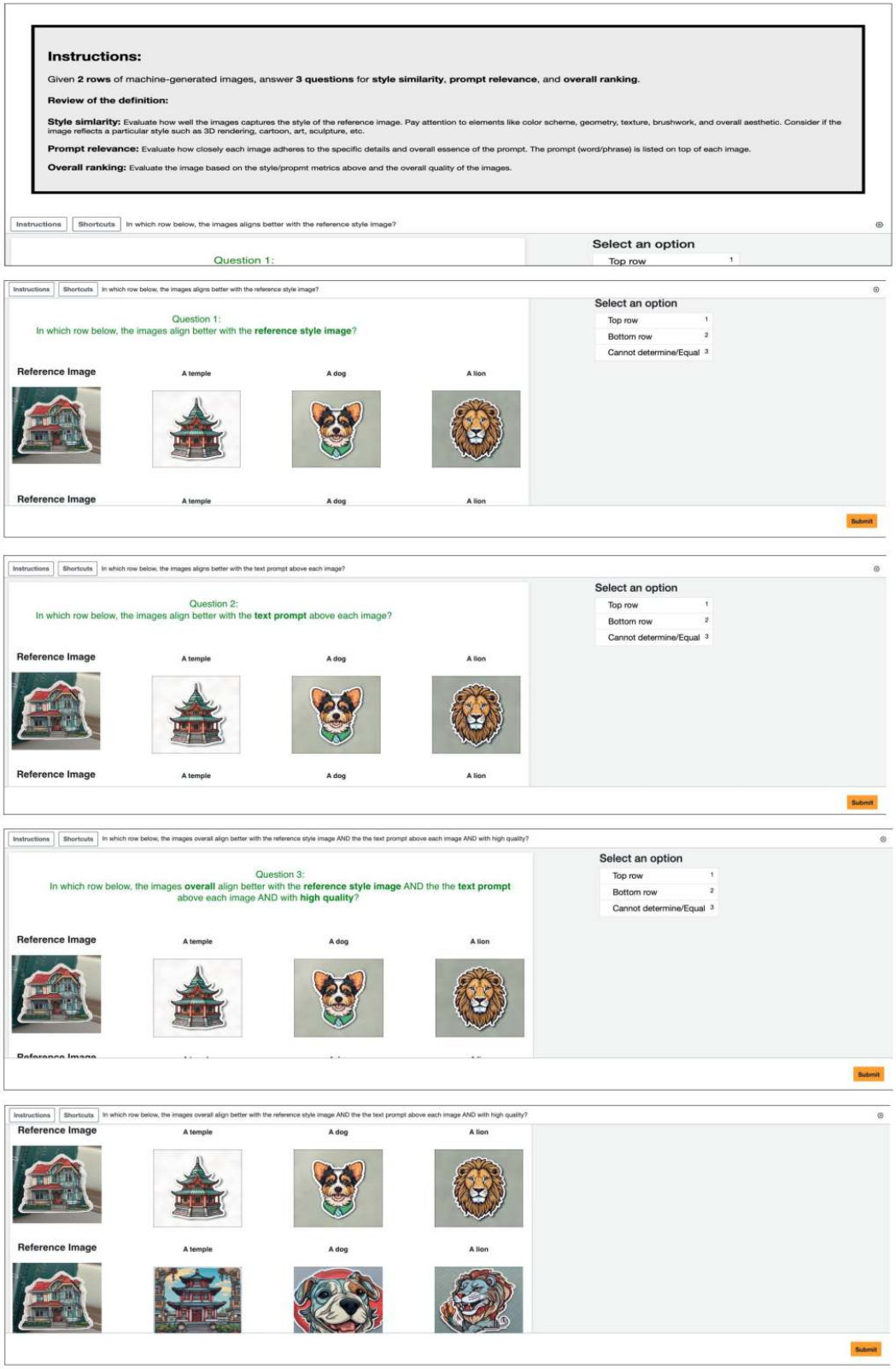

Figure 12: **User study interface:** Three randomly sampled outputs are shown for each method given a style reference image, forming two rows of images. The users are asked to answer three questions on (1) style alignment (2) prompt alignment and (3) overall alignment and quality.

