PROMPTMINER: Black-Box Prompt Stealing against Text-to-Image Generative Models via Reinforcement Learning and Fuzz Optimization

Mingzhe Li[1], **Renhao Zhang**[1], **Zhiyang Wen**[1], **Siqi Pan**[2]
Bruno Castro da Silva[1], **Juan Zhai**[1], **Shiqing Ma**[1]
[1]University of Massachusetts, Amherst [2]Dolby Laboratories

Abstract

Text-to-image (T2I) generative models such as Stable Diffusion and FLUX can synthesize realistic, high-quality images directly from textual prompts. The resulting image quality depends critically on well-crafted prompts that specify both subjects and stylistic modifiers, which have become valuable digital assets. However, the rising value and ubiquity of high-quality prompts expose them to security and intellectual-property risks. One key threat is the ***prompt stealing attack****, i.e., the task of recovering the textual prompt that generated a given image. Prompt stealing enables unauthorized extraction and reuse of carefully engineered prompts, yet it can also support beneficial applications such as data attribution, model provenance analysis, and watermarking validation. Existing approaches often assume white-box gradient access, require large-scale labeled datasets for supervised training, or rely solely on captioning without explicit optimization, limiting their practicality and adaptability. To address these challenges, we propose* **PROMPTMINER***, a black-box prompt stealing framework that decouples the task into two phases: (1) a reinforcement learning–based optimization phase to reconstruct the primary subject, and (2) a fuzzing-driven search phase to recover stylistic modifiers. Experiments across multiple datasets and diffusion backbones demonstrate that* PROMPTMINER *achieves superior results, with CLIP similarity up to 0.958 and textual alignment with SBERT up to 0.751, surpassing all baselines. Even when applied to in-the-wild images with unknown generators, it outperforms the strongest baseline by 7.5% in CLIP similarity, demonstrating better generalization. Finally,* PROMPTMINER *maintains strong performance under defensive perturbations, highlighting remarkable robustness. Code:* https://github.com/aaFrostnova/PromptMiner

1. Introduction

Text-to-image (T2I) generative models such as *Stable Diffusion* [33] *FLUX* [12], and *DALL·E* [31] have become widely used in our daily lives, due to their excellent performance in synthesizing realistic and creative images from descriptive prompts. A key factor in successfully generating high-quality images is careful prompt design. Well-crafted prompts often encode sensitive details and typically combine precise *subject* (core visual entity) with rich *modifiers* (stylistic and compositional cues) [6, 18]. However, obtaining high-quality prompts is far from trivial. Crafting an effective prompt requires substantial domain knowledge and iterative experimentation, as minor changes in phrasing, token order, or modifier choice can dramatically alter the generated image's composition. Moreover, such well-engineered prompts have become valuable digital assets—carefully guarded by creators and even traded on specialized marketplaces such as PromptBase [26] and PromptHero [27], highlighting the growing economic and creative significance of prompt design in the generative AI ecosystem.

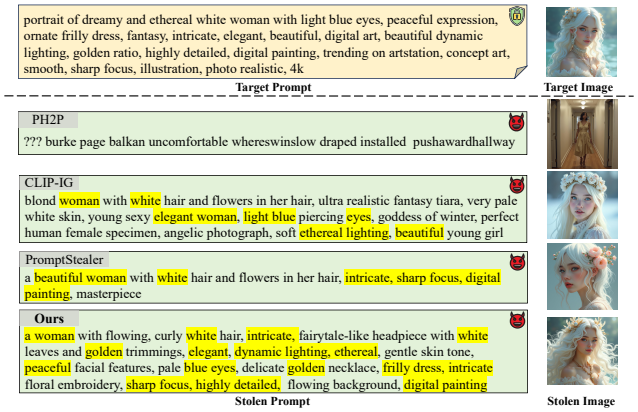


Figure 1. Illustration of the prompt stealing. Given a target image, the attacker aims to recover the prompt to generate a similar image.

