# Subject-driven Text-to-Image Generation via Preference-based Reinforcement Learning


**Yanting Miao**
Department of Computer Science
University of Waterloo, Vector Institute
y43miao@uwaterloo.ca

**William Loh**
Depart of Computer Science
University of Waterloo, Vector Institute
wmloh@uwaterloo.ca

**Suraj Kothawade**
Google
skothawade@google.com

**Pascal Poupart**
Depart of Computer Science
University of Waterloo, Vector Institute
ppoupart@uwaterloo.ca

**Abdullah Rashwan** *
Google
arashwan@google.com

**Yeqing Li**
Google
yeqing@google.com


## Abstract


Text-to-image generative models have recently attracted considerable interest, enabling the synthesis of high-quality images from textual prompts. However, these models often lack the capability to generate specific subjects from given reference images or to synthesize novel renditions under varying conditions. Methods like DreamBooth and Subject-driven Text-to-Image (SuTI) have made significant progress in this area. Yet, both approaches primarily focus on enhancing similarity to reference images and require expensive setups, often overlooking the need for efficient training and avoiding overfitting to the reference images. In this work, we present the $\lambda$-Harmonic reward function, which provides a reliable reward signal and enables early stopping for faster training and effective regularization. By combining the Bradley-Terry preference model, the $\lambda$-Harmonic reward function also provides preference labels for subject-driven generation tasks. We propose Reward Preference Optimization (RPO), which offers a simpler setup (requiring only 3% of the negative samples used by DreamBooth) and fewer gradient steps for fine-tuning. Unlike most existing methods, our approach does not require training a text encoder or optimizing text embeddings and achieves text-image alignment by fine-tuning only the U-Net component. Empirically, $\lambda$-Harmonic proves to be a reliable approach for model selection in subject-driven generation tasks. Based on preference labels and early stopping validation from the $\lambda$-Harmonic reward function, our algorithm achieves a state-of-the-art CLIP-I score of 0.833 and a CLIP-T score of 0.314 on DreamBench. Our Pytorch implementation is available at https://github.com/andrew-miao/RPO.


## 1 Introduction

In the evolving field of generative AI, text-to-image diffusion models [28, 13, 26, 29, 27, 23, 24] have demonstrated remarkable abilities in rendering scenes that are both imaginative and contextually

---

*This project was completed during work at Google

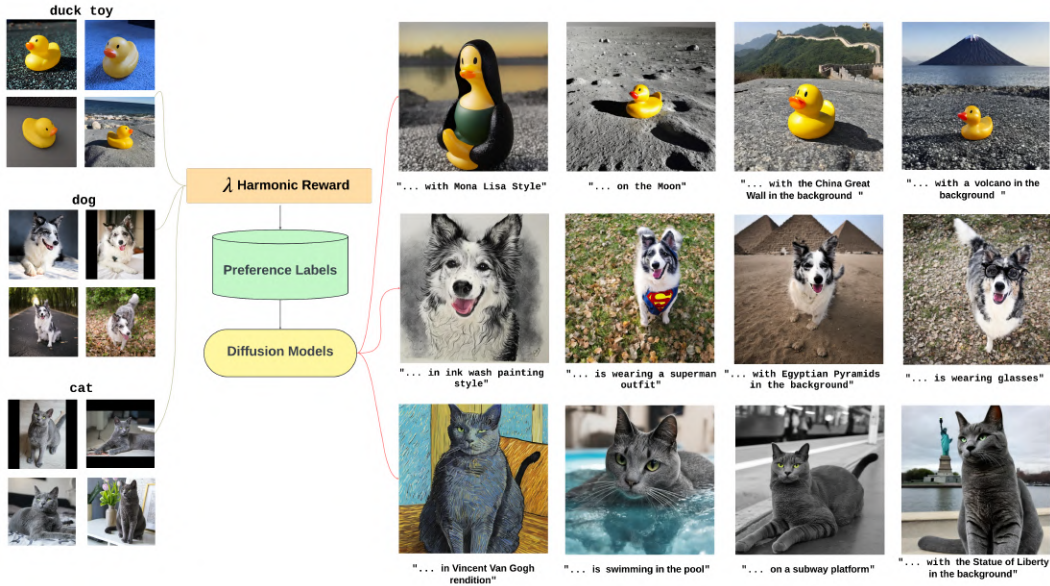

38th Conference on Neural Information Processing Systems (NeurIPS 2024).


Figure 1: We illustrate the $\lambda$-Harmonic reward function applied to the subject-driven generation task. Leveraging preference labels produced by the $\lambda$-Harmonic reward function, alongside a few reference images, our preference-based algorithm efficiently generates unseen scenes that are both faithful to the reference images and the textual prompts.

appropriate. However, these models often struggle with tasks that require the portrayal of specific subjects within text prompts. For instance, if provided with a photo of your cat, current diffusion models are unable to generate an image of your cat situated in the castle of your childhood dreams. This challenge necessitates a deep understanding of subject identity. Consequently, ***subject-driven*** text-to-image generation has attracted considerable interest within the community. Chen et al. [7] have noted that this task requires complex transformations of reference images. Additionally, Ruiz et al. [23] have highlighted that detailed and descriptive prompts about specific objects can lead to varied appearances in subjects. Thus, traditional image editing approaches and existing text-to-image models are ill-suited for subject-driven tasks.

Current subject-driven text-to-image generation methods are less expressive and expensive. Textual Inversion [12] performs poorly due to the limited expressiveness of frozen diffusion models. Imagic [15] is both time-consuming and resource-intensive during the fine-tuning phase. It requires text-embedding optimization for each prompt, fine-tuning of diffusion models, and interpolation between optimized and target prompts. The training process is complex and slow. These text-based methods require 30 to 70 minutes to fine-tune their models, which is not scalable for real applications. SuTI [7] proposes an in-context learning method for subject-driven tasks. However, SuTI demands half a million expert models for each different subject, making it prohibitively expensive. Although SuTI can perform in-context learning during inference, the setup of expert models remains costly. DreamBooth [23] provides a simpler method for handling subject-driven tasks. Nevertheless, DreamBooth requires approximately 1000 negative samples and 1000 gradient steps, and also needs fine-tuning of the text encoder to achieve state-of-the-art performance. Therefore, it is worthwhile to explore more efficient training methods: the setup should be as simple as possible. First, training should not include multiple optimization phases. Second, text-to-image alignment should fine-tune as few components as possible such as the UNet, but not the text encoder for each prompt. Third, faster evaluation and regularization should be enabled by early stopping based on model selection.

In this paper, we propose a $\lambda$-Harmonic reward function that enables early stopping and accelerates training. In addition, we incorporate the Bradley-Terry preference model to generate preference labels. We utilize preference-based reinforcement learning algorithms to finetune pre-trained diffusion models and to achieve text-to-image alignment without optimizing any text encoder or text embedding. The whole finetuning process including setup, training, validation, and model saving only takes 5 to 20 minutes on Cloud TPU V4. Our method, Reward Preference Optimization (RPO), only requires a

2

few input reference images and the finetuned diffusion model can generate images that preserve the identity of a specific subject while aligning well with textual prompts (Figure 1).

To show the effectiveness of our $\lambda$-Harmonic reward function, we evaluate RPO on diverse subjects and text prompts on DreamBench [23] and we report the DINO and CLIP-I/CLIP-T of RPO's generated images on this benchmark and compare them with existing methods. Surprisingly, our method requires a simple setup (3% of DreamBooth configuration) and with fewer gradient steps, but the experimental results outperform or match SOTA.

In summary, our contributions are as follows:

- We introduce the $\lambda$-Harmonic reward function, which permits early-stopping to alleviate overfitting in subject-driven generation tasks and to accelerate the finetuning process.
- By combining the $\lambda$-Harmonic reward function and a preference model, we present RPO, which only requires a cheap setup, but still can provide high quality results.
- We evaluate RPO and show the effectiveness of the $\lambda$-Harmonic function with diverse subjects and various prompts on DreamBench. We achieve results comparable to SOTA.

## 2 Related Works

Ruiz et al. [23] formulated a class of problems called *subject-driven generation*, which refers to preserving the appearance of a subject contextualized in different settings. DreamBooth [23] solves the issue of preserving the subject by binding it in textual space with a unique identifier for the subject in the reference images, and simultaneously generating diverse backgrounds by leveraging prior class-specific information previously learned. A related work that could possibly perform the same task is textual inversion [12]. However, its original objective is to produce a modification of the subject or property marked by a unique token in the text. While it can be used to preserve the subject and change the background or setting, the performance is underwhelming compared to DreamBooth in various metrics [23].

The prevalent issue in DreamBooth and textual inversion is the long training time [23, 12] since gradient-based optimization has to be performed on their respective models for each subject. Subject-driven text-to-image generator (SuTI) by [7] aims to alleviate this issue by employing apprenticeship learning. By scraping millions of images online, many expert models are trained for different clusters of images centered around different subjects, which allows an apprentice to learn quickly from the experts [7]. However, this is an incredibly resource intensive task with massive computational overhead during training.

In the field of natural language processing, direct preference optimization has found great success in large language models (LLM) [21]. By bypassing reinforcement learning from human feedback and directly maximizing likelihoods using preference data, LLMs benefit from more stable training and reduced dependency on an external reward model. Subsequently, this inspired Diffusion-DPO by [30], which applies a similar technique to diffusion models. However, this relies on a preference labelled dataset, which can be expensive to collect or not publicly available for legal reasons.

Fortunately, there are reward models that can serve as functional substitutes such as CLIP [19] and ALIGN [14]. ALIGN has a dual encoder architecture that was trained on a large dataset. The encoders can produce text and image embeddings, which allows us to obtain pairwise similarity scores by computing cosine similarity. There are also diffusion modelling techniques that can leverage reward models. An example is denoising diffusion policy optimization (DDPO) by Black et al. [3] that uses a policy gradient reinforcement learning method to encourage generations that lead to higher rewards.

## 3 Preliminary

In this section, we introduce the notation and some key concepts about text-to-image diffusion models and reinforcement learning.