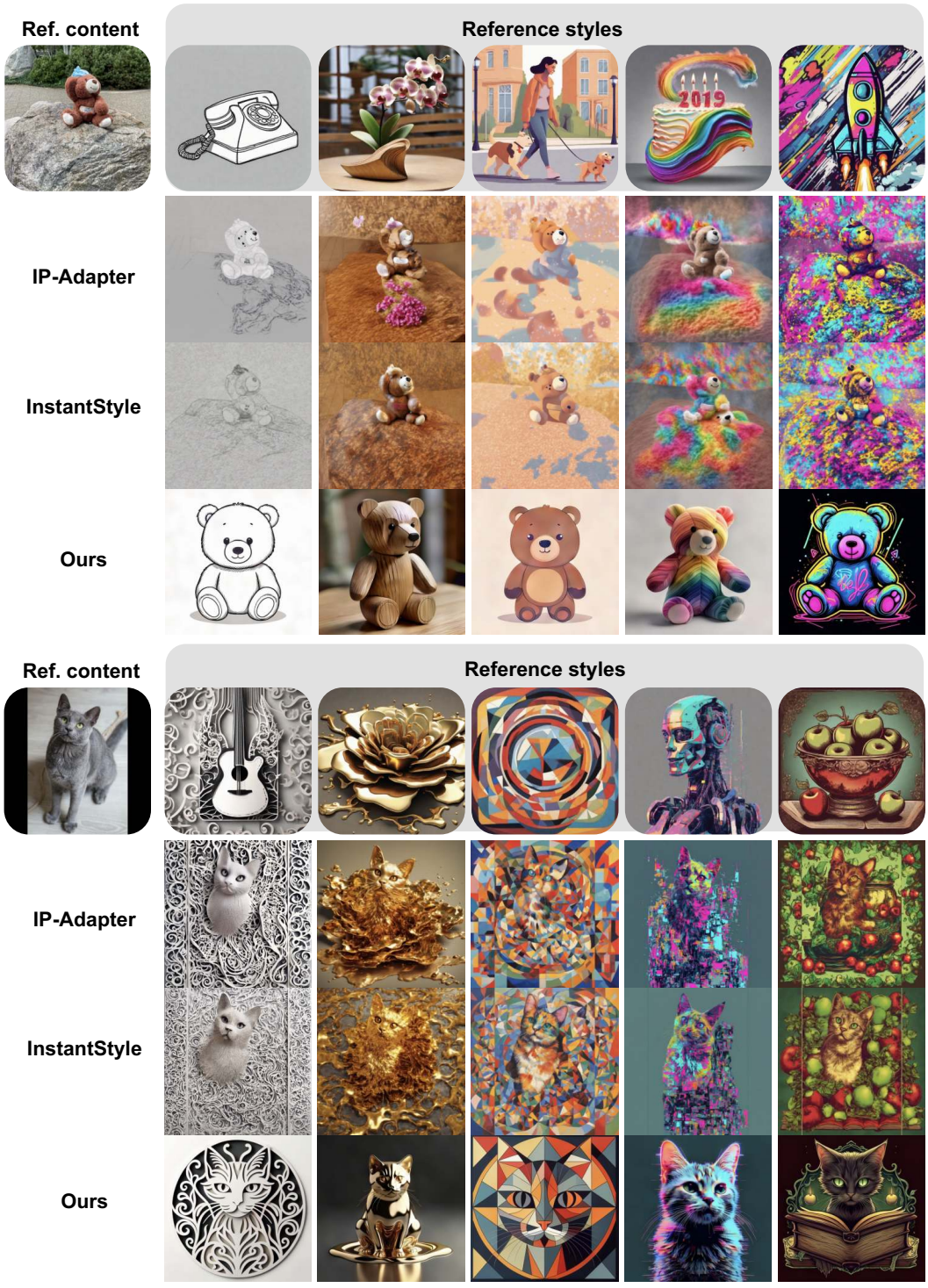

Figure 13: **Additional qualitative results for content-style composition:** Our results show better prompt and style alignment while preserving reference content without leaking contents from the reference style images (*e.g.* background of the first column and fruits in the last column,). Unlike compared baselines, our method is not restricted to a fixed pose of the reference content image, illustrating sample diversity.

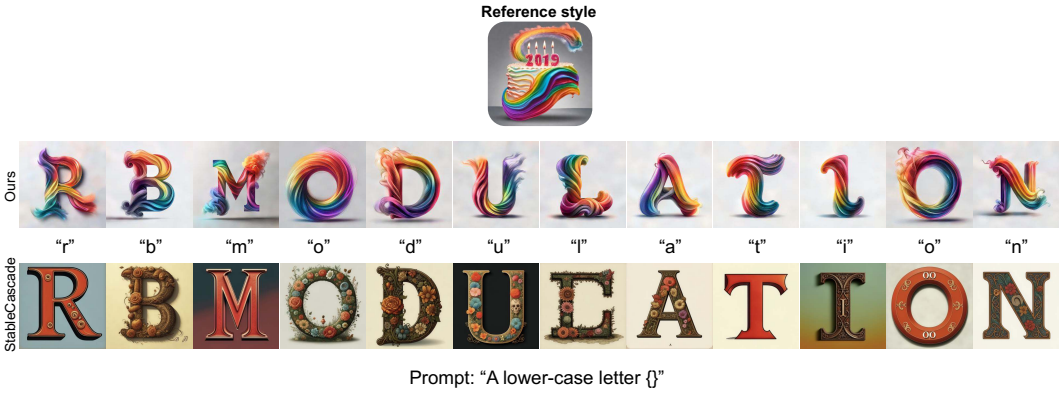

Figure 14: **Failure cases for stylization:** The top row shows the results of our method, RB-Modulation, while the bottom row displays the results of the backbone, StableCascade. Notably, the stylized images do not adhere to the prompt,"lower-case letter". This highlights the limitations imposed by the pre-trained generative priors on the capabilities of training-free personalization models (top row).