The growing value and ubiquity of high-quality prompts expose them to security and intellectual-property threats, most notably prompt stealing attacks or prompt inversion methods [11, 16, 19, 37]. In such attacks, adversaries aim to recover the most precise descriptive prompt to generate the target image as closely as possible. Beyond infringing intellectual property, an attacker can resell the stolen prompt on third-party marketplaces, directly monetizing others' work. This practice erodes the commercial interests of both plat-

forms and original creators. At the same time, reconstructing prompts from generated images also serves as a beneficial forensic tool for trustworthy AI, enabling model provenance analysis and watermark validation [22, 41, 42].

Currently, there is a wide array of cutting-edge prompt stealing or prompt inversion methods. First, white-box prompt inversion methods [19, 44] mostly focus on the subject and often yield linguistically awkward prompts. Crucially, the white-box assumption is rarely valid in commercial deployments, limiting practicality. In contrast, black-box methods like BLIP [14] and VGD [11] achieve prompt recovery by combining existing components such as CLIP, image captioning models, or large language models. However, they similarly lack the ability to reconstruct modifiers. CLIP-IG [25] and PromptStealer [37] consider both the subject and modifiers when recovering prompts. However, they lack an explicit optimization process, relying instead on direct generation from pretrained models. This direct generation often lacks similarity in image generation. Specifically, PromptStealer trains an inversion model for subject generation and a classification model for modifier prediction on large-scale, labeled datasets (i.e., Lexica [37]). This dataset-specific training also introduces a strong risk of overfitting, limiting the model's generalization ability across diverse images.

To overcome these limitations, we propose PROMPTMINER, a novel and effective black-box prompt stealing framework that achieves both semantic precision and stylistic fidelity without requiring model gradients or training on large-scale, labeled datasets. The core insight is to decouple prompt stealing into (i) a reinforcement learning-based optimization phase to reconstruct the primary subject, and (ii) a fuzzing-driven search phase to recover stylistic modifiers. Unlike prior white-box methods that depend on gradient access, PROMPTMINER first leverages reinforcement learning (RL) [39] to perform black-box optimization, precisely capturing the core visual entities and yielding concise, content-faithful subjects. Specifically, we formulate the prompt inversion as a Markov decision process (MDP) and solve it using RL with a shaped reward for faster convergence. Then, to enrich the recovered prompt with modifiers, we introduce a fuzz testing-powered optimization phase that treats modifier discovery as controlled exploration in prompt space: carefully designed mutators propose targeted edits, while a capacity-constrained seed pool maintains and exploits the most promising candidates. Together, these two phases enable efficient discrete prompt space search, preserving semantic grounding while systematically injecting modifiers crucial for high-fidelity image synthesis. As shown in Figure 1, PROMPTMINER outperforms baseline methods by producing more accurate prompts and generating images most similar to the target.

We conduct a comprehensive experimental assessment of PROMPTMINER. Our results on the three widely-used datasets (e.g., MS COCO [17], Flickr [45], and Lexica [36]) under four state-of-the-art T2I generation model (e.g., Stable Diffusion v1.5 [33], SDXL-Turbo [34], FLUX.1 dev [12], and Stable Diffusion 3.5 Medium [3]) show that PROMPTMINER is more effective that baselines with CLIP similarity up to 0.958 and textual alignment with SBERT up to 0.751. Moreover, PROMPTMINER also outperforms the strongest baseline by 7.5% in CLIP similarity on *in-the-wild* images where the target generative models are unknown, demonstrating its strong generalization ability. We further evaluate the robustness of PROMPTMINER under several potential defense strategies, including random noise injection, patch perturbation, and textual watermarking. These post-processing defenses exert only marginal influence on our performance, indicating that PROMPTMINER is inherently resilient to low-level visual perturbations due to its semantics-driven optimization process. Our contributions are summarized as follows:

- We introduce PROMPTMINER, a novel and effective black-box prompt stealing framework against text-to-image generative models. Unlike existing methods that assume white-box gradient access, rely on direct caption generation without explicit optimization, or depend on large, labeled datasets with poor adaptability, PROMPTMINER efficiently refines prompts, delivering precise and expressive reconstructions without relying on model gradients or large-scale labeled datasets.