**Text-to-Image Diffusion Models.** Diffusion models [13, 26, 28, 29, 27] are a family of latent variable models of the form $p_{\boldsymbol{\theta}}(\mathbf{x}_0) = \int_{\mathcal{X}} p_{\boldsymbol{\theta}}(\mathbf{x}_{0:T}) d\mathbf{x}_{1:T}$, where the $\mathbf{x}_1, \ldots, \mathbf{x}_T$ are noised latent variables of the same dimensionality as the input data $\mathbf{x}_0 \sim q(\mathbf{x}_0)$. The diffusion or forward process

3

is often a Markov chain that gradually adds Gaussian noise to the input data and each intermediate sample $\mathbf{x}_t$ can be written as

$$\mathbf{x}_t = \sqrt{\alpha_t}\mathbf{x}_0 + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_t, \quad \text{for all } t \in \{1, \ldots, T\}, \tag{1}$$

where $\alpha_t$ refers to the variance schedule and $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Given a conditioning tensor $\mathbf{c}$ (often a text embedding), the core premise of text-to-image diffusion models is to use a neural network $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t, \mathbf{c}, t)$ that iteratively refines the current noised sample $\mathbf{x}_t$ to obtain the previous step sample $\mathbf{x}_{t-1}$, This network can be trained by optimizing a simple denoising objective function, which is the time coefficient weighted mean squared error, the derivation is shown in Appendix A.1:

$$\mathbb{E}_{\mathbf{x}_0, \mathbf{c}, t, \boldsymbol{\epsilon}_t} \left[ \omega(t) \| \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t, \mathbf{c}, t) - \boldsymbol{\epsilon}_t \|_2^2 \right], \tag{2}$$

where $t$ is uniformly sampled from $\{1, \ldots, T\}$ and $\omega(t)$ is a time dependent weight that can be simplified to 1 according to [13, 27, 22].

**Reinforcement Learning and Diffusion DPO**    Reinforcement Learning for diffusion models [3, 11, 30] aims to solve the following optimization problem:

$$\mathbb{E}_{\mathbf{x}_{0:T} \sim p_{\boldsymbol{\theta}}(\mathbf{x}_{0:T}|\mathbf{c})} \left[ \sum_{t=1}^{T} R(\mathbf{x}_t, \mathbf{x}_{t-1}, \mathbf{c}) - \beta \mathbb{D}_{\text{KL}}(p_{\boldsymbol{\theta}}(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{c}) \| p_{\text{base}}(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{c})) \right], \tag{3}$$

where $\beta$ is a hyperparameter controlling the KL-divergence between the finetuned model $p_{\boldsymbol{\theta}}$ and the pre-trained base model $p_{\text{base}}$. In Equation (14) from Diffusion-DPO [30], the optimal $p_{\boldsymbol{\theta}}$ can be approximated by minimizing the negative log-likelihood:

$$\mathbb{E}_{\mathbf{x}_0^+, \mathbf{x}_0^-, t, \mathbf{x}_t^+, \mathbf{x}_t^-} \left[ -\log \sigma \left( \beta \left( \| \boldsymbol{\epsilon}_{\text{base}}(\mathbf{x}_t^+, \mathbf{c}, t) - \boldsymbol{\epsilon}_t^+ \|_2^2 - \| \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t^+, \mathbf{c}, t) - \boldsymbol{\epsilon}^+ \|_2^2 \right. \right. \right.$$
$$\left. \left. \left. - (\| \boldsymbol{\epsilon}_{\text{base}}(\mathbf{x}_t^-, \mathbf{c}, t) - \boldsymbol{\epsilon}_t^- \|_2^2 - \| \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t^-, \mathbf{c}, t) - \boldsymbol{\epsilon}^- \|_2^2) \right) \right) \right], \tag{4}$$

where $\boldsymbol{\epsilon}^+$ and $\boldsymbol{\epsilon}^-$ are independent samples from a Gaussian distribution, $\mathbf{x}_t^+$ and $\mathbf{x}_t^-$ are perturbed versions of $\mathbf{x}_0^+$ and $\mathbf{x}_0^-$ that depend on $\boldsymbol{\epsilon}^+$ and $\boldsymbol{\epsilon}^-$, and $\mathbf{x}_0^+$ is preferred to $\mathbf{x}_0^-$. A detailed description is given in Appendix A.1.

**Additional notation.**    We use $\mathbf{x}_{\text{ref}}$ and $\mathbf{x}_{\text{gen}}$ to represent the reference image and generated image, respectively. $\mathcal{I}_{\text{ref}}$ denotes the set of reference images, and $\mathcal{I}_{\text{gen}}$ is the set of generated images. $\mathbb{P}(\mathbf{x} \succ \tilde{\mathbf{x}})$ represents the probability that $\mathbf{x}$ is more preferred than $\tilde{\mathbf{x}}$.

# 4   Method

We present our $\lambda$-Harmonic reward function that provides reward signals for subject-driven tasks to reduce the risk that the learned model will overfit to the reference images. Based on this reward function, we use the Bradley-Terry model to sample preference labels and a preference algorithm to finetune the diffusion model by optimizing both a similarity loss and a preference loss.

## 4.1   Reward Preference Optimization

In contrast to other fine-tuning applications [30, 18, 21, 20], there is no human feedback in the subject-driven text-to-image generation task. The model only receives a few reference images and a prompt with a specific subject. Hence, we first propose the $\lambda$-Harmonic reward function that can leverage the ALIGN model [14] to provide feedback based on the generated image fidelity: similarity to the given reference images and faithfulness to the text prompts.

$\lambda$**-Harmonic Reward Function.**    The normalized ALIGN-I and ALIGN-T scores can be denoted as

$$\text{ALIGN-I}(\mathbf{x}, \mathcal{I}_{\text{ref}}) := \frac{1}{|\mathcal{I}_{\text{ref}}|} \sum_{\tilde{\mathbf{x}} \in \mathcal{I}_{\text{ref}}} \frac{\text{CosSim}(f_\phi(\mathbf{x}), f_\phi(\tilde{\mathbf{x}})) + 1}{2} \qquad \text{(Image alignment)}$$

$$\text{ALIGN-T}(\mathbf{x}, \mathbf{c}) := \frac{\text{CosSim}(f_\phi(\mathbf{x}), g_\phi(\mathbf{c})) + 1}{2} \qquad \text{(Text alignment)},$$

4

where CosSim is the cosine similarity, $f_\phi(\mathbf{x})$ is the image feature extractor and $g_\phi(\mathbf{c})$ is the text encoder in the ALIGN model. Given a reference image set $\mathcal{I}_{\text{ref}}$, the $\lambda$-Harmonic reward function can be defined by a weighted harmonic mean of the ALIGN-I and ALIGN-T scores,

$$r(\mathbf{x}, \mathbf{c}; \lambda, \mathcal{I}_{\text{ref}}) := \frac{1}{\dfrac{\lambda}{\text{ALIGN-I}(\mathbf{x}, \mathcal{I}_{\text{ref}})} + \dfrac{1 - \lambda}{\text{ALIGN-T}(\mathbf{x}, \mathbf{c})}}. \qquad (5)$$

Compared to the arithmetic mean, there are two advantages to using the harmonic mean: (1) according to AM-GM-HM inequalities [10], the harmonic mean is a lower bound of the arithmetic mean and maximizing this "pessimistic" reward can also improve the arithmetic mean of ALIGN-I and ALIGN-T scores; (2) the harmonic mean is more sensitive to the smaller of the two scores, i.e., a larger reward is only achieved when both scores are relatively large.

For a simple example, consider $\lambda = 0.5$. If there are two images, $\mathbf{x}$ and $\tilde{\mathbf{x}}$, where the first image achieves an ALIGN-I score of 0.9 and an ALIGN-T score of 0.01, and the second image receives an ALIGN-I score of 0.7 and an ALIGN-T score of 0.21, we may prefer the second image because it has high similarity to the reference images and is faithful to the text prompts. However, using the arithmetic mean would assign both images the same reward of 0.455. In contrast, the harmonic mean would assign the first image a reward of 0.020 and the second image a reward of 0.323, aligning with our preferences. During training, we set $\lambda_{\text{train}} = 0$, which means the reward model will focus solely on text-to-image alignment because the objective function consists only of a loss for image-to-image alignment. Note that we set $\lambda_{\text{val}}$ to a different value for validation, which evaluates the fidelity of the subject and faithfulness of the prompt. Details can be found in Section 5.

**Dataset.** The set of images for subject-driven generative tasks can usually be represented as $\mathcal{I} = \mathcal{I}_{\text{ref}} \cup \mathcal{I}_{\text{gen}}$, where $\mathcal{I}_{\text{gen}}$ is the image set generated by the base model. DreamBooth [23] requires two different prompts, $\mathbf{c}$ and $\mathbf{c}_{\text{pr}}$, where $\mathbf{c}$ includes a reference to the subject while $\mathbf{c}_{\text{pr}}$ refers to the prior class of the subject but not the subject. For example, $\mathbf{c}$ can be ``a photo of [V] dog'' and $\mathbf{c}_{\text{pr}}$ can be ``a photo of a dog'', where ``[V]'' is a unique token that refers to the subject and dog is the prior class of the subject. DreamBooth then uses $\mathbf{c}_{\text{pr}}$ to generate a variety of images $\mathcal{I}_{\text{gen}}$ in the prior class to avoid overfitting to the reference images via a regularizer. Typically, the size of the set of generated images is around 1000, i.e., $|\mathcal{I}_{\text{gen}}| = 1000$, which is time-consuming and space-intensive in real applications. However, the diffusion model can only maximize the similarity score and still receives a high reward based on this uninformative prompt $\mathbf{c}_{\text{pr}}$. Our method aims to balance the trade-off between similarity and faithfulness. Thus, for efficiency, we introduce 8 novel training prompts, $\mathbf{c}_{\text{mod}}$ of the form ``a [V] [class noun] [modification]'' where modification includes *artistic style transfer, re-contextualization*, and *accessorization*. For example, $\mathbf{c}_{\text{mod}}$ can be ``a [V] dog is on the Moon''. These training prompts can be pre-specified or generated by other Large Language Models[2]. The full list of training prompts is provided in the supplementary material (Figure 6). We feed these training prompts to the base model and generate 4 images for each training prompt, i.e., $|\mathcal{I}_{\text{gen}}| = 32$.

