IPGO: INDIRECT PROMPT GRADIENT OPTIMIZATION FOR TEXT-TO-IMAGE MODEL PROMPT FINETUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-to-Image (T2I) Diffusion models have become the state-of-the-art for image generation, yet they often fail to align with specific reward criteria such as aesthetics or human preference. We propose **Indirect Prompt Gradient Optimization (IPGO)**, a novel and parameter-efficient framework that enhances prompt embeddings by injecting a few learnable text embeddings as prefix and suffix around the original prompt embeddings. IPGO leverages low-rank approximation and rotation, while enforcing range, orthonormality, and conformity to ensure stability. We evaluate IPGO against six baseline methods under prompt-wise training with three reward models targeting image aesthetics, image-text alignment, and human preferences across three datasets of varying prompt complexity. The results show that, despite using only a single NVIDIA L4 GPU and over 250 times fewer parameters, IPGO consistently outperforms all baselines over strong competitors such as DRaFT-1 and TextCraftor. Ablation studies further highlight the contributions of each IPGO component and optimization constraint, while additional experiments demonstrate IPGO's adaptability across various T2I diffusion models.

1 INTRODUCTION

Text-to-Image (T2I) Diffusion models have emerged as state-of-the-art pipelines for image generation (Liu et al., 2024; Zhang et al., 2023). However, images generated from user prompts often fail to meet specific downstream objectives, such as aesthetic quality or alignment with human preferences (Liu et al., 2024). Further aligning generated images with human evaluations is therefore highly desirable. A number of approaches have been proposed to address this challenge Liu & Chilton (2022); Black et al. (2023); Prabhudesai et al. (2023); Liu et al. (2024); Hao et al. (2024); Li et al. (2024b); Fan et al. (2024); Li et al. (2024b), but they typically rely on data- or computing-intensive training paradigms such as Supervised Fine-Tuning (SFT) and Reinforcement Learning (RL), or require significant modifications to the generative model for each downstream task. As a result, there is continued interest in developing more efficient frameworks for alignment-driven image optimization.

In this paper, we introduce a novel approach, called **I**ndirect **P**rompt **G**radient **O**ptimization (IPGO), which improves image quality during inference by optimizing the prompt itself. Our method is inspired by the linguistic notion of semantic heads or tails – short phrases placed at the beginning or end of a clause to disambiguate, provide context, and add emphasis, intensity, or meaning to it. Analogously, IPGO *injects a few embeddings* at the beginning (prefix) and end (suffix) of the original text prompt embeddings. We then employ constrained gradient-based optimization of a rotated low-rank approximation to these embeddings to enhance the alignment of the visual representation of the prompt with human judgments through reward guidance. This prefix-suffix tuning strategy offers a modular, parameter-efficient approach to prompt optimization, requiring no modifications to the diffusion model or text encoder and enabling fast training.

We evaluate IPGO on the Stable Diffusion model (Rombach et al., 2022) using a single L4 GPU with 22.5GB of VRAM. Experiments target three reward models in single-objective optimization settings: (i) image aesthetics (Schuhmann, 2024), (ii) image-text alignment (Radford et al., 2021), and (iii) human preference scores (Wu et al., 2023). The main contributions of this study are summarized as follows:

1. We propose IPGO, a novel **parameter-efficient**, gradient-based approach to prompt optimization in the text embedding space for reward guidance of T2I diffusion models at inference. This approach optimizes **rotated low-rank prefix and suffix embeddings** inserted at both the beginning and end of the original prompt embeddings, under orthonormality, conformity, and range constraints.
2. Our experiments on three different datasets, using SDv1.5 and three reward functions, show that for prompt-wise training at inference, IPGO **consistently outperforms six state-of-the-art methods**, achieving an average **improvement of 1-3% – comparable to gains reported in previous work** (Black et al., 2023; Clark et al., 2023; Hao et al., 2024; Li et al., 2024b). These improvements hold over the strongest benchmarks (TextCraftor and DRaFT-1), **while requiring over 250 times fewer parameters.**
3. Ablation studies highlight the **individual contributions of the constraints** imposed on the optimization, as well as the other IPGO components, including low-rank approximations and the rotation. Futhermore, an additional experiment demonstrates that IPGO can be **applied across a wide range of diffusion models**, including more advanced architectures like SDXL and SD3.

2 Related Work

Text-to-Image Diffusion Probabilistic Models Foundational work in T2I generation using diffusion models includes diffusion-probabilistic models (Sohl-Dickstein et al., 2015), score-based generative models (Song & Ermon, 2019) and the landmark denoising diffusion probabilistic model (DDPM; Ho et al., 2020). Subsequent models, such as GLIDE (Nichol et al., 2021) and Imagen (Saharia et al., 2022) also operate the diffusion process directly in the pixel space. In contrast, methods like Stable Diffusion (Rombach et al., 2022) and DALL-E (Ramesh et al., 2022) apply the diffusion process in a low-dimensional embedding space. Notably, Stable Diffusion has demonstrated superior image quality and efficiency (Zhang et al., 2023), and several extensions to it have been proposed (e.g., Esser et al., 2024; Peebles & Xie, 2023; Podell et al., 2023). A key challenge with diffusion models, however, is their potential misalignment with human preferences. A stream of recent work addresses this by controlling models towards preferred properties, either during training or via training-free methods (Liu et al., 2024).

Parameter Efficient Finetuning (PEFT) (Han et al., 2024) enables task-specific adaptations of LLMs and T2I Diffusion Models by optimizing a small subset of parameters while keeping most parameters frozen (Han et al., 2024), leading to lower computational demands. Some key PEFT approaches include LoRA, which injects trainable low-rank matrix approximations in the otherwise fixed LLM architecture (Hu et al., 2021), and Prefix Tuning, which inserts trainable parameters in various layers of the LLM (Li & Liang, 2021).

Training-based alignment (Liu et al., 2024) uses supervised fine-tuning (SFT) of the diffusion model combined with reinforcement learning from human feedback (RLHF) to align the model with human preferences, approximated via a reward model. Models in this category, such as ReFL (Xu et al., 2024), DDPO (Black et al., 2023), AlignProp (Prabhudesai et al., 2023), DRaFT (Clark et al., 2023), DPOK (Fan et al., 2024), and DPO-Diffusion (Wang et al., 2024), rely on gradient-based fine-tuning of the diffusion model. Alternatively, models can be directly optimized on preference data using methods like Diffusion-DPO (Wallace et al., 2024), D3PO (Yang et al., 2024), and SPO (Liang et al., 2024). Training-based alignment methods often require considerable computational resources.

Training-free alignment (Liu et al., 2024) aligns diffusion models with human preferences *without the need for fine-tuning the diffusion model*. The first stream of research uses both manual and systematic approaches to prompt optimization (Oppenlaender, 2023; Wang et al., 2023). Automatic prompt optimization methods, such as Promptist (Hao et al., 2024) and OPT2I (Mañas et al., 2024)), leverage LLMs to refine prompts. The second stream focuses on modifying negative prompts using LLMs (e.g., DPO-Diffusion, Wang et al., 2024) or directly learning negative embeddings (e.g., ReNeg, Li et al., 2024a)). The third stream involves editing the initial latent state, as seen in ReNO (Eyring et al., 2024) for one-step diffusion models. The fourth stream optimizes prompt text embeddings, to which our IPGO belongs, as detailed below.

Alignment through prompt embedding optimization includes methods such as PEZ (Wen et al., 2024), which aligns an image with text embeddings of prompts that reflect both the image content and style. Textual Inversion (Gal et al., 2022) aligns new word tokens with novel objects or styles. TextCraftor (Li et al., 2024b) and TexForce (Chen et al., 2024) align generated images with rewards by fine-tuning the CLIP text encoder within the diffusion pipeline.

Our proposed method **IPGO also optimizes prompt text embeddings**. However, *unlike prompt embedding methods* such as PEZ (Wen et al., 2024) and Textual Inversion (Gal et al., 2022), IPGO operates without accessing ground-truth images, but leverages abstract reward models to guide prompt optimization within the existing text embedding space. *In contrast to TextCraftor (Li et al., 2024b) and TexForce (Chen et al., 2024)*, which change the text embedding space by fine-tuning the entire text encoder, IPGO explores the embedding space without altering the encoder's parameters, and keeps the original prompt intact, which allows better user control over the prompt's visual representation, and uses much less than one percent of the parameters of these models, thus allowing for faster training. In addition, *counter to Adapter-based PEFT approaches such as LoRA (Hu et al., 2021)* which insert trainable parameters in multiple layers of a pre-trained and frozen model, for example in the attention blocks in Transformer layers, IPGO inserts the trainable prefix and suffix directly into the prompt embeddings, and is thus much more parameter efficient.