- We formulate prompt stealing as an MDP and optimize an adapter on top of a frozen captioner with reinforcement learning, using potential-based reward shaping for dense guidance and a fuzz testing-powered optimization phase to discover effective modifiers. This design yields concise, precise subjects and systematically injects style and composition cues.

- We conduct extensive evaluations across three widely-used datasets and four advanced text-to-image generative models, where PROMPTMINER achieves state-of-the-art performance in both image similarity and textual alignment. PROMPTMINER further generalizes to *in-the-wild* images (unknown target models) and remains robust under common post-processing defenses.

2. Related Work

Image Captioning. A straightforward way to obtain prompts is to apply captioning models or advanced vision language models (VLMs), such as BLIP [14], BLIP-2 [15], or GPT-4o [9]. Given an input image x, a captioning model or a VLM generates a textual description as the prompt in

an autoregressive manner:

$$P(p_{0:T} \mid x) = \prod_{t=0}^{T} P(p_t \mid p_{<t}, x), \tag{1}$$

where p_t denotes the t-th token, T is the sequence length determined when the model emits an end-of-sequence token, and $p_{0:T} = \{p_0, \ldots, p_T\}$ represents the complete prompt. At each step t, the model conditions on both the visual features of x and the textual prefix $p_{<t}$ to predict the distribution over the next token. While these models generate fluent descriptions, they fail to ensure high similarity when re-generating the image with a diffusion model.

Prompt Stealing Attack and Prompt Inversion. Recent gradient-based works [16, 19, 44] focus on optimizing textual prompts to better align with image features. PEZ [44] introduces a gradient-based discrete optimization method for hard prompt discovery with CLIP [20], and PH2P [19] proposes hard prompt optimization that aligns prompts with image features directly in diffusion models. However, these inverted prompts often lack semantic fluency and naturalness, resulting in outputs unintelligible to humans. Another recent work, Visually Guided Decoding (VGD) [11], is gradient-free and uses a large language model plus CLIP-based guidance to generate coherent, human-readable prompts. Nevertheless, VGD primarily focuses on identifying a single salient subject and overlooks the richer modifier details. Building on findings from [18], effective prompts typically follow a *"Subject, Modifiers"* structure, where modifiers capture nuanced visual semantics beyond the main entity. In contrast, CLIP Interrogator [25] and PromptStealer [37] extend the generation process by explicitly adding descriptive modifiers from a predefined pool or by decomposing prompts into subject and modifier components. However, both approaches lack a targeted optimization process that refines the composed prompts toward the specific image. Moreover, PromptStealer relies on large-scale labeled datasets for supervised training, which limits its scalability and adaptability to unseen domains. Similarly, Prometheus [50] depends heavily on an extensive prompt base constructed from pre-collected prompt data to guide its modifier search, making it less flexible in the open world.

3. Methodology

3.1. Threat Model

Adversary's Goal. Given a target image generated by a text-to-image generative model, the goal of prompt stealing is to invert the prompt used to generate the target image or a prompt that leads to highly similar images. As such, the attack is typically evaluated by (1) image similarity between the generated image and the target image, including perceptual and semantic similarities; (2) prompt similarity (or textual alignment) between the inverted and original prompts, measuring the semantic similarity at token- and sentence-levels.

Adversary's Capability. We assume that the adversary is only provided with a target image produced by a text-to-image generative model and has black-box access to the target text-to-image generative model. Additionally, the adversary does not have access to large-scale image-caption datasets or to prompt base with commercially traded prompts.

Defender's Capability. The defenders are primarily focused on protecting high-quality prompts from being stolen. We assume they can apply post-hoc transformations to images generated with protected prompts before those images are disseminated.