Once we obtain reward signals, we adopt the Bradley-Terry model [4] to generate preference labels. In particular, given a tuple $(\mathbf{x}_{\text{ref}}, \mathbf{x}_{\text{gen}}, \mathbf{c}_{\text{mod}})$, we sample preference labels $y$ from the following probability model:

$$\mathbb{P}(\mathbf{x}_{\text{ref}} \succ \mathbf{x}_{\text{gen}}) := \frac{\exp(r(\mathbf{x}_{\text{ref}}, \mathbf{c}_{\text{mod}}; \lambda, \mathcal{I}_{\text{ref}}))}{\exp(r(\mathbf{x}_{\text{ref}}, \mathbf{c}_{\text{mod}}; \lambda, \mathcal{I}_{\text{ref}})) + \exp(r(\mathbf{x}_{\text{gen}}, \mathbf{c}_{\text{mod}}; \lambda, \mathcal{I}_{\text{ref}}))}. \qquad (6)$$

**Learning.** The learning objective function consists of two parts — similarity loss and preference loss. The similarity loss is designed to minimize the KL divergence between the distribution of reference images and the learned distribution $p_{\boldsymbol{\theta}}(\mathbf{x})$, which is equivalent to minimizing:

$$\mathcal{L}_{\text{sim}}(\boldsymbol{\theta}) := \mathbb{E}_{\mathbf{x}_{\text{ref}}, \mathbf{c}, t, \boldsymbol{\epsilon}_{\text{ref}}} \left[ \|\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_{\text{ref},t}, \mathbf{c}, t) - \boldsymbol{\epsilon}_{\text{ref}}\|_2^2 \right], \quad t \sim \mathcal{U}\{1, \ldots, T\}, \quad \boldsymbol{\epsilon}_{\text{ref}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \quad (7)$$

The preference loss aims to capture the preference signals and fit the preference model, Eq. (6). Therefore, we use binary cross-entropy as the objective function for the preference loss. Combining

---

[2]SuTI [7] utilizes PaLM [9] to generate unseen prompts during training

5

**Reference Images**

Similar Loss

Diffusion Models

Preference Loss

$\lambda$-Harmonic Reward

"A [V] dog is on the Moon"

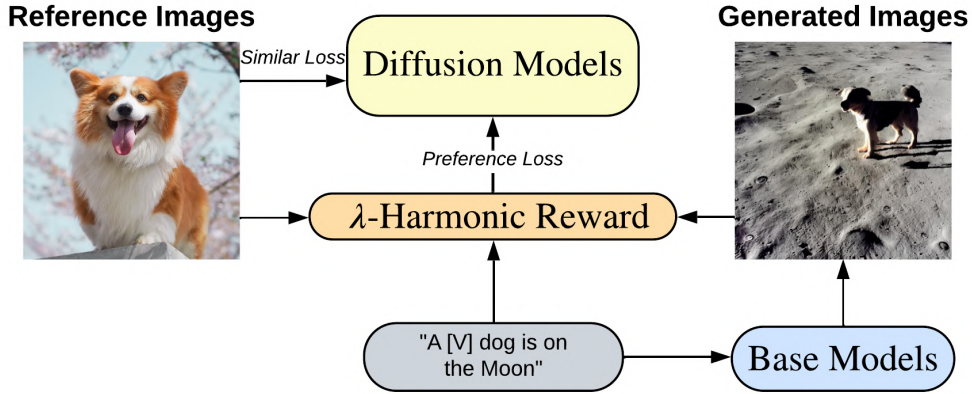
Base Models

**Generated Images**

Figure 2: Overview of the finetuning phase for RPO. First, the base diffusion model generates a few images based on novel training prompts. Second, we compute the rewards for both reference and generated images using Equation (5). Then, preference labels are sampled according to the preference distribution, as defined in Equation (6). Finally, the diffusion model is trained by minimizing both the similarity loss (Equation (7)) and preference loss (Equation (8)).

the DPO objective function in Eq. 4, the loss function can be written as follows:

$$\mathcal{L}_{\text{pref}}(\boldsymbol{\theta}) := \mathbb{E}_{\mathbf{x}_{\text{ref}},\mathbf{x}_{\text{gen}},\mathbf{c}_{\text{mod}},y,t,\boldsymbol{\epsilon}_{\text{ref}},\boldsymbol{\epsilon}_{\text{gen}}} \left[ y \log \sigma \left( \beta \ell_{\boldsymbol{\theta}}(\mathbf{x}_{\text{ref}},\mathbf{x}_{\text{gen}},\mathbf{c}_{\text{mod}},y,t,\boldsymbol{\epsilon}_{\text{ref}},\boldsymbol{\epsilon}_{\text{gen}}) \right) \right.$$

$$\left. + (1-y) \log \sigma \left( -\beta \ell_{\boldsymbol{\theta}}(\mathbf{x}_{\text{ref}},\mathbf{x}_{\text{gen}},\mathbf{c}_{\text{mod}},y,t,\boldsymbol{\epsilon}_{\text{ref}},\boldsymbol{\epsilon}_{\text{gen}}) \right) \right], \tag{8}$$

where

$$\ell_{\boldsymbol{\theta}}(\mathbf{x}_{\text{ref}},\mathbf{x}_{\text{gen}},\ \mathbf{c}_{\text{mod}},y,t,\boldsymbol{\epsilon}_{\text{ref}},\boldsymbol{\epsilon}_{\text{gen}}) := \|\boldsymbol{\epsilon}_{\text{base}}(\mathbf{x}_{\text{ref},t},\mathbf{c}_{\text{mod}},t) - \boldsymbol{\epsilon}_{\text{ref}}\|_2^2 - \|\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_{\text{ref},t},\mathbf{c}_{\text{mod}},t) - \boldsymbol{\epsilon}_{\text{ref}}\|_2^2$$

$$- (\|\boldsymbol{\epsilon}_{\text{base}}(\mathbf{x}_{\text{gen},t},\mathbf{c}_{\text{mod}},t) - \boldsymbol{\epsilon}_{\text{gen}}\|_2^2 - \|\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_{\text{gen},t},\mathbf{c}_{\text{mod}},t) - \boldsymbol{\epsilon}_{\text{gen}}\|_2^2),$$

and $t \sim \mathcal{U}\{1,\ldots,T\}$ and $\boldsymbol{\epsilon}_{\text{ref}}, \boldsymbol{\epsilon}_{\text{gen}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Combining these two loss functions together, the objective function for finetuning is written as

$$\mathcal{L}(\boldsymbol{\theta}) = \mathcal{L}_{\text{sim}}(\boldsymbol{\theta}) + \mathcal{L}_{\text{pref}}(\boldsymbol{\theta}) \tag{9}$$

Figure 2 presents an overview of the training method, which includes the base model generated samples, the ALIGN reward model, and the preference loss. Note that $\mathcal{L}_{\text{pref}}$ serves as a regularizer for approximating the text-to-image alignment policy. Conversely, DreamBooth [23] adopts $\mathbb{D}_{\text{KL}}(p_{\text{base}}(\mathbf{x}_{t-1} \mid \mathbf{x}_t,\mathbf{c})|p_{\boldsymbol{\theta}}(\mathbf{x}_{t-1} \mid \mathbf{x}_t,\mathbf{c}))$ as its regularizer, which cannot guarantee faithfulness to the text-prompt. Based on this loss function and preference model, we only need a few hundred gradient steps and a small set size of $|\mathcal{I}_{\text{gen}}|$ to achieve results that are comparable to, or even better than, the state of the art. The fine-tuning process, which includes generating images, training, and validation, takes about 5 to 20 minutes on a single Google Cloud Platform TPUv4-8 (32GB) for Stable Diffusion.

## 5  Experiments

In this section, we present the experimental results demonstrated by RPO. We investigate several questions. First, can our algorithm learn to generate images that are faithful both to the preference images and to the textual prompts, according to preference labels? Second, if RPO can generate high-quality images, which part is the key component of RPO: the reference loss or the early stopping by the $\lambda$-Harmonic reward function? Third, how do different $\lambda_{\text{val}}$ values used during validation affect performance in RPO? We refer readers to Appendix A.2 for details on the experimental setup, Appendix A.7 for the skill set of RPO, Appendix A.4 for the limitations of the RPO algorithm, and Appendix A.5 for future work involving RPO.

6

## 5.1 Dataset and Evaluation

**DreamBench.** In this work, we use the DreamBench dataset proposed by DreamBooth [23]. This dataset contains 30 different subject images including backpacks, sneakers, boots, cats, dogs, and toy, etc. DreamBench also provides 25 various prompt templates for each subject and these prompts are requiring the learned models to have the following abilities: re-contextualization, accessorization, property modification, and attribute editing.

**Evaluation Metrics.** We follow DreamBooth [23] and SuTI [7] to report DINO [5][3] and CLIP-I [19] for evaluating image-to-image similarity score and CLIP-T [19] for evaluating the text-to-image similarity score. We also use our $\lambda$-Harmonic reward as a evaluation metric for the overall fidelity and the default value of $\lambda = 0.3$. For evaluation, we follow DreamBooth [23] and SuTI [7] to generate 4 images per prompt, 3000 images in total, which provides a robust evaluation.

**Baseline algorithms.** DreamBooth [23]: A test-time fine-tuning method. This algorithm requires approximately $|\mathcal{I}_{gen}| = 1000$ and 1000 gradient steps to finetune the UNet and text-encoder components. SuTI [7]: A pre-trained method that requires half a million expert models and introduces cross-attention layers into the original diffusion models. Textual Inversion [12]: A text-based method that optimizes the text embedding but freezes the diffusion models. Re-Imagen [8]: An information retrieval-based algorithm that modifies the backbone network architectures and introduces cross-attention layers into the original diffusion models. DisenBooth [6]: A test-time fine-tuning method that generates subject-driven images by optimizing textual identity-preserving embeddings. Custom Diffusion [16]: a test-time fine-tuning method, focused on efficient training by optimizing the key and value projection matrices in the cross-attention layers of diffusion models. ELITE [31]: A test-time fine-tuning approach consisting of two stages: one for training the textual embeddings and another for preserving identity. IP-Adapter [32]: a pretrained method where the decoupled cross-attention layer captures the textual signal while integrating reference images. SSR-Encoder [33]: also a pretrained method that highlights selective regions and extracts detailed features for subject-driven image generation.