In the benchmarking experiments we will demonstrate that IPGO outperforms six benchmark approaches, including PEFT and fully finetuned models, across three datasets and three reward models. *IPGO outperforms its closest competitors TextCraftor and DRaFT-1 in most of the scenarios, but with less than 0.5% of the full parameters, by a significant margin of 1-3%.*

3 PRELIMINARIES

Diffusion Models Diffusion models generate images conditioned on a text prompt by sequentially denoising an image from pure Gaussian noise using an error model ϵ_ϕ (Rombach et al., 2022), parameterized by ϕ. The model ϵ_ϕ predicts the noise in each image x_t, which is obtained by progressively adding Gaussian noise ϵ to the original image x_0 at each step $t = 0, .., T$ (Ho & Salimans, 2022).

Reward Models Typically, a generated image is evaluated using a pre-trained reward model, $\mathcal{S}$, which assesses how well the image aligns with human evaluations. For each image x generated by the diffusion model in response to a prompt p, the reward model assigns a reward $\mathcal{S}(x, p)$. This reward $\mathcal{S}(x, p)$ is then used to guide the diffusion model towards generating images with a higher reward. Widely used reward models are the LAION aesthetic predictor V2 (Schuhmann, 2024), the CLIP loss derived from the multi-modal CLIP model(Radford et al., 2021), and the human preference score HPSv2 (Wu et al., 2023). These models have played a critical role in aligning the outputs of diffusion models with human preferences in research and practice.

4 METHODS

Suppose we have a prompt p; a trained reward model $S(x, p)$ on image x and the prompt p; a text encoder $\mathcal{T}(\cdot)$ which converts p to its text embeddings $\mathcal{T}(p) \in \mathbb{R}^{d\times K}$, where d is the embedding dimension and K is the length of the tokenized prompt; and finally a diffusion model characterized by $q_{\text{image}}(x_0|\mathcal{E}, z_T)$, the probability distribution of the image x_0, given prompt-text embeddings $\mathcal{E}$ and a fixed latent state z_T at timestep T.

4.1 IPGO

IPGO adds to the original embeddings $\mathcal{T}(p)$, of a text prompt p, a prefix V_{pre} and a suffix V_{suf}, consisting of N_{pre} and N_{suf} trainable embeddings, each of dimension d, and parameterized by Ω_{IPGO} (see the following paragraphs). IPGO inserts the prefix at the beginning and the suffix at the end of $\mathcal{T}(p)$, the embeddings of the prompt p, thereby producing an augmented set of text embeddings:

$$\mathcal{E}(V_{\text{pre}}, p, V_{\text{suf}};\ \Omega_{\text{IPGO}}) = V_{\text{pre}} \oplus \mathcal{T}(p) \oplus V_{\text{suf}}, \tag{1}$$

where $\mathcal{E}(V_{\text{pre}}, p, V_{\text{suf}};\ \Omega_{\text{IPGO}}) \in \mathbb{R}^{d\times(N_{\text{pre}}+K+N_{\text{suf}})}$ and $\oplus$ stands for the concatenation in the second dimension. IPGO optimizes Ω_{IPGO} such that the expected rewards of the images sampled from

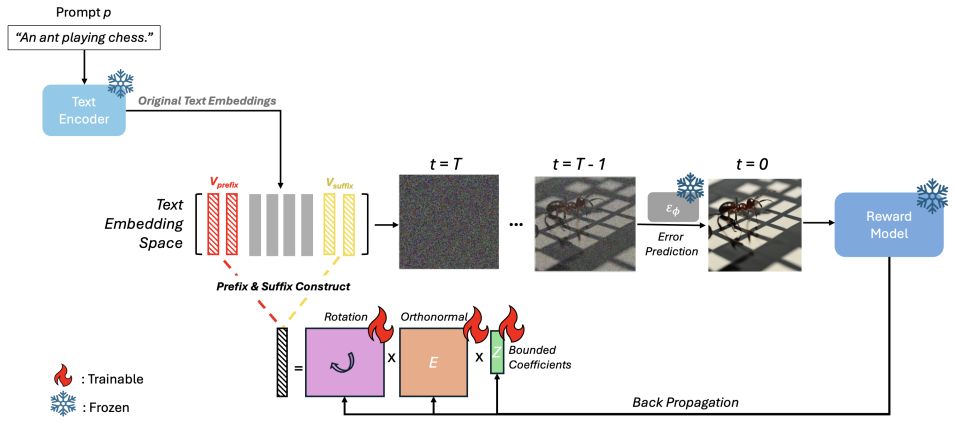


Figure 1: IPGO inserts trainable prefix and suffix embeddings, leveraging low-rank approximation and rotation, to the text embeddings of the prompt in the CLIP text encoder space/text embedding space, and then sends back reward signals through backpropagation under three constraints: Orthonormality, Range and Conformity.

q_{image} conditioned on a fixed z_T and $\mathcal{E}(V_{\text{pre}}, p, V_{\text{suf}};\ \Omega_{\text{IPGO}})$ are maximized, which is equivalent to minimizing the following single-reward loss function:

$$\mathcal{L}(\Omega_{\text{IPGO}}) = -\mathbb{E}_{x_0 \sim q_{\text{image}}(x_0 | \mathcal{E}(V_{\text{pre}}, p, V_{\text{suf}};\ \Omega_{\text{IPGO}}), z_T)} \mathcal{S}\left(x_0, p\right). \tag{2}$$

In the following sections, we present the motivation behind our approach and outline the overall framework. Figure 1 presents a schematic overview of the IPGO methodology.

Constrained Prefix-Suffix Tuning. Inspired by Prefix-Tuning (Li & Liang, 2021), IPGO adds extra continuously differentiable embeddings before and after the original text embeddings, as described by equation 1. However, Li & Liang (2021) show that *directly updating prefix embeddings may lead to unstable optimization*. Thus, rather than directly optimizing V_{pre} and V_{suf}, we reparameterize them as rotated linear combinations of a set of low-dimensional learnable embeddings. We parameterize V_* ($*$ stands for "pre" or "suf" from here on) by the following:

$$V_* = \tilde{R}_{2,\theta_2^*} \tilde{R}_{1,\theta_1^*} E_* Z_*, \tag{3}$$

where $E_* \in \mathbb{R}^{d \times m_*}$ is a trainable set of *base text embeddings* and m_* is the length of the basis. $Z_* \in \mathbb{R}^{N_* \times m_*}$ are linear coefficients for the basis. The intuition behind equation 3 is that the product $E_* Z_*$ can be seen as an m_*-dimensional low-rank approximation to the original (unconstrained) $d-$dimensional embedding, using an orthonormal basis E_*, akin to LoRA (Hu et al., 2021), with parameters constrained to $Z_* \in [-1, 1]$. The d dimensions of $E_* Z_*$ are rotated pairwise via orthogonal rotation matrices R_{1,θ_1} and R_{2,θ_2}. Rotation parameterizations of embeddings have been previously used by a.o. Su et al. (2024). Here, the basis E_* helps explore a subspace of the text embedding space that is rotated to fit with the reward guidance. Exploring the orthogonal subspace along with the rotation parameters is more efficient than exploring the original embeddings (see Appendix A). This intuition motivates the constraints below.

Rotation. We apply two rotation matrices $\tilde{R}_{1,\theta_1^*}$ and $\tilde{R}_{2,\theta_2^*}$. The rotation matrices are composed of the 2×2 elementary matrix controlled by angle $\theta \in (-\frac{\pi}{2}, \frac{\pi}{2}]$:

$$R_{e,\theta} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}. \tag{4}$$

Given θ_1^* and θ_2^*, we define $\tilde{R}_{1,\theta_1^*} \in \mathbb{R}^{d \times d}$ and $\tilde{R}_{2,\theta_2^*} \in \mathbb{R}^{d \times d}$ by:

$$\tilde{R}_{1,\theta_1^*} = I_{d/2} \otimes R_{e,\theta_1^*}, \quad \tilde{R}_{2,\theta_2^*} = \begin{bmatrix} R_{e,\theta_2^*,(2)} & & \\ & I_{d/2-1} \otimes R_{e,\theta_2^*} & \\ & & R_{e,\theta_2^*,(1)} \end{bmatrix}, \tag{5}$$

where $\otimes$ is the tensor product, I_a is the identity matrix of size a, $R_{e,\theta_2^*,(i)}$ is the i^{th} row of the elementary rotation matrix R_{e,θ_2^*}. To interpret, $\tilde{R}_{1,\theta_1^*}$ rotates pairs $(2j-1, 2j)$ and $\tilde{R}_{2,\theta_2^*}$ rotates pairs $(2j, 2j+1)$ of the coordinates of the embedding $E_* Z_*$, where $j = 1, \ldots, d/2$ and the $(d+1)^{th}$ coordinate is the 1^{st} coordinate. Rotations are advantageous in two ways. First, they introduce non-linearity. Second, the rotation parameterization accelerates the search process via their gradient directions (see Appendix A). The pairwise rotation parameterization in equations 5, is much more parsimonious than a full $d \times d$ rotation matrix.