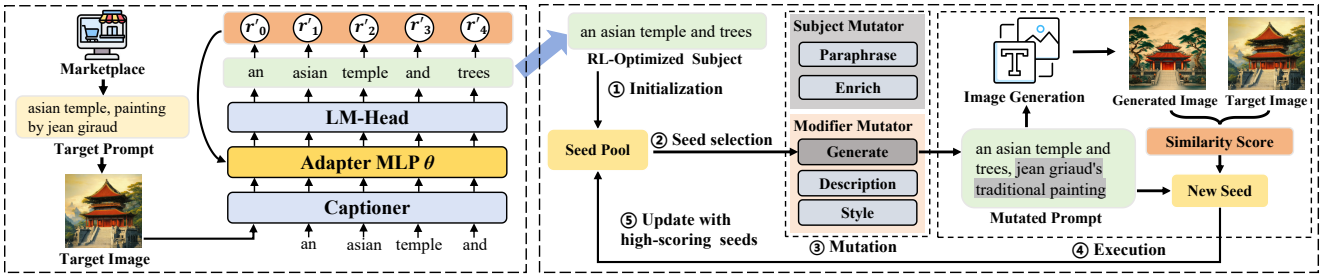
3.2. Overview

We introduce an effective prompt stealing attack against text-to-image generative models called PROMPTMINER, consisting of two main phases. In the *phase I*, we formulate the prompt inversion as a Markov Decision Process (MDP) and solve it using reinforcement learning with shaped reward. In the *phase II*, we perform a fuzz testing-powered Prompt Optimization to iteratively refine the RL-derived prompt, exploring richer descriptions and stylistic modifiers while maximizing similarity between generated and target images. The overview of PROMPTMINER is shown in Figure 2.

3.3. RL-Based Prompt Inversion

Recalling the autoregressive prompt generation in image captioning models, at each step, the model observes the current partially constructed prompt and a representation of the target image, chooses the next token to append. We can formulate this process as an MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$ and solve it using reinforcement learning [39]. The **state** $s_t \in \mathcal{S}$ is defined as $s_t = \{p_{0:t-1}, x\}$, where $p_{0:t-1}$ is the partial prompt and x is the target image. The **action** $a_t \in \mathcal{A}$ selects the next token p_t from the vocabulary; and the **transition** $\mathcal{P}(s_{t+1} \mid s_t, a_t)$ deterministically appends the selected token to form the new prefix. The γ is the **discount factor**. The episode terminates when an end-of-sequence token is produced, yielding a complete prompt $p_{0:T}$ and a synthesized image $\hat{x}$, T is the sequence length determined when the model emits an end-of-sequence token. The objective is to learn a policy $\pi_\theta(a_t \mid s_t)$ that maximizes the expected discounted return $\mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{T} \gamma^t R(s_t, a_t)\right]$.

Reward Design. Since our goal is to recover a prompt whose synthesized image $\hat{x}$ is as close as possible to the target image x, a naive design of **reward** $\mathcal{R}$ is to measure the similarity between the generated image $\hat{x}$ and the target image x directly. In specific, after generating the complete prompt $p_{0:T}$, we synthesize an image $\hat{x}$ with the target text-to-image generative model and compute the reward $r_t =$



Phase I: **RL-Based Prompt Inversion** ***Phase II:*** **Fuzz Testing-Powered Prompt Optimization**

Figure 2. Overview of PROMPTMINER. Our method comprises two phases: (*I*) a reinforcement learning–based optimization phase to reconstruct the primary subject, and (*II*) a fuzzing-driven search phase to recover stylistic modifiers.

$\mathcal{R}(s_t, a_t)$ as:

$$r_t = \begin{cases} 0, & 0 \le t < T, \\ \Psi(x, \hat{x}), & t = T, \end{cases} \tag{2}$$

where $\Psi(\cdot)$ is the cosine similarity between the generated image $\hat{x}$ and the target image x. It is calculated with an image encoder $f_{\text{img}}(\cdot)$ from CLIP as:

$$\Psi(\hat{x}, x) = \frac{f_{\text{img}}(\hat{x}) \cdot f_{\text{img}}(x)}{\|f_{\text{img}}(\hat{x})\| \, \|f_{\text{img}}(x)\|}. \tag{3}$$

We invoke the target text-to-image generative model only after a complete prompt has been produced, so no environment feedback is available at intermediate steps, and we set $r_t = 0$ for $0 \le t < T$.