## 5.2 Results

**Quantitative Results.** We begin by addressing the first question. We use a quantitative evaluation to compare RPO with other existing methods on three metrics (DINO, CLIP-I, CLIP-T) in DreamBench to validate the effectiveness of RPO. The experimental results on DreamBench is shown in Table 1. We observe that RPO can perform better or on par with existing works on all three metrics. Compared to DreamBooth, RPO only requires 3% of the negative samples, but RPO can outperform DreamBooth on the CLIP-I and CLIP-T scores by 3% given the same backbone. Our method outperforms all baseline algorithms in the CLIP-T score, establishing a new SOTA result. This demonstrates that RPO, by solely optimizing UNet through preference labels from the $\lambda$-Harmonic reward function, can generate images that are faithful to the input prompts. Similarly, our CLIP-I score is also the highest, which indicates that RPO can generate images that preserve the subject's visual features. In terms of the DINO score, our method is almost the same as DreamBooth when using the same backbone. We conjecture that the reason RPO achieves higher CLIP scores and lower DINO score is that the $\lambda$-Harmonic reward function prefers to select images that are semantically similar to the textual prompt, which may result in the loss of some unique features in the pixel space.

**Qualitative Results.** We use the same prompt as SuTI [7], and the generated images are shown in Figure 3. RPO generates images that are faithful to both reference images and textual prompts. We noticed a semantic mistake in the first prompt used by SuTI [7]; it should be `A dog eating a cherry from a bowl`. Furthermore, each reference bowl image contains blueberries, and the ambiguous prompt caused the RPO-trained model to become confused during the inference phase. However, RPO still preserves the unique appearance of the bowl. For instance, while the text on the bowl is incorrect or blurred in the SuTI and DreamBooth results, RPO accurately retains the words *Bon Appetit* from the reference bowl images. Although existing methods can produce images highly faithful to the reference images, they may not align as well with the textual prompts. We also

---

[3]DINO encodes only images and makes a better fine-grained understanding of images than CLIP. But it cannot provide signals on text-image alignment.

7

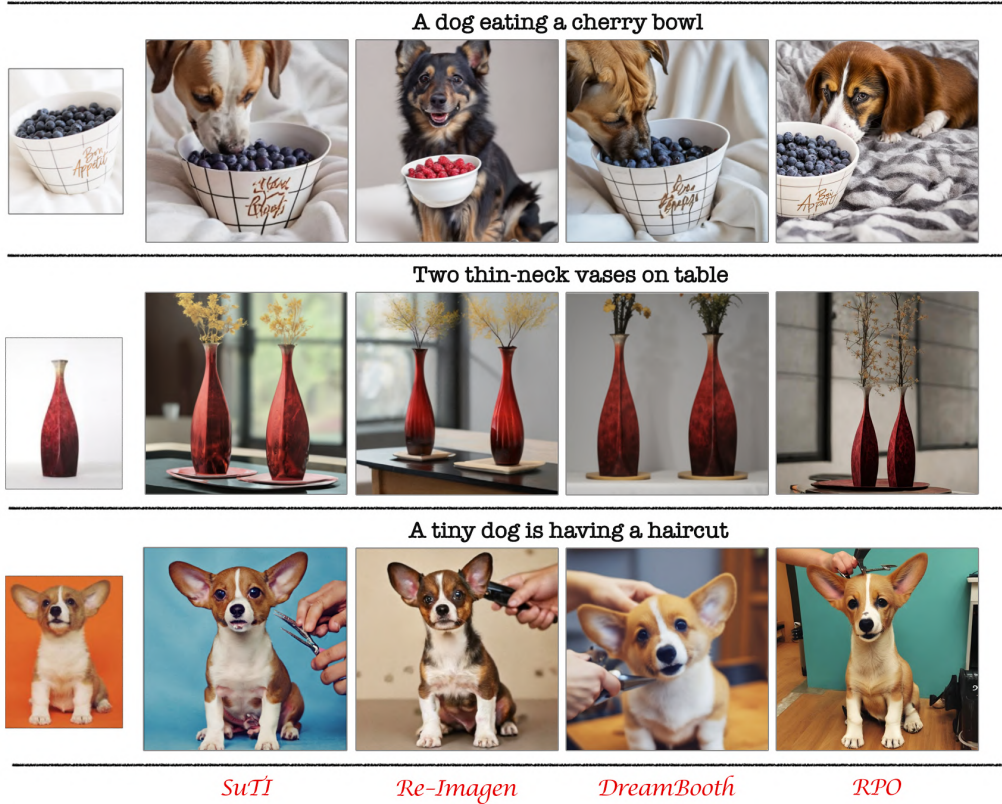



Figure 3: Qualitative comparison with other subject-driven text-to-image methods, adapted from [7]

Table 1: Quantitative comparison for the number of iterations, subject fidelity and prompt fidelity.

| Method | Backbone | Iterations ↓ | DINO ↑ | CLIP-I ↑ | CLIP-T ↑ |
|---|---|---|---|---|---|
| Reference Images | N/A | N/A | 0.774 | 0.885 | N/A |
| DreamBooth [23] | Imagen [24] | 1000 | 0.696 | 0.812 | 0.306 |
| DreamBooth [23] | SD [22] | 1000 | 0.668 | 0.803 | 0.305 |
| Textual inversion [12] | SD [22] | 5000 | 0.569 | 0.780 | 0.255 |
| SuTI [7] | Imagen [24] | $1.5 \times 10^5$ | **0.741** | 0.819 | 0.304 |
| Re-Imagen [8] | Imagen [24] | $2 \times 10^5$ | 0.600 | 0.740 | 0.270 |
| DisenBooth [6] | SD[22] | 3000 | 0.574 | 0.755 | 0.255 |
| Custom Diffusion [16] | SD[22] | 500 | 0.695 | 0.801 | 0.245 |
| ELETE [31] | SD[22] | 3000 | 0.652 | 0.765 | 0.255 |
| IP-Adapter [32] | SD[22] | $10^6$ | 0.608 | 0.809 | 0.274 |
| SSR-Encoder [33] | SD[22] | $10^6$ | 0.612 | 0.821 | **0.314** |
| Ours: RPO | SD [22] | **400** | 0.652 | **0.833** | **0.314** |

8

Table 2: Ablation study on regularization to evaluate fidelity across multiple subjects and prompts. Standard deviation is included.

| Method | DINO ↑ | CLIP-I ↑ | CLIP-T ↑ | 0.3-Harmonic ↑ |
|---|---|---|---|---|
| Pure $\mathcal{L}_{\text{sim}}$ | **0.695 ± 0.077** | **0.852 ± 0.043** | 0.285 ± 0.027 | 0.660 ± 0.016 |
| $\mathcal{L}_{\text{pref}}$ w/o early-stopping | 0.688 ± 0.082 | 0.845 ± 0.042 | 0.296 ± 0.027 | 0.663 ± 0.014 |
| Early-stopping w/o $\mathcal{L}_{\text{pref}}$ | 0.575 ± 0.124 | 0.799 ± 0.052 | 0.323 ± 0.025 | 0.672 ± 0.013 |
| RPO ($\lambda_{\text{val}} = 0.3$) | 0.581 ± 0.113 | 0.798 ± 0.039 | **0.329 ± 0.021** | **0.673 ± 0.013** |

Table 3: Different validation $\lambda_{\text{val}}$-Harmonic reward comparison for evaluating fidelity over multiple subjects and prompts. Standard deviation is included.

| Configuration | DINO ↑ | CLIP-I ↑ | CLIP-T ↑ | 0.3-Harmonic ↑ |
|---|---|---|---|---|
| $\lambda_{\text{val}} = 0.3$ | 0.581 ± 0.113 | 0.798 ± 0.039 | **0.329 ± 0.021** | **0.673 ± 0.013** |
| $\lambda_{\text{val}} = 0.5$ | 0.652 ± 0.082 | 0.833 ± 0.041 | 0.314 ± 0.022 | 0.671 ± 0.008 |
| $\lambda_{\text{val}} = 0.7$ | **0.679 ± 0.085** | **0.850 ± 0.045** | 0.304 ± 0.023 | 0.667 ± 0.011 |

provide an example in Appendix A.3 that shows how RPO can handle the failure case observed in DreamBooth and SuTI. Additionally, we find that the RPO model can generate reasonable images even for highly imaginative prompts, as shown in Appendix A.7 (Figure 16 and Figure 17). These images demonstrate that the RPO-trained model does not overfit the training data. Instead, the $\lambda$-Harmonic function provides a method for selecting a model capable of text-to-image alignment, even when the textual prompts are highly imaginative.

## 5.3 Ablation Study and Method Analysis

**Preference Loss and $\lambda$-Harmonic Ablation.** We investigate the second question through an ablation study. Two regularization components are introduced into RPO: reference loss as a regularizer and early stopping by $\lambda_{\text{val}}$-Harmonic reward function. Consequently, we compare four methods: (1) Pure $\mathcal{L}_{\text{sim}}$, which only minimizes the image-to-image similarity loss $\mathcal{L}_{\text{sim}}$; (2) $\mathcal{L}_{\text{pref}}$ w/o early stopping, which employs $\mathcal{L}_{\text{pref}}$ as a regularizer but omits early stopping by $\lambda_{\text{val}}$-Harmonic reward function; (3) Early stopping w/o $\mathcal{L}_{\text{pref}}$, which uses $\lambda_{\text{val}}$-Harmonic reward function as a regularization method but excludes $\mathcal{L}_{\text{pref}}$; (4) Full RPO, which utilizes both $\mathcal{L}_{\text{pref}}$ and early stopping by the $\lambda_{\text{val}}$-Harmonic reward function. We choose the default value $\lambda_{\text{val}} = 0.3$ in this ablation study.