Constraints. To optimize the subspace and preserve the structural integrity of the text embeddings generated by the text encoder $\mathcal{T}$, three constraints are imposed. First, we impose an ***Orthonormality*** constraint on the text embedding basis E_*, i.e., $E_* E_*^T = I_{m_*}$. Second, we impose a ***Range*** constraint on the affine transformation coefficients $Z_* \in [-1, 1]$, which normalizes the lengths of the prefix and suffix embeddings so that they do not perturb the semantics of the original prompt. Third, we add a ***Conformity*** constraint to ensure that the average of the IPGO embeddings (equation 1) are the same as the average of the original prompt, *mean* $(\mathcal{T}(p))$, promoting coherence of the generated prefix and suffix embeddings with the original prompt (see Appendix C for details).

Thus, IPGO has parameters $\Omega_{\text{IPGO}} = \{E_{\text{pre}}, E_{\text{suf}}, \theta_1^{\text{pre}}, \theta_2^{\text{pre}}, \theta_1^{\text{suf}}, \theta_2^{\text{suf}}, Z_{\text{pre}}, Z_{\text{suf}}\}$, optimized by the objective in equation 2, and subject to $Z_{*,(ij)} \in [-1, 1]$, and $(\theta_1^*, \theta_2^*) \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right]^2$, and the orthonormality and conformity constraints.

5 Experiments and Results

We conduct a set of experiments to evaluate the performance of IPGO across six benchmark models on three datasets. In Section 5.1, we describe the experiment settings, while Section 5.2 introduces the benchmarks, and Section 5.3 presents the results.

5.1 Experiment Settings

Datasets. Three datasets are considered: the COCO image captions (Lin et al., 2014), DiffusionDB (Wang et al., 2022), and Pick-a-Pic (Kirstain et al., 2023). These datasets represent a wide range of prompts and images with varying levels of complexity. To assess the performance of IPGO across different categories of image captions, we conduct separate evaluations for COCO images in the following five categories: Persons, Rooms, Vehicles, Natural Scenes, and Buildings. For each category, we randomly select 60 captions. Additionally, we randomly select 300 prompts from both the DiffusionDB and Pick-a-Pic datasets, resulting in a total of 900 prompts for evaluation. The number of prompts in our experiments substantially exceeds those used in recent experiments, such as (Black et al., 2023) and (Wang et al., 2024), both of which relied on approximately 600 prompts.

Training. All experiments with IPGO, which has a total of 0.47M parameters, are conducted on a single NVIDIA L4 GPU with 22.5GB of memory. IPGO takes at most 12GB of memory for all tasks. The backbone diffusion model used is Stable-Diffusion (SD)v-1.5, chosen for its balance between generation quality and computational efficiency (Li et al., 2024b; Podell et al., 2023). Note that IPGO is directly applicable to other diffusion models as well (as shown in Section 6.2). For comparability, all models and experiments are run in identical computational environments. In contrast, the benchmark TextCraftor (introduced below) requires a single A100 GPU.

Reward Models. To ensure the flexibility and broad applicability of IPGO, we consider publicly available reward models $\mathcal{S}$. Specifically, we use the LAION aesthetic predictor V2 (Schuhmann, 2024), the CLIP loss from the multimodal CLIP model (Radford et al., 2021), and the human preference score v2 (Wu et al., 2023). These widely used reward models capture a broad spectrum of criteria, effectively representing the diverse rewards relevant to text-image alignment tasks.

5.2 Benchmarks

We evaluate IPGO using the following six benchmarks, which represent the current state of the art (SOTA) in the categories discussed in Section 2. The first baseline is **Stable diffusion with a raw prompt** (Rombach et al., 2022), against which we expect IPGO to enhance performance

across all datasets and reward models. The second baseline is **TextCraftor** (Li et al., 2024b), using a fine-tunable text encoder with 123M parameters, representing the current SOTA among text-embedding-based methods. We also use two training-based methods: **DRaFT** (Clark et al., 2023) and **DDPO** (Black et al., 2023). For DRaFT, we select the DRaFT-1 variant with LoRA of rank 3 as the baseline (#parameters: 0.60M), due to its low computational cost and competitive performance (Clark et al., 2023). For DDPO (#parameters: 0.79M), we apply the default LoRA configuration. Furthermore, we include two training-free methods: **DPO-Diff** (Wang et al., 2024), and **Promptist** (Hao et al., 2024). Promptist is a multi-objective optimization method, but is applied to single-objective optimization here. Detailed qualitative comparisons between several benchmarks and IPGO can be found in Appendix B. The hyperparameter settings are provided in Table 8 in Appendix C.

5.3 Prompt-Wise Image Optimization at Inference

We train all methods listed in Table 7, using a single prompt at a time. Single image optimization during inference is more flexible and addresses concerns regarding generalization to unseen prompts (Eyring et al., 2024). For all six benchmarks we use default configurations for learning and sampling. The best loss value achieved during training is used to represent the final performance of each method. In addition to comparing the absolute loss, we compute the *percentage improvements* IPGO gains (in parentheses) over the best baseline. We also report the t *statistics and its* p *value* of the overall average improvement of IPGO over the best baseline on all three rewards.

Alignment. Table 1 presents the results of IPGO and benchmark methods for semantic alignment across three datasets. With CLIP reward, IPGO outperforms all six benchmarks in all scenarios except for COCO-Buildings. Note strong alignment for COCO-Person prompts in particular, and for the more complex prompts from the DiffusionDB. IPGO achieves the highest average alignment scores across all prompts, surpassing the top benchmark DRaFT-1 by 1.8% (t-value= 3.9, $p = 4\text{e}{-05}$). It improves alignment by 17.2% over the original SDv1.5 diffusion model (t-value = 23.8, $p < 1\text{e}{-10}$).

Dataset	*IPGO (↑)*	*SD v1.5*	*TextCraftor*	*DRaFT-1*	*DDPO*	*DPO-Diff*	*Promptist*
COCO							
Person	**0.3160** (2.4)	0.2637	<u>0.3085</u>	0.3067	0.2865	0.2911	0.2598
Room	**0.2883** (3.5)	0.2482	<u>0.2786</u>	0.2782	0.2648	0.2755	0.2398
Vehicle	**0.2986** (1.8)	0.2514	0.2928	<u>0.2934</u>	0.2755	0.2881	0.2474
Natural Scenes	**0.2922** (1.6)	0.2539	<u>0.2876</u>	0.2802	0.2614	0.2815	0.2307
Buildings	0.2846 (-1.8)	0.2439	<u>0.2859</u>	**0.2898**	0.2718	0.2794	0.2377
DiffusionDB	**0.3247** (2.5)	0.2759	0.3146	<u>0.3167</u>	0.3024	0.2929	0.2753
Pick-a-Pic	**0.3125** (0.2)	0.2681	0.3077	<u>0.3103</u>	0.2946	0.2980	0.2612
Avg. Reward	**0.3110** (1.8)	0.2654	0.3041	<u>0.3056</u>	0.2897	0.2913	0.2599

Table 1: Comparison of IPGO's **alignment scores** with six benchmarks across 900 prompts from three datasets. Bold/underline denote highest/second-highest scores. In parentheses are percentage improvements over the second-best performing model, DRaFT-1, *which are similar in magnitude to those reported in prior literature* (see the text).

Aesthetics. Table 2 presents the results for LAION aesthetics scores. IPGO outperforms all benchmarks across every dataset. Aesthetics scores are particularly high for the COCO-Person, Pick-a-Pick and DiffusionDB prompts. IPGO's average reward score is the highest across all datasets, with an improvement of 3.2% (t-value$= 6.5$, $p = 5\text{e}{-10}$) over the best benchmark, TextCraftor, and an improvement of 16.5% (t-value$= 34.0$, $p < 1\text{e}{-10}$) over the original SDv1.5 model.

Preferences. Table 3 presents the results for HPSv2 human preference scores. Again, IPGO outperforms all benchmarks on all datasets; its highest preference scores are achieved for COCO-Person and COCO-Vehicle prompts. IPGO's average reward score across all datasets is the highest, achieving an average improvement of 1.0% (t-value$= 1.7$, $p = 0.046$) over the strongest benchmark TextCraftor, where smaller effect size and statistical significance is caused by heterogeneity in HPSv2 scores across images, and 6.0% (t-value=17.4, $p < 1\text{e}{-10}$) over the original SDv1.5 model.

Dataset	*IPGO (↑)*	*SD v1.5*	*TextCraftor*	*DRaFT*	*DDPO*	*DPO-Diff*	*Promptist*
COCO							
Person	**6.2174** (4.7)	5.2447	5.8365	5.7761	5.5777	4.2865	<u>5.9401</u>
Room	**5.7549** (2.8)	5.0931	<u>5.5994</u>	5.4426	5.3700	4.1589	5.5993
Vehicle	**5.8567** (3.3)	4.9608	<u>5.6699</u>	5.5063	5.4219	4.0197	5.5643
Natural Scenes	**5.9301** (3.3)	5.0558	<u>5.7436</u>	5.6156	5.5099	4.2952	5.6483
Buildings	**5.7987** (1.9)	5.0326	<u>5.6909</u>	5.4294	5.3484	4.2777	5.6431
DiffusionDB	**6.3469** (0.2)	5.5012	<u>6.3318</u>	6.1100	5.9644	4.4350	5.6291
Pick-a-Pic	**6.2684** (4.5)	5.3289	<u>5.9978</u>	5.9048	5.7547	4.3565	5.6954
Avg. Reward	**6.1735** (2.7)	5.3025	<u>6.0117</u>	5.8563	5.72156	4.3330	5.6678

Table 2: Comparison of IPGO's **aesthetics scores** with six benchmarks across 900 prompts from three datasets. Bold/underline denote highest/second-highest scores. In parentheses percentage improvements over the second-best performing model, TextCraftor, *which are similar in magnitude to those reported in prior literature* (see the text).

Dataset	*IPGO (↑)*	*SD v1.5*	*TextCraftor*	*DRaFT-1*	*DDPO*	*DPO-Diff*	*Promptist*
COCO							
Person	**0.2950** (1.4)	0.2796	<u>0.2905</u>	0.2786	0.2819	0.2481	0.2680
Room	**0.2817** (0.4)	0.2673	<u>0.2806</u>	0.2646	0.2711	0.2430	0.2596
Vehicle	**0.2917** (0.4)	0.2761	<u>0.2905</u>	0.2755	0.2814	0.2491	0.2679
Natural Scenes	**0.2866** (0.6)	0.2721	<u>0.2848</u>	0.2667	0.2741	0.2487	0.2600
Buildings	<u>0.2867</u> (-0.5)	0.2719	**0.2882**	0.2723	0.2782	0.2580	0.2634
DiffusionDB	**0.2729** (0.4)	0.2594	<u>0.2719</u>	0.2602	0.2634	0.2381	0.2585
Pick-a-Pic	**0.2753** (0.6)	0.2621	<u>0.2741</u>	0.2647	0.2672	0.2509	0.2591
Avg. Reward	**0.2788** (0.5)	0.2650	<u>0.2776</u>	0.2655	0.2693	0.2461	0.2605

Table 3: Comparison of IPGO's **human preference scores** with six benchmarks across 900 prompts from three datasets. Bold/underline denote highest/second-highest scores. In parentheses percentage improvements over the second-best performing model, TextCraftor, *which are similar in magnitude to those reported in prior literature* (see the text).

Qualitative interpretation. Figure 2 qualitatively compares a non-cherry-picked sample of images generated with IPGO using the HPSv2 reward to those generated with the raw prompt and with TextCraftor and DRaFT-1, the best performing benchmarks (note that IPGO achieves smaller improvements over these benchmarks for HPSv2 than for the other two reward models, and that the computational environment affects the quality of each image in Figure 2 equally). Unlike DRaFT-1 and TextCraftor, which often drastically alter the image layout from that produced by the raw prompt, IPGO tends to modify or add details, while preserving the layout produced with the original prompt, thus providing enhanced control over image generation. Additional examples can be found in Appendix H.

Summary. Across all 126 (6 baselines $\times$ 7 scenarios $\times$ 3 rewards) comparisons, *IPGO yields the best performance in all but 2 of the cases*, yielding a *significant 6-17% improvement over the raw prompt*, and a *significant 1-3% improvement* over the best-performing benchmarks, across three rewards, which is similar in magnitude to improvements reported for prior models (e.g., Black et al., 2023; Clark et al., 2023; Hao et al., 2024; Li et al., 2024b), while retaining the global image layout obtained from the original prompt.

6 Further Experiments

6.1 Ablation Studies

We provide in-depth ablation studies on the components of our IPGO framework, including the three constraints, rotation, and the lengths of the prefix and suffix. All experiments are conducted on the full COCO dataset using 300 prompts across three rewards. The Stable Diffusion pipeline SDv1.3 is configured with 30 inference steps, and all models are trained for up to 30 epochs.

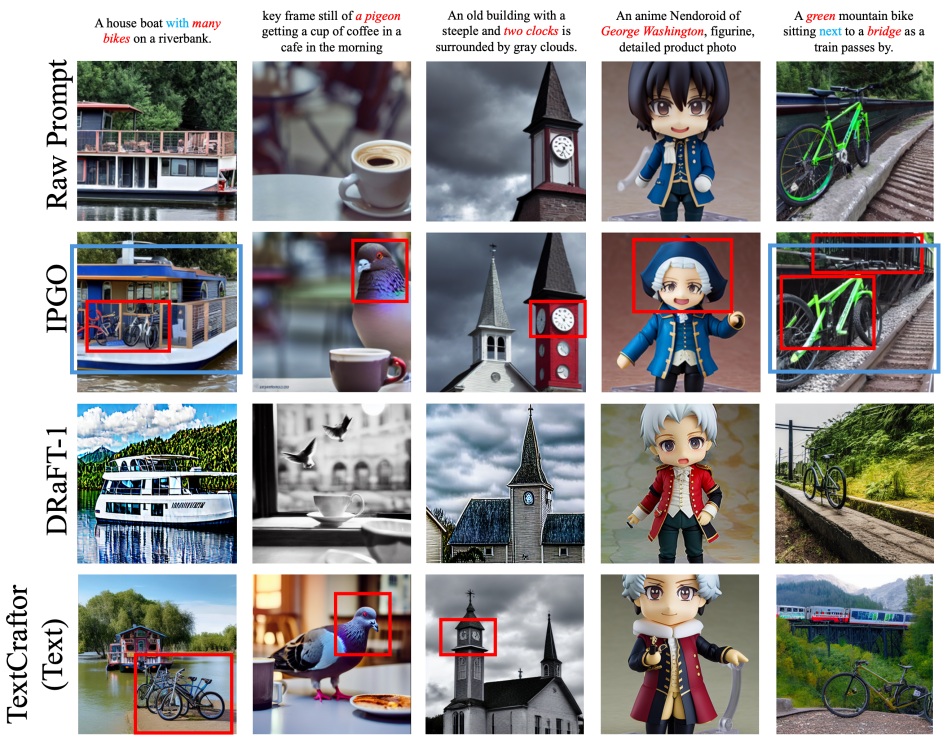


Figure 2: Example images generated with Stable Diffusion v1.5 using the raw prompt (row 1), IPGO (row 2), DRaFT-1 (row 3) and TextCraftor (row 4), towards the HPSv2 reward.

Effects of the Constraints and Rotation. We evaluate the effects of our three constraints, Orthonormality (O), Conformity (C), and Range (R), as well as the rotation component on the performance of IPGO. Specifically, we compare six scenarios, the full IPGO model that incorporates all constraints, three versions of IPGO where each of the O, C, or R constraints is omitted, IPGO without the rotation component, and finally, the IPGO without any constraints, the low-rank design or rotations (in other words, directly fine-tuning the prefix and suffix text embeddings). Table 4 presents the results.

Scenarios	*Aesthetics*	*Alignment*	*Human Preference*
Full IPGO	**6.0626**	<u>0.2771</u>	**0.2766**
w/o O	5.9892	0.2709	0.2733
w/o C	5.9422	0.2767	0.2763
w/o R	5.9151	0.2770	0.2703
w/o Rotation	5.9247	**0.2779**	0.2762
w/o O, C, R, Rotation, Low-Rank Design	5.1462	0.2442	0.2630

Table 4: Ablation experiments to test the effects of each constraint (O, C, R) and the rotation parametrization, relative to the Full IPGO model.