Table 2 lists the evaluation results of these four methods on DreamBench. We observe that without early stopping, $\mathcal{L}_{\text{pref}}$ can still prevent overfitting to the reference images and improve text-to-image alignment, though the regularization effect is weak. Specifically, the 0.3-Harmonic only shows a

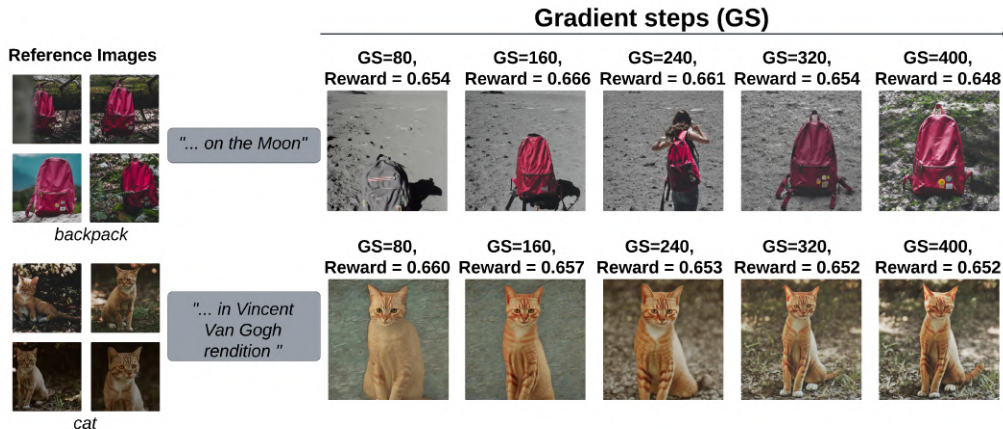



Figure 4: Changes in the $0.3$-Harmonic reward value during RPO training process.

9

Figure 5: Different $\lambda_{\mathrm{val}}$'s will lead to different results. A small $\lambda_{\mathrm{val}}$ assigns a higher a weight for text-to-image alignment and leads to diverse generation. A large $\lambda_{\mathrm{val}}$ may also cause overfitting.

marginal improvement of 0.003 over pure $\mathcal{L}_{\mathrm{sim}}$ and 0.001 over early stopping without $\mathcal{L}_{\mathrm{pref}}$. The early stopping facilitated by the $\lambda_{\mathrm{val}}$-Harmonic reward function plays a crucial role in counteracting overfitting, helping the diffusion models retain the ability to generate high-quality images aligned with textual prompts. To provide a deeper understanding of the $\lambda$-Harmonic reward validation, we present two examples from during training in Figure 4, covering both objects and live subjects. We found that the model tends to overfit at a very early stage, i.e., within 200 gradient steps, where $\lambda$-Harmonic can provide correct reward signals for the generated images. For the backpack subject, the generated image receives a low reward at gradient step 80 due to its lack of fidelity to the reference images. However, at gradient step 400, the image is overfitted to the reference images, and the model fails to align well with the input text, resulting in another low reward. $\lambda$-Harmonic assigns a high reward to images that are faithful to both the reference image and textual prompts.

**Impact of $\lambda_{\mathrm{val}}$.** We examine the third question by selecting different $\lambda_{\mathrm{val}}$ values from the set $\{0.3, 0.5, 0.7\}$ as the validation parameters for the $\lambda$-Harmonic reward. According to Equation 5, we believe that as $\lambda_{\mathrm{val}}$ increases, the $\lambda$-Harmonic reward function will give higher weight to the image-to-image similarity score. This will make the generated images more closely resemble the reference images, however, there is also a risk of overfitting. Table 3 shows the results of three different $\lambda_{\mathrm{val}}$ values on DreamBench. As we expected, a larger $\lambda_{\mathrm{val}}$ makes the images better preserve the characteristics of the reference images, but it also reduces the text-to-image alignment score. Figure 5 shows us an example. In this example, different $\lambda_{\mathrm{val}}$ values lead to different outcomes due to varying strengths of regularization. A smaller $\lambda_{\mathrm{val}} = 0.3$ can generate more varied results, but seems somewhat off from the reference images. $\lambda_{\mathrm{val}} = 0.5$ preserves some characteristics beyond the original subject, such as the sofa, but also maintains alignment between text and image. However, when $\lambda_{\mathrm{val}} = 0.7$ is chosen as an excessively large value, the model actually overfits to the reference images, ignoring the prompts. We have additional comparisons, including different $\lambda_{\mathrm{train}}$'s, different Pythagorean means, and aesthetic scores [25] in Appendix A.3.

## 6 Conclusion

We introduce the $\lambda$-Harmonic reward function to derive preference labels and employ RPO to finetune the diffusion model for subject-driven text-to-image generation tasks. Additionally, the $\lambda$-Harmonic reward function serves as a validation method, enabling early stopping to mitigate overfitting to reference images and speeding up the finetuning process.


## Acknowledgments and Disclosure of Funding

We thank Shixin Luo and Hongliang Fei for providing constructive feedback. This work was supported by a Google grant with Cloud TPUs from Google's TPU Research Cloud (TRC). We also thank the Vector Institute, the Canada CIFAR AI Chair program and the Natural Sciences and Engineering Research Council of Canada for their support.



## References

[1] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless



10



assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

[2] Charlotte Bird, Eddie Ungless, and Atoosa Kasirzadeh. Typology of risks of generative text-to-image models. In *Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society*, pages 396–410, 2023.

[3] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301*, 2023.

[4] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[5] 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*, pages 9650–9660, 2021.

[6] Hong Chen, Yipeng Zhang, Simin Wu, Xin Wang, Xuguang Duan, Yuwei Zhou, and Wenwu Zhu. Disenbooth: Identity-preserving disentangled tuning for subject-driven text-to-image generation. *arXiv preprint arXiv:2305.03374*, 2023.

[7] 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. *Advances in Neural Information Processing Systems*, 36, 2024.

[8] Wenhu Chen, Hexiang Hu, Chitwan Saharia, and William W Cohen. Re-imagen: Retrieval-augmented text-to-image generator. *arXiv preprint arXiv:2209.14491*, 2022.

[9] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

[10] Dušan Djukić, Vladimir Janković, Ivan Matić, and Nikola Petrović. *The IMO compendium: a collection of problems suggested for the International Mathematical Olympiads: 1959-2004*, volume 119. Springer, 2006.

[11] Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. Reinforcement learning for fine-tuning text-to-image diffusion models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[12] 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. *arXiv preprint arXiv:2208.01618*, 2022.

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

[14] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021.

[15] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6007–6017, 2023.

[16] 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*, pages 1931–1941, 2023.



11



[17] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[18] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[19] 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*, pages 8748–8763. PMLR, 2021.

[20] Rafael Rafailov, Joey Hejna, Ryan Park, and Chelsea Finn. From $r$ to $q^*$: Your language model is secretly a q-function. *arXiv preprint arXiv:2404.12358*, 2024.

[21] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024.

[22] 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*, pages 10684–10695, 2022.

[23] 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*, pages 22500–22510, 2023.

[24] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

[25] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022.

[26] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.

[27] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021.

[28] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

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

[30] Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. *arXiv preprint arXiv:2311.12908*, 2023.

[31] Yuxiang Wei, Yabo Zhang, Zhilong Ji, Jinfeng Bai, Lei Zhang, and Wangmeng Zuo. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15943–15953, 2023.

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



12



[33] Yuxuan Zhang, Yiren Song, Jiaming Liu, Rui Wang, Jinpeng Yu, Hao Tang, Huaxia Li, Xu Tang, Yao Hu, Han Pan, et al. Ssr-encoder: Encoding selective subject representation for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8069–8078, 2024.


# A Appendix

## A.1 Background

**Reinforcement Learning.** In Reinforcement Learning (RL), the environment can be formalized as a Markov Decision Process (MDP). An MDP is defined by a tuple $(\mathcal{S}, \mathcal{A}, P, R, \rho_0, T)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P$ is the transition function, $R$ is the reward function, $\rho_0$ is the distribution over initial states, and $T$ is the time horizon. At each timestep $t$, the agent observes a state $\mathbf{s}_t$ and selects an action $\mathbf{a}_t$ according to a policy $\pi(\mathbf{a}_t|\mathbf{s}_t)$, and obtains a reward $R(\mathbf{s}_t, \mathbf{a}_t)$, and then transit to a next state $\mathbf{s}_{t+1} \sim P(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$. As the agent interacts with the MDP, it produces a sequence of states and actions, which is denoted as a trajectory $\boldsymbol{\tau} = (\mathbf{s}_0, \mathbf{a}_0, \mathbf{s}_1, \mathbf{a}_1, \ldots, \mathbf{s}_{T-1}, \mathbf{a}_{T-1})$. The RL objective is to maximize the expected value of cumulative reward over the trajectories sample from its policy:

$$\mathbb{E}_{\boldsymbol{\tau} \sim p^\pi(\boldsymbol{\tau})} \left[ \sum_{t=0}^{T-1} R(\mathbf{s}_t, \mathbf{a}_t) \right]$$