First, the *Full IPGO consistently performs best when none of the constraints is omitted*. Each reward benefits most from a specific optimization constraint. For aesthetics and human preference rewards, the range (R) constraint yields the most performance gains. However, for the CLIP alignment score, orthonormality (O) is the most important. Therefore, although each constraint's contribution towards the final solutions depends on the reward model, *IPGO with all three constraints combined adapts to different reward tasks to each deliver consistent image alignment*.

Second, the *effects of the rotation parametrization also depend on the reward model*: it helps aesthetics scores to improve the most (2.3%), then the human preference scores (0.1%), but it not necessarily improves alignment scores. Rotation is further explained in the Appendix A.

Third, *the Full IPGO outperforms the naive IPGO without any constraints and parameterization designs with a large margin* (17.8%, 13.5%, and 5.2% on aesthetics, alignment and human preference scores), showing the effectiveness of IPGO's parameterization and constraints in optimization over naive, unconstrained text embedding learning.

Prefix and suffix lengths. Next, we test all combinations of prefix and suffix lengths of 0, 5, 10, or 15 embeddings, excluding the $(0, 0)$ combination. Note that these scenarios include cases with only a prefix ($N_{suf} = 0$), or only a suffix ($N_{pre} = 0$). We conduct experiments with the full COCO dataset of 300 prompts and use the alignment CLIP score as the reward. We sample the images with 30 inference steps and optimize with 30 epochs.

Table 5 contains the results. First, by comparing the average rewards from the scenarios with the same total number of embeddings (e.g. $(5, 15)$, $(10, 10)$ and $(15, 5)$), we find that *equal prefix and suffix lengths tend to give a better performance, with* $(10, 10)$ *yielding the best performance*. Second, a longer prefix and suffix do not necessarily bring more performance gains, as illustrated by the difference between the performance of $(10, 10)$ and $(15, 15)$. Extremely long prefix and suffix can lead to over-parameterization which damages the image semantics, as illustrated in the Appendix E.

N_{suf} \ N_{pre}	0	5	10	15
0		0.281	0.286	0.287
5	0.284	0.284	0.286	0.284
10	0.286	0.284	**0.289**	0.283
15	0.285	0.284	0.288	0.288

Table 5: Average CLIP scores for various combinations of prefix length, N_{pre} and suffix length, N_{suf}.

6.2 Adaptivity of IPGO to Other Diffusion Models

We next illustrate that IPGO can be used with different diffusion models. In the experiments reported heretofore we choose to implement IPGO with SD-v1.5, and here we illustrate IPGO, along with TextCraftor as a benchmark, for two newer versions of Stable Diffusion, SDXL and SD3, for HPSv2 human preference rewards on 100 randomly selected prompts from the DiffusionDB data. Images are sampled with 30 steps of inference. Optimizations take 30 epochs. For Textcraftor, we only fine-tune the main text encoder due to computation limits.

Table 6 shows that *IPGO improves human preference scores over the original prompt for the SD3 (3.2%) and SDXL (4.9%) diffusion models as well*. Its performance improvement over TextCraftor also holds up for these two diffusion models.

Diffusion model	SDXL	SD3
Original	0.2523	0.2625
TextCraftor	0.2626	0.2686
IPGO	**0.2646**	**0.2710**

Table 6: HPSv2 reward for IPGO and TextCraftor on different Stable Diffusion Models for 100 randomly selected prompts from the DiffusionDB dataset.

7 Conclusion

IPGO is a parameter-efficient, gradient-based prompt-level optimization framework for alignment of generated images with prompt semantics, aesthetics, and human preferences. IPGO explores, but does not alter, the prompt embedding space via learnable rotated low-rank prefix and suffix embeddings, guided by reward gradients and constrained by range, orthonormality, and conformity. Extensive experiments over six benchmarks across three tasks, three datasets, and various diffusion model backbones demonstrate IPGO's performance gains on prompt-wise training at inference. We leave generalization to different alignment tasks, batch training, multi-criterion optimization, and interpreting the optimized pre- and suffix embeddings as topics for future research.

Ethics Statement

This work adheres to the ICLR Code of Ethics. All datasets used are publicly available and have been cited appropriately. No personally identifiable or sensitive information is included. We have taken care to ensure that our methods and results do not introduce or propagate harmful biases beyond those already present in standard benchmark datasets. The potential societal impacts of this research, both positive and negative, are discussed in Appendix F of the paper.

Reproducibility Statement

We have taken several measures to ensure the reproducibility of our results. A complete description of our model architecture and training procedure is provided in Section 4 of the main paper, with further implementation details in Appendix C. The datasets used in our experiments are described in Section 5. An anonymous Git repository containing the source code and scripts will be made available during the discussion phase to facilitate reproducibility.

References

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

Chaofeng Chen, Annan Wang, Haoning Wu, Liang Liao, Wenxiu Sun, Qiong Yan, and Weisi Lin. Enhancing diffusion models with text-encoder reinforcement learning. In *European Conference on Computer Vision*, pp. 182–198. Springer, 2024.

Kevin Clark, Paul Vicol, Kevin Swersky, and David J Fleet. Directly fine-tuning diffusion models on differentiable rewards. *arXiv preprint arXiv:2309.17400*, 2023.

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

Luca Eyring, Shyamgopal Karthik, Karsten Roth, Alexey Dosovitskiy, and Zeynep Akata. Reno: Enhancing one-step text-to-image models through reward-based noise optimization. *Advances in Neural Information Processing Systems*, 37:125487–125519, 2024.

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. *Advances in Neural Information Processing Systems*, 36, 2024.

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.

Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and S Qian Zhang. Parameter-efficient fine-tuning for large models: A comprehensive survey. arxiv 2024. *arXiv preprint arXiv:2403.14608*, 10, 2024.

Yaru Hao, Zewen Chi, Li Dong, and Furu Wei. Optimizing prompts for text-to-image generation. *Advances in Neural Information Processing Systems*, 36, 2024.

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

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

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

Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

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

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021.

Xiaomin Li, Yixuan Liu, Takashi Isobe, Xu Jia, Qinpeng Cui, Dong Zhou, Dong Li, You He, Huchuan Lu, Zhongdao Wang, et al. Reneg: Learning negative embedding with reward guidance. *arXiv preprint arXiv:2412.19637*, 2024a.

Yanyu Li, Xian Liu, Anil Kag, Ju Hu, Yerlan Idelbayev, Dhritiman Sagar, Yanzhi Wang, Sergey Tulyakov, and Jian Ren. Textcraftor: Your text encoder can be image quality controller. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7985–7995, 2024b.

Zhanhao Liang, Yuhui Yuan, Shuyang Gu, Bohan Chen, Tiankai Hang, Ji Li, and Liang Zheng. Step-aware preference optimization: Aligning preference with denoising performance at each step. *arXiv preprint arXiv:2406.04314*, 2024.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 740–755. Springer, 2014.

Buhua Liu, Shitong Shao, Bao Li, Lichen Bai, Zhiqiang Xu, Haoyi Xiong, James Kwok, Sumi Helal, and Zeke Xie. Alignment of diffusion models: Fundamentals, challenges, and future. *arXiv preprint arXiv:2409.07253*, 2024.

Vivian Liu and Lydia B Chilton. Design guidelines for prompt engineering text-to-image generative models. In *Proceedings of the 2022 CHI conference on human factors in computing systems*, pp. 1–23, 2022.

Oscar Mañas, Pietro Astolfi, Melissa Hall, Candace Ross, Jack Urbanek, Adina Williams, Aishwarya Agrawal, Adriana Romero-Soriano, and Michal Drozdzal. Improving text-to-image consistency via automatic prompt optimization. *arXiv preprint arXiv:2403.17804*, 2024.

Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

Jonas Oppenlaender. A taxonomy of prompt modifiers for text-to-image generation. *Behaviour & Information Technology*, pp. 1–14, 2023.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

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

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*, pp. 8748–8763. PMLR, 2021.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

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*, pp. 10684–10695, 2022.

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.

C Schuhmann. Laoin aesthetic predictor. `https://laion.ai/blog/laion-aesthetics/`, 2024.

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

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

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

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

Ruochen Wang, Ting Liu, Cho-Jui Hsieh, and Boqing Gong. On discrete prompt optimization for diffusion models. *arXiv preprint arXiv:2407.01606*, 2024.

Yunlong Wang, Shuyuan Shen, and Brian Y Lim. Reprompt: Automatic prompt editing to refine ai-generative art towards precise expressions. In *Proceedings of the 2023 CHI conference on human factors in computing systems*, pp. 1–29, 2023.

Zijie J Wang, Evan Montoya, David Munechika, Haoyang Yang, Benjamin Hoover, and Duen Horng Chau. Diffusiondb: A large-scale prompt gallery dataset for text-to-image generative models. *arXiv preprint arXiv:2210.14896*, 2022.

Yuxin Wen, Neel Jain, John Kirchenbauer, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Hard prompts made easy: Gradient-based discrete optimization for prompt tuning and discovery. *Advances in Neural Information Processing Systems*, 36, 2024.

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

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36, 2024.

Kai Yang, Jian Tao, Jiafei Lyu, Chunjiang Ge, Jiaxin Chen, Weihan Shen, Xiaolong Zhu, and Xiu Li. Using human feedback to fine-tune diffusion models without any reward model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8941–8951, 2024.

Chenshuang Zhang, Chaoning Zhang, Mengchun Zhang, and In So Kweon. Text-to-image diffusion models in generative ai: A survey. *arXiv preprint arXiv:2303.07909*, 2023.

A Optimization of Rotation Parameterizations

We show the intuition how the rotations help accelerate the optimization process. First, let us consider the optimization in the two-dimensional space. Suppose we have a minimization problem

$$\text{argmin}_x f(x), \quad x \in \mathbb{R}^2. \tag{6}$$

We assume this problem only has one global minimum x^*, which therefore lies in the subspace spanned by itself. Now we parameterize $x = R_{e,\theta} y$, with $y \in \mathbb{R}^2$ and $R_{e,\theta}$ the elementary rotation matrix in equation 4. We update parameters step by step. We initialize x_0 by $\theta_0 = 0$ and $y_0 = 0$ at the origin. A gradient update step moves x_0 along the gradient of x_0, with a suitable step size, to x_1, with θ_0 unchanged. Assume we are currently at $x_t = R_{e,\theta_t} y_t$. We update θ_t by solving θ_{t+1} from:

$$\nabla_x f(x_t)^T \frac{dR}{d\theta}\bigg|_{\theta_{t+1}} y_t = 0. \tag{7}$$

Note $\frac{dR}{d\theta}\big|_{\theta_{t+1}} = R_{e,\frac{\pi}{2}} R_{e,\theta_{t+1}}$, the elementary rotation matrix with angle θ_{t+1} rotated 90 degrees counterclockwise. Therefore, graphically, the optimal θ_{t+1} is the one that rotates y_t, with the origin as the rotation axis, to a point such that the vector pointing to it is parallel to the gradient at that point. In other words, this is the point where the circle with radius $\|x_t\|$ is tangent to the contour of f. Then for any suitable step sizes along the corrected gradient towards the new point x_{t+1}, the total path length between the origin x_0 and the optimal point x_* is equal to the distance $\|x_*\|_2$, which is the shortest path between the initial point and the optimal point, and therefore **optimal** among all possible paths between the initial point to the optimum point. Figure 3 visualizes the argument. The left panel shows an optimization path with rotation, which makes the total path length be exactly equal to the shortest path (the purple line) since the updated points are selected at the tangent point between the circles (red and dashed) and the contour. However, there is no guarantee that a regular gradient descent takes that shortest path, as illustrated on the right panel.

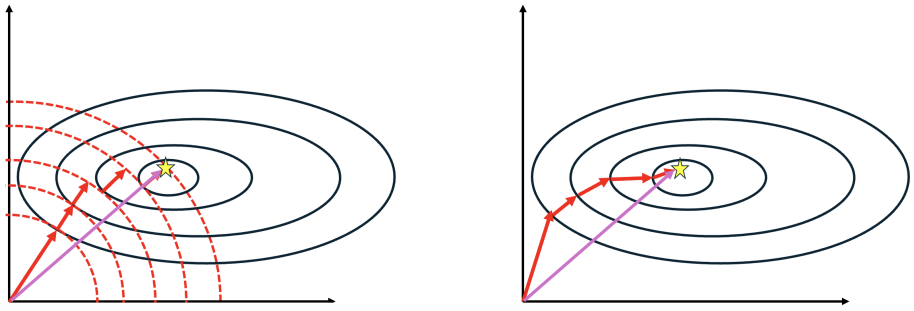


Figure 3: The figure compares the optimizations with rotation parametrization (left) and without (right). The yellow star represents the optimum point. The red dashed lines on the left are the circles with radius of the length of the current x_t. The purple line is the shortest distance between the initial point, the origin, and the optimum point. With rotation parametrization, the updates must be along the shortest total path.

In a high-dimensional space, since IPGO rotates each neighboring pair of coordinates, the high-dimensional rotation can be disassembled into separate 2-dimensional rotations. Therefore, our argument above in the 2D space can be extended to the high-dimensional space. In other words, the high-dimensional rotation should guide a relatively shorter optimization path towards the optimal solution. Nonetheless, one difference should be noted. In our IPGO algorithm, because the rotation angles to be optimized are shared across all neighboring pairs of coordinates, overall the rotation component of the IPGO will select the angle that on average benefits the optimization path the most. The optimization along the average path could lead to less efficient optimization updates in some of the 2D subspaces, which may therefore require more optimization updating steps for convergence. This effect can be observed in our ablation studies on rotation for CLIP alignment, shown in Table 4.

B Qualitative Comparisons between IPGO and Baselines

Table 7 qualitatively compares our IPGO algorithm with the benchmarking algorithms outlined in Section 5.2. As can be seen from the table, IPGO is the only method that leverages reward gradient computation, supports prompt modification, and maintains low computational search costs.

Algorithms	*Reward Gradient*	*Prompt Modification*	*Search Space Cost*
Promptist	✗	✓	High (SFT required)
DPO-Diff	✓	✓	High (External LLM required)
TextCraftor	✓	✓	High (Text Encoder Fine-Tuning)
DRaFT	✓	✗	Low
DDPO	✗	✗	High (Many samples required)
IPGO (ours)	✓	✓	Low

Table 7: A qualitative comparison between IPGO and all baselines, focusing on three key aspects: the ability to compute reward gradients, support for prompt modification, and the computational cost of searching the prompt space.

C Implementation Details

This section provides details on IPGO's implementation and training.

Image Generation A DDIM Scheduler with a guidance weight of 7.5 is employed and the generated images have a resolution of 512×512 pixels. During optimization, IPGO truncates the backpropagation at the 2nd-to-last sampling step. we set the number of inference step for image generation as 50. We found similar performances with other sampling strategies, such as PNDM and LMSD.

Optimization We train IPGO using Adam (Kingma, 2014) optimizer without a weight decay. We start with a learning rate of $1\mathrm{e}{-3}$ and reduce it by a factor of 0.9 every 10 epochs, continuing this schedule for a total of 50 epochs. We truncate gradients at the 2nd-to-last step with checkpointing. We apply gradient clipping across all our experiments, selecting a gradient clipping norm of $c = 1.0$.

Hyperparameters We set the hyperparameters of IPGO, DDPO, and DRaFT-1 to ensure that the total number of trainable parameters is comparable. IPGO includes the hyperparameters: $m_{\text{pre}}(m_{\text{suf}})$, the number of the base text embeddings for prefix (suffix); and $N_{\text{pre}}(N_{\text{suf}})$, the length of the prefix (suffix). DRaFT-1 uses the LoRA parameters in the UNet. For DDPO and TextCraftor we use the default configurations. Detailed hyperparameter settings for IPGO, TextCraftor, DRaFT-1 and DDPO are provided in Table 8.

Methods	*Hyperparameter*	*Value*
IPGO	$m_{\text{pre}}, m_{\text{suf}}$	300
	$N_{\text{pre}}, N_{\text{suf}}$	10
Total #parameters		0.47M
TextCraftor	Default Configuration	
Total #parameters		123M
DRaFT-1	LoRA rank	3
Total #parameters		0.60M
DDPO	Default DDPO Trainer Configuration	
Total #parameters		0.79M

Table 8: Hyperparameter settings for IPGO, TextCraftor, DRaFT-1 and DDPO

IPGO's Constraints IPGO has three constraints: Orthonormality, Value and Conformity constraints. We enforce the orthonormality constraint with `orthogonal()` module in Pytorch. For the value constraint, we clamp the parameters to satisfy the constraint after each update. Finally, we use a soft conformity constraint in the optimization by adding a conformity penalty to the objective, the negative image reward. Define the conformity penalty by

$$P_{\text{conf}} = \|\textit{mean}(\mathcal{E}(V_{\text{pre}}, p, V_{\text{suf}};\ \Omega_{\text{IPGO}})) - \textit{mean}(\mathcal{T}(p))\|_2^2. \tag{8}$$

The $\textit{mean}(\cdot)$ is defined by,

$$\textit{mean}(\{\mathbf{v}_i\}_{i=1}^L) = \frac{1}{L}\sum_{i=1}^{L}\mathbf{v}_i, \tag{9}$$

where $\{\mathbf{v}_i\}_{i=1}^L$, $\mathbf{v}_i \in \mathbb{R}^d$, is the input set of text embeddings.