**Diffusion models.** [26, 13, 29] consider a variance schedule $\{\beta_t\}_{t=1}^T$. For each data point $\mathbf{x}_0 \sim p_{\text{data}}(\mathbf{x}_0)$, a Markov chain is constructed by transition probability $p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1 - \beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I})$. Consequently, $p_{\alpha_t}(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\alpha_t}\mathbf{x}_0, (1 - \alpha_t)\mathbf{I})$, where $\alpha_t = \prod_{t'=1}^t (1 - \beta_t)$. [28, 29] show that training diffusion models, $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$, is equivalent to score matching, i.e., training a score function $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$:

$$\mathbb{E}_{t \sim \mathcal{U}\{1, \ldots, T\}, \mathbf{x}_0 \sim p_{\text{data}}, \mathbf{x}_t \sim p_{\alpha_t}(\mathbf{x}_t|\mathbf{x}_0)} \left[ \|\nabla_{\mathbf{x}_t} \log p_{\alpha_t}(\mathbf{x}_t \mid \mathbf{x}_0) - \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\|_2^2 \right]$$

$$= \mathbb{E}_{t \sim \mathcal{U}\{1, \ldots, T\}, \mathbf{x}_0 \sim p_{\text{data}}, \mathbf{x}_t \sim p_{\alpha_t}(\mathbf{x}_t|\mathbf{x}_0)} \left[ \left\| -\frac{1}{1 - \alpha_t}(\mathbf{x}_t - \sqrt{\alpha_t}\mathbf{x}_0) - \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) \right\|_2^2 \right]$$

$$= \mathbb{E}_{t \sim \mathcal{U}\{1, \ldots, T\}, \mathbf{x}_0 \sim p_{\text{data}}, \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})} \left[ \left\| -\sqrt{1 - \alpha_t}\boldsymbol{\epsilon} - \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) \right\|_2^2 \right] \quad \because \mathbf{x}_t = \sqrt{\alpha_t}\mathbf{x}_0 + \sqrt{1 - \alpha_t}\boldsymbol{\epsilon}$$

$$= \mathbb{E}_{t \sim \mathcal{U}\{1, \ldots, T\}, \mathbf{x}_0 \sim p_{\text{data}}, \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})} \left[ (1 - \alpha_t)\|\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \boldsymbol{\epsilon}\|_2^2 \right] \quad \text{let } \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) := -\frac{\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)}{\sqrt{1 - \alpha_t}}$$

$$= \mathbb{E}_{t \sim \mathcal{U}\{1, \ldots, T\}, \mathbf{x}_0 \sim p_{\text{data}}, \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})} \left[ \omega(t)\|\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \boldsymbol{\epsilon}\|_2^2 \right], \quad \text{where } \omega(t) := 1 - \alpha_t.$$

**Diffusion MDP** We formalize the denoising process as the following *Diffusion MDP*:

$$\mathbf{s}_{T-t} = (\mathbf{x}_t, \mathbf{c}), \quad \mathbf{a}_{T-t} = \mathbf{x}_{t-1}, \quad \pi_{\boldsymbol{\theta}}(\mathbf{a}_{T-t} \mid \mathbf{s}_{T-t}) = p_{\boldsymbol{\theta}}(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{c}),$$

$$\rho_0 = p(\mathbf{c}) \times \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad R(\mathbf{s}_{T-t}, \mathbf{a}_{T-t}) = R(\mathbf{x}_t, \mathbf{x}_{t-1}, \mathbf{c}) = \begin{cases} r(\mathbf{x}_0, \mathbf{c}) & \text{if } t = 1, \\ 0 & \text{otherwise} \end{cases}$$

where $r(\mathbf{x}_0, \mathbf{c})$ can be a reward signal for the denoised image. The transition kernel is deterministic, i.e., $P(\mathbf{s}_{T-t+1} \mid \mathbf{s}_{T-t}, \mathbf{a}_{T-t}) = (\mathbf{a}_{T-t}, \mathbf{c}) = (\mathbf{x}_{t-1}, \mathbf{c})$. For brevity, the trajectory $\boldsymbol{\tau}$ is defined by $(\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_T)$. Hence, the trajectory distribution for given diffusion models can be denoted as the joint distribution $p_{\boldsymbol{\theta}}(\mathbf{x}_{0:T} \mid \mathbf{c})$. In particular, the RL objective function for finetuning diffusion models can be re-written as the following optimization problem:

$$\mathbb{E}_{\mathbf{x}_{0:T} \sim p_{\boldsymbol{\theta}}(\mathbf{x}_{0:T}|\mathbf{c})} \left[ \sum_{t=1}^T R(\mathbf{x}_t, \mathbf{x}_{t-1}, \mathbf{c}) - \beta \mathbb{D}_{\text{KL}}(p_{\boldsymbol{\theta}}(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{c}) \| p_{\text{base}}(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{c})) \right],$$

where $\beta$ is a hyperparameter controlling the KL-divergence between the finetune model $p_{\boldsymbol{\theta}}$ and the base model $p_{\text{base}}$. This constraint prevents the learned model from losing generation diversity and falling into 'mode collapse' due to a single high cumulative reward result. In practice, this KL-divergence has become a standard constraint in large language model finetuning [1, 18, 21].

Given preference labels and following the Direct Preference Optimization (DPO) framework [30, 21, 20], we can approximate the optimal policy $p^*$ by minimizing the upper bound:

$$\mathbb{E}_{\mathbf{x}_0^+, \mathbf{x}_0^-, t, \mathbf{x}_t^+, \mathbf{x}_t^-} \left[ -\log \sigma \left( \beta \big( \|\boldsymbol{\epsilon}_{\text{base}}(\mathbf{x}_t^+, \mathbf{c}, t) - \boldsymbol{\epsilon}_t^+\|_2^2 - \|\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t^+, \mathbf{c}, t) - \boldsymbol{\epsilon}^+\|_2^2 \right. \right.$$

$$\left. \left. - (\|\boldsymbol{\epsilon}_{\text{base}}(\mathbf{x}_t^-, \mathbf{c}, t) - \boldsymbol{\epsilon}_t^-\|_2^2 - \|\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\mathbf{x}_t^-, \mathbf{c}, t) - \boldsymbol{\epsilon}^-\|_2^2) \big) \right) \right],$$

14

```
prompt_list = [
    f'a {unique_token} {subject_token} in Vincent Van Gogh rendition',
    f'a {unique_token} {subject_token} on the Moon',
    f'a green {unique_token} {subject_token}',
    f'a {unique_token} {subject_token} with a China Great Wall in the background',
    f'a {unique_token} {subject_token} on top of wooden roof',
    f'a {unique_token} {subject_token} with the Statue of Liberty in the background',
    f'a {unique_token} {subject_token} with a volcano in the background',
    f'a {unique_token} {subject_token} on a subway platform',
]
```
Object prompts

```
prompt_list = [
    f'a {unique_token} {subject_token} wearing a green outfit',
    f'a {unique_token} {subject_token} in Vincent Van Gogh rendition',
    f'a {unique_token} {subject_token} with a China Great Wall in the background',
    f'a {unique_token} {subject_token} is swimming in a pool',
    f'a {unique_token} {subject_token} on a subway platform',
    f'a {unique_token} {subject_token} with the Statue of Liberty in the background',
    f'a {unique_token} {subject_token} on the Moon',
    f'a green {unique_token} {subject_token}',
]
```
Live subject prompts

Figure 6: Training prompts for objects and live subjects.

Table 4: RPO Hyperparameters

| Parameter | Value |
|---|---|
| *Optimization* | |
| optimizer | AdamW [17] |
| learning rate | $5 \times 10^{-6}$ |
| weight decay | 0.01 |
| gradient clip norm | 1.0 |
| *RPO* | |
| regularizer weight, $\beta$ | 1.0 |
| gradient steps | 400 |
| training preference weight, $\lambda_{\text{train}}$ | 0.0 |
| validation preference weight (default), $\lambda_{\text{val}}$ | 0.3 |

where $\epsilon^+$ and $\epsilon^-$ are independent samples from a Gaussian distribution, $\mathbf{x}_t^+$ and $\mathbf{x}_t^-$ are perturbed versions of $\mathbf{x}_0^+$ and $\mathbf{x}_0^-$ that depend on $\epsilon^+$ and $\epsilon^-$, and $\mathbf{x}_0^+$ is preferred to $\mathbf{x}_0^-$. A detailed derivation can be found in [30].

## A.2 Experimental Details

**Training Prompts.** We collect 8 training prompts: 6 re-contextualization, 1 property modification and 1 artistic style transfer for objects. 5 re-contextualization, 1 attribute editing, 1 artistic style transfer and 1 accessorization for live subjects. The trainig prompts are shown in Figure 6.

**Experimental Setup** During training we use $\lambda_{\text{train}} = 0$ for the $\lambda$-Harmonic reward function to generate the preference labels. We evaluate the model performance by $\lambda_{\text{val}}$-Harmonic per 40 gradient steps during training time and save the checkpoint that achieve the highest validation reward. Table 4 lists the common hyperparameters used in the generating skill set and the $\lambda_{\text{val}}$ used in the default setting.

## A.3 Additional Comparisons

**Comparison to DreamBooth and SuTI.** We observe RPO that can be faithful to both reference images and the input prompt. To investigate whether RPO can provide better quality than DreamBooth and SuTI, we follow SuTI paper and pick the robot toy as an example to demonstrate the effectiveness of RPO (Figure 7). In this example, DreamBooth is faithful to the reference image but it does not provide a good text-to-image alignment. SuTI provides an result that is fidelty to textual prompt. However, SuTI lacks fidelity to the reference image, i.e., the robot should stand with its wheels instead of legs. [7] use DreamBooth to finetune SuTI (Dream-SuTI) further to solve this failure case. Instead, RPO can generate an image not only faithful to the reference images but also align well with the input prompts.

**Comparison to different $\lambda_{\text{val}}$.** We have also added more samples for comparison of different $\lambda_{\text{val}}$ values (see Figure 8). We find that $\lambda_{\text{val}} = 0.5$ encourages the learned model to retain output diversity while still aligning with textual prompts. However, the generated images invariably contain a sofa, which is unrelated to the subject images. This occurs because every training image includes a sofa. A large $\lambda_{\text{val}}$ weakens the regularization strength and leads to overfitting. Nevertheless, a small value of

15

Table 5: Different $\lambda_{\text{train}}$-Harmonic reward comparison for evaluating fidelity over multiple subjects and prompts.

| Configuration | DINO ↑ | CLIP-I ↑ | CLIP-T ↑ |
|---|---|---|---|
| $\lambda_{\text{train}} = 0.0$ (default) | $0.581 \pm 0.113$ | $0.798 \pm 0.039$ | $\mathbf{0.329 \pm 0.021}$ |
| $\lambda_{\text{train}} = 0.3$ | $0.646 \pm 0.083$ | $0.815 \pm 0.037$ | $0.315 \pm 0.026$ |
| $\lambda_{\text{train}} = 0.5$ | $0.649 \pm 0.080$ | $0.829 \pm 0.039$ | $0.314 \pm 0.026$ |
| $\lambda_{\text{train}} = 0.7$ | $\mathbf{0.651 \pm 0.088}$ | $\mathbf{0.831 \pm 0.033}$ | $0.314 \pm 0.026$ |

Table 6: Comparison for harmonic mean and arithmetic mean reward function.

| Configuration | DINO ↑ | CLIP-I ↑ | CLIP-T ↑ |
|---|---|---|---|
| $\lambda_{\text{val}} = 0.3$ (arithmetic mean) | $0.638 \pm 0.083$ | $0.823 \pm 0.037$ | $0.318 \pm 0.027$ |
| $\lambda_{\text{val}} = 0.5$ (arithmetic mean) | $\mathbf{0.702 \pm 0.078}$ | $\mathbf{0.857 \pm 0.047}$ | $0.295 \pm 0.017$ |
| $\lambda_{\text{val}} = 0.7$ (arithmetic mean) | $0.678 \pm 0.085$ | $0.851 \pm 0.041$ | $0.299 \pm 0.026$ |
| $\lambda_{\text{val}} = 0.3$ (harmonic mean) | $0.581 \pm 0.113$ | $0.798 \pm 0.039$ | $\mathbf{0.329 \pm 0.021}$ |
| $\lambda_{\text{val}} = 0.5$ (harmonic mean) | $0.652 \pm 0.082$ | $0.833 \pm 0.041$ | $0.314 \pm 0.022$ |
| $\lambda_{\text{val}} = 0.7$ (harmonic mean) | $0.679 \pm 0.085$ | $0.850 \pm 0.045$ | $0.304 \pm 0.023$ |

$\lambda_{\text{val}}$ can potentially eliminate background bias. We highlight that this small $\lambda_{\text{val}}$ not only encourages diversity but also mitigates background bias in identity preservation and enables the model to focus on the subject.

**Comparison to different $\lambda_{\text{train}}$.** During our experiments, we use the default value for $\lambda_{\text{train}} = 0$. Thus, we also investigate how different $\lambda_{\text{train}}$ will effect the performance of RPO. We test three different values of $\lambda_{\text{train}} = \{0.3, 0.5, 0.7\}$ and report their results in DreamBench (Table 5). We observe that the image-to-image alignment increases with larger , but the text-to-image alignment decreases because the preference model tends to favor alignment with reference images and ignores the prompt alignment.

**Comparison to different Pythagorean means.** Further, we examine different Pythagorean means of ALIGN-I and ALIGN-T scores. We replace the harmonic mean as the arithmetic mean and test three different $\lambda_{\text{val}} = \{0.3, 0.5, 0.7\}$. We report the results of the arithmetic mean reward in Table 6. The arithmetic mean is not very sensitive to smaller values; it tends to maximize higher values to achieve a better final score. In practice, ALIGN-I will receive a higher value (this effect can be seen from CLIP-I and CLIP-T in Tables 1 to 3). Thus, the model will tend to optimize image-to-image alignment and achieve good results on DINO and CLIP-I but have a lower score for text-to-image alignment.

**Comparison to DPO** The original DPO is not suitable for subject-driven tasks because the datasets do not contain preference labels. We introduce the $\lambda$-harmonic function and design a variant of DPO for this task. We implement a DPO diffusion [30] (without similarity loss) using preference labels for image-to-image similarity and text-to-image alignment. We choose $\lambda_{\text{train}} = 0.5$ since this value assigns equal weights to the image-to-image similarity and text-to-image alignment. For a fair comparison, we also repost the results from RPO with the same $\lambda_{\text{train}}$ value. The results on DreamBench for these two methods are shown in Table 7. These results show that DPO can capture the text-to-image alignment from the preference labels. However, without $\mathcal{L}_{\text{sim}}$, DPO faces a significant overfitting problem; i.e., it achieves high text-to-image alignment but cannot preserve the subject's unique features.

**Aesthetic comparison.** Limited evaluation metrics are a common issue in subject-driven tasks. In Table 8, we report the average aesthetic scores [25] of the real reference images in DreamBench and the average aesthetic scores obtained with the best CLIP I/T $\lambda$ configuration ($\lambda_{\text{val}} = 0.5$) in DreamBench. We observe that RPO does not decrease the quality of images. Instead, the generated images achieve slightly better quality than the reference images.

16

Table 7: DPO comparison for evaluating fidelity over multiple subjects and prompts on DreamBench

| Method | DINO ↑ | CLIP-I ↑ | CLIP-T ↑ |
|---|---|---|---|
| DPO | 0.338 | 0.702 | **0.334** |
| Ours: RPO | **0.649** | 0.819 | 0.314 |

Table 8: Aesthetic score comparison for the real reference images and RPO generated images

| Method | Aesthetic Score ↑ |
|---|---|
| Real images | $5.145 \pm 0.312$ |
| Ours: RPO ($\lambda_{\mathrm{val}} = 0.5$) | $\mathbf{5.208 \pm 0.327}$ |

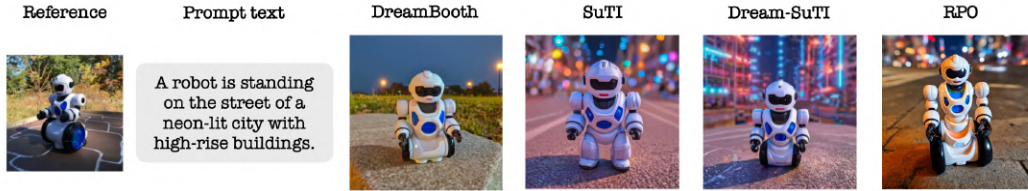



Figure 7: Comparison between DreamBooth, SuTI, Dream-SuTI, and RPO, adapted from [7]

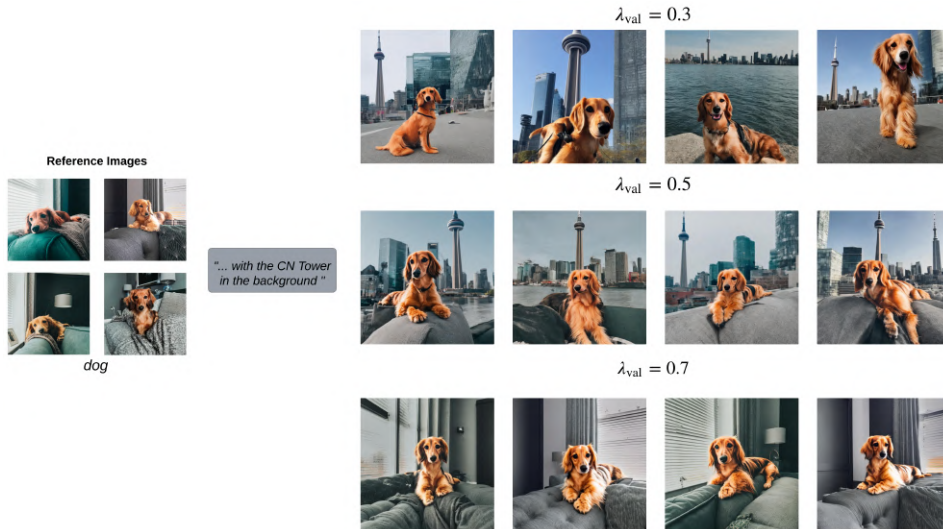



Figure 8: Additional samples of different $\lambda_{\mathrm{val}}$. A small $\lambda_{\mathrm{val}}$ leads to more diverse generated images.

17

### A.4 Limitations

Figure 9 illustrates some failure examples of RPO. The first issue is context-appearance entanglement. In Figure 9, the learned model correctly understands the keyword *blue house*; however, the appearance of the subject is also altered by this context, e.g. the color of the backpack has changed, and there is a house pattern on the backpack. The second issue is incorrect contextual integration. We conjecture that this failure is due to the rarity of the textual prompt. For instance, imagining a cross between a chow chow and a tiger is challenging, even for humans. Third, although RPO provides regularization, it still cannot guarantee the avoidance of overfitting. As shown in Figure 9, this may be because, to some extent, the visual appearance of sand and bed sheets is similar, which has led to overfitting issues in the model.

### A.5 Future Work

The overfitting failure case leads to a future work direction: *can online RL improve regularization and avoid overfitting?* The second direction for future work involves implementing the LoRA version for RPO and comparing it to LoRA DreamBooth. Last but not least, we aim to identify or construct open-source, subject-driven datasets for comparison. Currently, DreamBench is the only open-source dataset we can access and evaluate for model performance. Nevertheless, we should create a larger dataset that includes more diverse subjects to verify the effectiveness of different algorithms.

### A.6 Broader Impacts

The nature of generative image models is inherently subjected to criticism on the issue of privacy, security and ethics in the presence of nefarious actors. However, the core of this paper remains purely on an academic mission to extend the boundaries of generative models. The societal consequences of democratizing such powerful generative models is more thoroughly discussed in other papers. For example, Bird et al. outlines the classes of risks in text-to-image models [2] in greater detail and should be directed to such papers. Nevertheless, we play our part in the management of such risks by avoiding the use of identifiable parts of humans in the reference sets.

### A.7 Skill Set

The skill set of the RPO-trained model is varied and includes re-contextualization (Figure 10), artistic style generation (Figure 11), expression modification (Figure 12), subject accessorization (Figure 13), color editing (Figure 14), multi-view rendering (Figure 15), novel hybrid synthesis (Figure 16), and novel prompt generation (Figure 17).