Then the optimization loss to be minimized, conditioned on x_0, becomes:

$$\mathcal{L}(\Omega_{\text{IPGO}}) = -\mathcal{S}\left(x_0, p\right) + \gamma P_{\text{conf}}, \tag{10}$$

where $\mathcal{S}(x_0, p)$ is one of the Aesthetic, CLIP and HPSv2 reward scores as a function of the image x_0 and prompt p, and γ is the conformity coefficient. In our experiments we set $\gamma = 1\text{e}{-3}$.

The Outline of Our IPGO Algorithm Here we delineate the algorithm of the full IPGO with all constraints and parameterization designs included.

Algorithm 1 IPGO

Input: Raw prompt p, prefix/suffix generator $G_{\text{pre}}(\Omega_{\text{IPGO}})$/$G_{\text{suf}}(\Omega_{\text{IPGO}})$ controlled by Ω_{IPGO}, text encoder $\mathcal{T}(p)$, diffusion model $x \sim q_{\text{image}}(\cdot)$, z_T the initial latent noise, image reward model $S(x, p)$, conformity penalty coefficient γ, learning rate η, number of epochs $Epochs$.
Output: Optimal prefix/suffix generators G°_{pre}/G°_{suf}.
for $i = 0$ **to** $Epochs$ **do**
  Original prompt embedding: $V_0 = \mathcal{T}(p)$.
  Prefix: $V_{\text{pre}} = G_{\text{pre}}(\Omega_{\text{IPGO}})$
  Suffix: $V_{\text{suf}} = G_{\text{suf}}(\Omega_{\text{IPGO}})$
  Insert prefix and suffix: $\mathcal{E}(V_{\text{pre}}, p, V_{\text{suf}};\ \Omega_{\text{IPGO}}) = V_{\text{pre}} \oplus V_0 \oplus V_{\text{suf}}$.
  Sample image: $x_0 \sim q_{\text{image}}(x_0|\mathcal{E}(V_{\text{pre}}, p, V_{\text{suf}};\ \Omega_{\text{IPGO}}), z_T)$.
  Compute reward: $r = \mathcal{S}(x_0, p)$.
  Compute objective: $\mathcal{L} = -r + \gamma P_{\text{conf}}$.
  Compute gradient: $g = \nabla_{\Omega_{\text{IPGO}}}\mathcal{L}$.
  Update prefix and suffix: $\Omega_{\text{IPGO}} \leftarrow \Omega_{\text{IPGO}} - \eta g$.
  Enforce orthonormality and Value constraints.
end for
Return $G^{\circ}_{\text{pre}}(\Omega_{\text{IPGO}})$ and $G^{\circ}_{\text{suf}}(\Omega_{\text{IPGO}})$.

D Additional Ablation Studies

In addition to the ablation experiments in the main text, we design two more ablation scenarios to investigate the effects of the size of the base text embeddings and the relationship between the prefix/suffix lengths and the raw prompt length.

For both additional ablations, following the settings in the previous ablations, we use the Stable Diffusion v1.5 as the base diffusion model, with number of inference steps 30. The Adam optimization starts with a learning rate $1\text{e}{-3}$ with a decay factor 0.9 at every 10 steps.

Varying $m = m_{\text{pre}} = m_{\text{suf}}$. We first investigate the effect of m_*, the size of the learnable base text embeddings for the prefix and suffix. We randomly selected 30 prompts from the COCO dataset, and we conduct ablation studies with $m = 150, 300, 600$, which are about 20%, 40% and 80% of the text embedding space of dimension 768. As reward models, we choose the CLIP reward and the human preference reward (HPS). The prefix and suffix lengths are both 10.

Reward	$m = 150$	$m = 300$	$m = 600$
CLIP	0.287	0.296	0.303
HPS	0.263	0.266	0.267

Table 9: Results of ablations on m_{pre} and m_{suf}, the sizes of the sets of the base text embeddings of prefix and suffix.

Table 9 shows the results. Not surprisingly, more parameters lead to larger performance gains, shown for both rewards. However, it is interesting to see a diminishing margin of performance gain when we increase m. The performance improvement from increasing m from 300 to 600 is less substantial than the improvement gained by increasing m from 150 to 300. Therefore, $m = 300$ achieves a good balance between the number of total parameters and the final performance.

N_{pre} and N_{suf} based on Raw Prompt Length. Next, we test the relationship between the length of the raw prompt and the lengths of prefix and suffix. We use the CLIP reward for optimizations. We choose 30 prompts among which the first 10 prompts are simple prompts, such as "Man", "Woman" and "Student", the second 10 prompts are medium-complexity prompts of the similar topics of the 10 simple prompts, selected from the COCO dataset. The last 10 prompts are even more complex versions of the second 10 prompts by inquiring ChatGPT with "Could you make the following 10 prompts more complex:." For example, the complex version of "A person walking in the rain while holding an umbrella." is "A middle-aged person in a long, tattered trench coat walks down a cobblestone street, their brightly colored umbrella catching the dim glow of streetlights as rain cascades around them." We make sure that the complex sentences do not exceed the limit of 77 tokens.

We optimize each prompt with $N_{\text{pre}} = N_{\text{suf}} = N \in \{2, 10, 15\}$ with respect to the CLIP reward. For each prompt, we record the value of N that improves the output image the most. Then we count the frequencies of each $N \in \{2, 10, 15\}$ and calculate their proportions. Finally, we use this distribution of the proportions of prompts that respectively have $N = 2, 10, 15$ as their best prefix and suffix lengths in each prompt group as the evaluation metric. We denote this distribution as $D_N(2, 10, 15)$: $D_N(2, 10, 15) = (a\%, b\%, c\%)$ means that $a\%$ (or $b\%$ or $c\%$) of the prompts in the target prompt group have $N = 2$ (or $N = 10$ or $N = 15$) as their best prefix/suffix lengths.

From the ablation results, *we do not observe a significant correlation between the prompt length and the prefix and suffix lengths.* The simple prompt group has $D_N(2, 10, 15) = (30\%, 50\%, 20\%)$; the medium-complex prompt group has $D_N(2, 10, 15) = (50\%, 20\%, 30\%)$; and the complex prompt group has $D_N(2, 10, 15) = (20\%, 40\%, 40\%)$. We find no correlation between the raw prompt length and the optimal prefix-suffix length. However, we recommend using fewer inserted embeddings for very short prompts to avoid over-parameterization (an illustration follows).

E Overparameterization

IPGO faces the risk of overparameterization when the lengths of the prefix and suffix significantly exceed that of the raw prompt. For example, Figure 4 illustrates this issue with an extreme example, showcasing the evolution of images generated during the optimization of the simple prompt "cat", which only has one single token, with very long prefix and suffix of $N_{\text{pre}} = N_{\text{suf}} = 30$ for aesthetics improvement. In the first several steps, IPGO produces images that display a cat, but at later steps, the object in the image changes to a person, and at even later steps the specific person also changes. Apparently, if the prefix and suffix are too long, then in spite of the conformity constraint, optimization of the inserted embeddings overwhelms the semantic structure of the image and harms alignment of the image with the *original* prompt. Therefore, we recommend using shorter prefix and suffix lengths for shorter prompts. An alternative solution is to extend IPGO to multi-criterion optimization, using both aesthetics and prompt-image alignment rewards.

F Societal Impact

This paper presents work that contributes to the field of Text-to-Image generation models and their applications. In the Machine Learning community, the new method introduced by this paper can

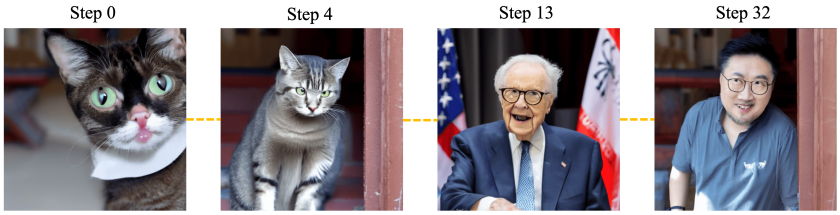


Figure 4: Images generated during IPGO's optimization on the prompt "Cat" with $N_{\text{pre}} = N_{\text{suff}} = 30$ for aesthetics. Because of overparameterization the images that are produced show poor alignment with the *original* prompt.

broaden the current horizon on fine-tuning diffusion models. In practice, our method can be applied to image related tasks such as automatic real-time image editing.