18

## Context-apperance entanglement

*A [V] backpack with a blue house
in the background*

## Incorrect contextual integration

*A cross of a [V] chow
chow and a tiger*

## Overfitting

*A [V] pink sunglasses in
the Sahara Desert*

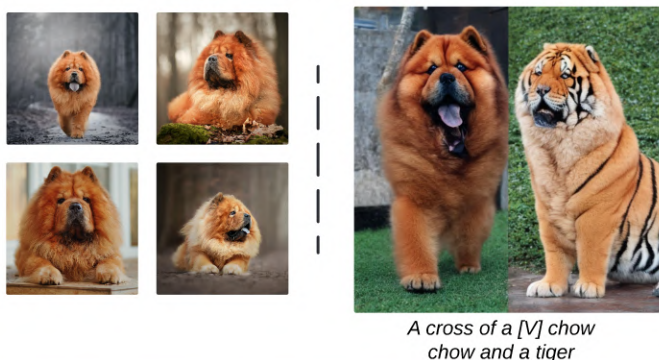
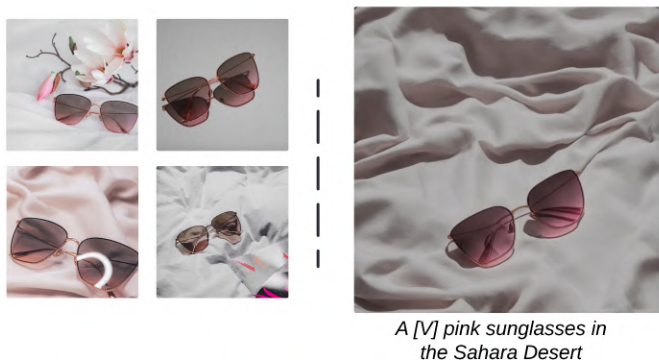



Figure 9: RPO's failure modes include: (1) The context and the appearance of the subject becoming entangled. (2) The trained model failing to generate an image with respect to the input prompt. (3) The trained model still overfitting to the training set.


19


**Reference Images**

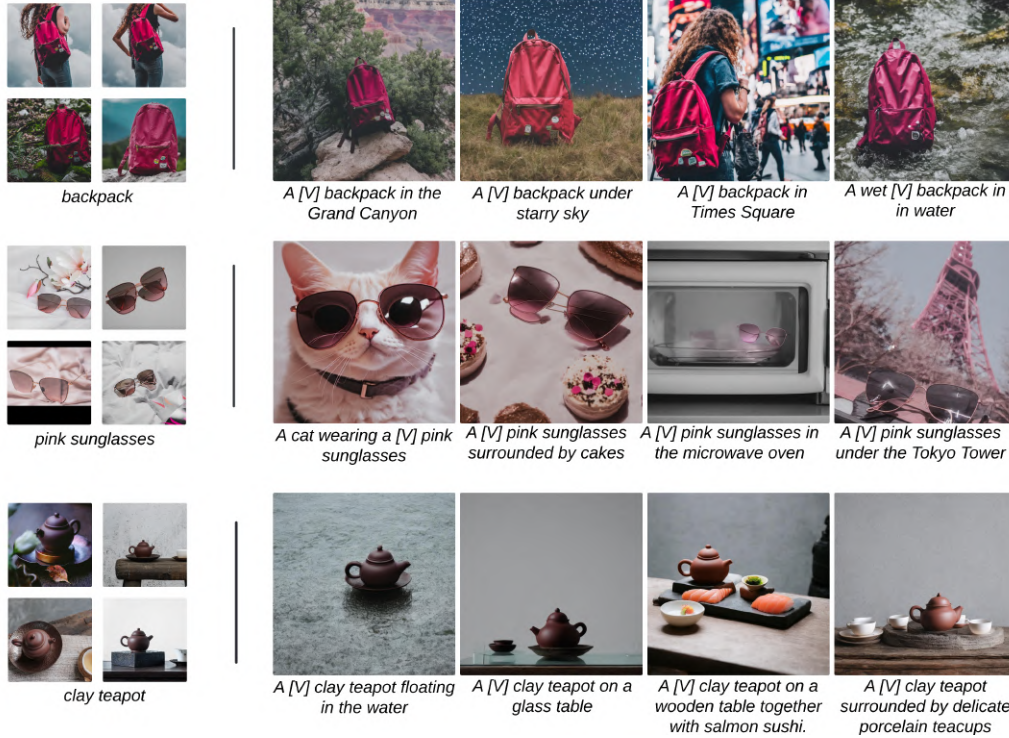



Figure 10: Re-contextualization samples from the RPO algorithm. RPO is able to generate images of specific subjects in unseen environments while preserving the identity and details of the subjects and be faithful to the input prompts. Reference images are shown on the left. We display the generated images along with their textual prompts on the right.

**Reference Images**

Figure 11: Artistic style rendition samples from the RPO algorithm. These art renditions can be applied to a dog given a prompt of ``a [painter] styled painting of a [V] dog''. The identity of the subject is preserved and faithfully imitates the style of famous painters.

**Reference Images**

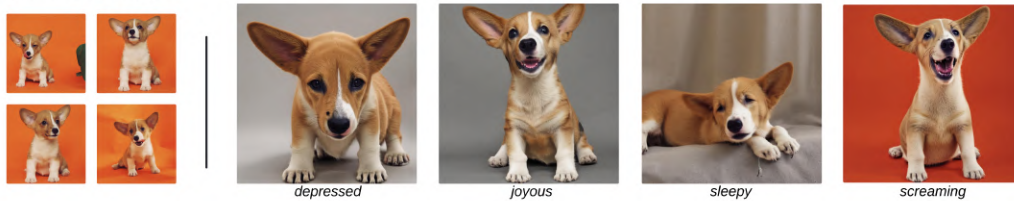



Figure 12: Expression modification samples from the RPO algorithm. RPO can integrate subject with various unseen expressions in the reference images. We also notice the pose of generated images, e.g., a [V] sleepy dog, were not displayed in the training set.

20

Figure 13: Accessories samples from the RPO algorithm. Conditioned the prompts, ''`a [V] chow chow wearing a [target outfit]`'', the generated images retains the unique features of the reference images, e.g., the hair color and breed of the subject dog. The interaction between subject dog and the outfits is realistic.

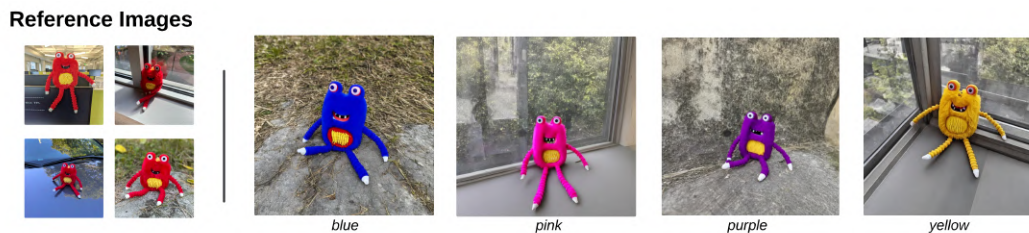



Figure 14: Color editing samples from the RPO algorithm. We display color modifications using prompts such as ''`a [target color] [V] monster toy`''. The identity of the subject is preserved.

Figure 15: Multi-view samples from the RPO algorithm. We generate images from specified viewpoints of the subject. For the top and bottom views, we use the prompts ''`a [V] cat looking up/down at the camera`''. For the back and side views, we apply the prompts ''`a back/side view of a [V] cat`''.

21

**Reference Images**

bear          koala          lion          panda

Figure 16: Novel hybrid synthesis samples from the RPO algorithm. We apply the prompts ''a cross of a [V] chow chow and [target species]'' to the RPO-trained model to generate these images. We highlight that RPO can synthesize new hybrids that retain the identity of the subject chow chow and perform property modifications.

**Reference Images**

" a cyberpunk [V] dog with neon fur patterns and robotic limbs walking down a rain-soaked alleyway in a bustling city"

"a superhero [V] dog soaring through the sky over a city skyline, with a cape billowing in the wind"

"an astronaut [V] dog in a sleek space suit floating weightlessly inside a spacecraft."

"an underwater [V] dog wearing a diving suit swimming through a vibrant coral reef surrounded by exotic marine life"

Figure 17: Novel prompts results from the RPO-trained model. We use ChatGPT to create 4 different highly imaginative prompts for a specific dog. We select 4 images for each prompt. The results demonstrate the generalization capability of RPO for handling novel prompts.

22

## NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

**The checklist answers are an integral part of your paper submission.** They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS paper checklist",**
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

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

   Answer: [Yes]

   Justification: The scope and contributions are clearly defined in the abstract and introduction.

   Guidelines:
   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

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

   Answer: [Yes]

   Justification: The limitations of our technique is discussed in the Appendix where we show certain cases that are not desirable.

23

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

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

   Answer: [NA]

   Justification: Our works focus on the experimental/empirical performance of subject-driven generation only. Due to the page constraints and the scope, we decided not to pursue theoretical results for now.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

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

   Answer: [Yes]

   Justification: The details are mentioned in the Appendix A.2 where it states specific configurations needed to reproduce the main experiments, including the prompts.

   Guidelines:

24

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [Yes]

   Justification: Our paper uses the DreamBench dataset for the experiments, which is publicly available for anyone interested in reproducing the main results. The code will be submitted as a ZIP file.

   Guidelines:
   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.

25

- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

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

   Answer: [Yes]

   Justification: The details are mentioned in the Appendix A.2. In Table 3, the $\lambda$ which is an important hyperparameter in our experiments, is stated along with the metrics.

   Guidelines:
   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

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

   Answer: [Yes]

   Justification: Experiments are conducted for each subject across various prompts, where a metric is computed individually. The standard deviation over all prompts and subjects is provided in the ablation study in the experiment section.

   Guidelines:
   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

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

   Answer: [Yes]

   Justification: The compute resources used in this paper is described in Method section.

26

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

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

   Answer: [Yes]

   Justification: For our research and experiments, we use public datasets and models that have been used in prior published works. Our research does not involve direct interaction with human subjects.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

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

    Answer: [Yes]

    Justification: Given the nature of our research, it necessitates the general discussion of the potential negative societal impacts which takes place in Appendix A.6.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

27

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

Answer: [No]

Justification: Safeguards would fall outside the scope of our research and we mainly focus on subjects that are inanimate. No experiments explicitly use personally identifiable features of humans as a subject.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

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

Answer: [Yes]

Justification: The authors of Stable Diffusion, DreamBench and SuTI are cited and credited in this paper.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

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

Answer: [Yes]

Justification: The asset mainly consist of code which contains instruction on how to run it.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.

28

- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

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

Answer: [NA]

Justification: We do not crowdsource images by ourselves; we use images from DreamBench which has been vetted through.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

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

Answer: [NA]

Justification: As mentioned previously, there is no crowdsourcing involved.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

29