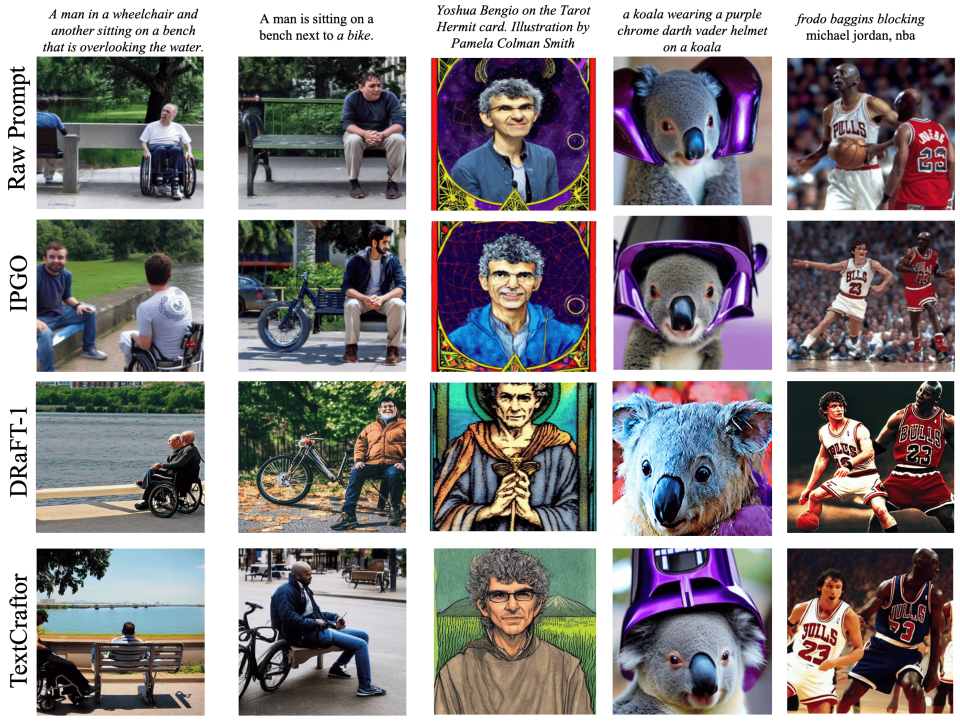
G LLM USAGE

Large Language Models (LLMs) were used in limited capacities to support this work. Specifically, they were only employed for grammar correction and language polishing during manuscript preparation. Additionally, LLMs were used to assist in additional ablation studies in Appendix D by generating diverse image prompts. No parts of the scientific analysis, experimental design, or core findings relied on LLM outputs.

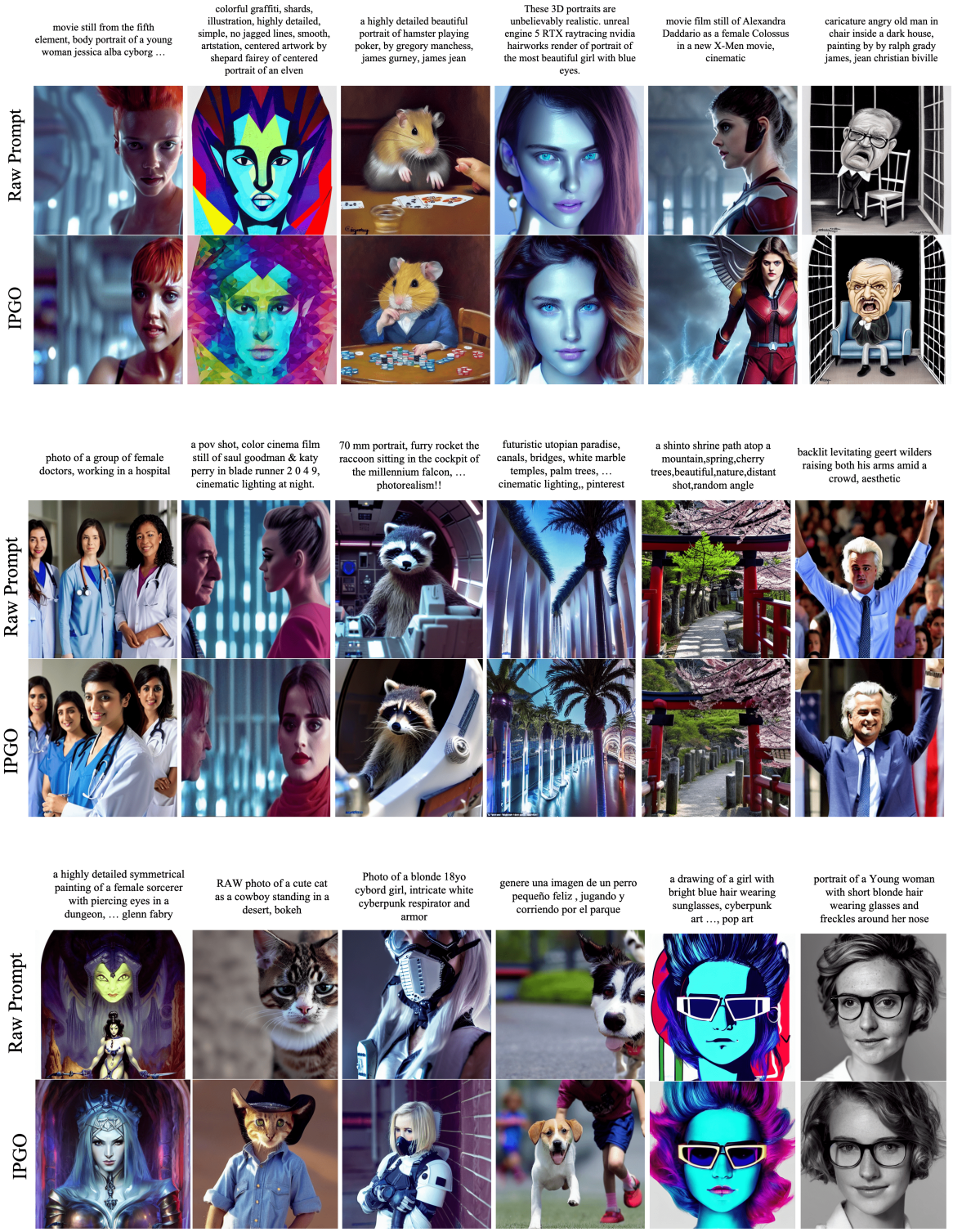
H ADDITIONAL IMAGE EXAMPLES

All optimized images shown in this section were optimized with respect to the human preference reward HPSv2.

H.1 ADDITIONAL IPGO COMPARISONS WITH DRAFT-1 AND TEXTCRAFTOR



H.2 OTHERS


movie still from the fifth element, body portrait of a young woman jessica alba cyborg …
colorful graffiti, shards, illustration, highly detailed, simple, no jagged lines, smooth, artstation, centered artwork by shepard fairey of centered portrait of an elven
a highly detailed beautiful portrait of hamster playing poker, by gregory manchess, james gurney, james jean
These 3D portraits are unbelievably realistic. unreal engine 5 RTX raytracing nvidia hairworks render of portrait of the most beautiful girl with blue eyes.
movie film still of Alexandra Daddario as a female Colossus in a new X-Men movie, cinematic
caricature angry old man in chair inside a dark house, painting by by ralph grady james, jean christian biville
Raw Prompt
IPGO
photo of a group of female doctors, working in a hospital
a pov shot, color cinema film still of saul goodman & katy perry in blade runner 2 0 4 9, cinematic lighting at night.
70 mm portrait, furry rocket the raccoon sitting in the cockpit of the millennium falcon, … photorealism!!
futuristic utopian paradise, canals, bridges, white marble temples, palm trees, … cinematic lighting,, pinterest
a shinto shrine path atop a mountain,spring,cherry trees,beautiful,nature,distant shot,random angle
backlit levitating geert wilders raising both his arms amid a crowd, aesthetic
Raw Prompt
IPGO
a highly detailed symmetrical painting of a female sorcerer with piercing eyes in a dungeon, … glenn fabry
RAW photo of a cute cat as a cowboy standing in a desert, bokeh
Photo of a blonde 18yo cybord girl, intricate white cyberpunk respirator and armor
genere una imagen de un perro pequeño feliz , jugando y corriendo por el parque
a drawing of a girl with bright blue hair wearing sunglasses, cyberpunk art …, pop art
portrait of a Young woman with short blonde hair wearing glasses and freckles around her nose
Raw Prompt
IPGO

H.3 More Visual Comparisons

For each image pair, the top image is generated by SDv1.5, the bottom image is optimized by IPGO.