However, such reward r_t based only on image similarity leads to sparse supervision, as the agent receives feedback only at the end of each episode. This makes optimization inefficient and unstable, especially for long prompts with large action spaces. To address this, we utilize **potential-based reward shaping [23]** to provide additional intermediate rewards r'_t for denser learning signals. The overall **shaped reward** $r'_t = \mathcal{R}'(s_t, a_t)$ is defined as:

$$r'_t = \begin{cases} \gamma\Phi(s_{t+1}) - \Phi(s_t), & 0 \le t < T, \\ r_t - \Phi(s_t), & t = T, \end{cases} \tag{4}$$

where $\Phi(\cdot)$ is the potential function. Specifically, we use CLIP text-image similarity to define it as:

$$\Phi(s_t) = \beta \cdot \frac{f_{\text{text}}(p_{1:t}) \cdot f_{\text{img}}(x)}{\|f_{\text{text}}(p_{1:t})\| \, \|f_{\text{img}}(x)\|}, \tag{5}$$

where β is a scaling coefficient, $f_{\text{text}}(\cdot)$ and $f_{\text{img}}(\cdot)$ are the CLIP text and image encoders that map prompts and images into a shared embedding space.

This reward design integrates dense potential-based shaping with structured terminal feedback, resulting in faster convergence and more stable optimization compared to sparse end-only rewards. Theoretically, as shown in Section K, the potential-based formulation preserves the optimal policy, ensuring that shaping affects only the learning dynamics rather than the underlying objective. Empirically, as shown in Figure 5, the proposed potential-based reward shaping accelerates convergence and improves training efficiency while maintaining stability.

Model Architecture. As shown in Figure 2, we build our agent on top of a frozen captioner. Specifically, when the captioner outputs a hidden state $h_t \in \mathbb{R}^d$ at step t, we introduce a trainable adapter θ that parameterizes the policy:

$$\tilde{h}_t = \theta(h_t), \tag{6}$$

where $\tilde{h}_t$ is then fed into the *frozen* LM head of the captioner to obtain token logits and the action distribution.

Following prior works that leverage imitation or supervised pretraining to stabilize and accelerate subsequent reinforcement learning [8, 24, 30, 38], our training process consists of two key stages: *(1)* imitation learning (IL) to warm-start the policy using expert trajectories generated by a captioner, and *(2)* RL fine-tuning with reward shaping.

Imitation Learning Warm-start. We first perform imitation learning to initialize the adapter. Concretely, we sample expert token sequences $\{(y_1, \ldots, y_T)\}$ from the frozen captioner and replay them to collect hidden states $\{h_t\}$. The adapter θ is then trained with a standard cross-entropy objective against the next-token targets, using the *frozen* LM head to produce logits:

$$\mathcal{L}_{\text{IL}} = -\frac{1}{N} \sum_{(h_t, y_{t+1})} \log P_{\text{LM}}\big(y_{t+1} \mid \tilde{h}_t\big), \tag{7}$$

where $P_{\text{LM}}(\cdot \mid \tilde{h}_t)$ denotes the token distribution output by the frozen LM head given the adapted hidden state $\tilde{h}_t$. Only the adapter parameters θ are updated in this stage; the backbone and LM head remain frozen. This warm start provides a strong semantic initialization that stabilizes and accelerates subsequent RL training.

PPO Optimization. We second train the adapter θ with Proximal Policy Optimization (PPO) [35], using the clipped surrogate objective:

$$\mathcal{L}_{\text{PPO}} = \mathbb{E}_t\Big[\min\big(\rho_t(\theta)A_t,\ \text{clip}(\rho_t(\theta), 1-\epsilon, 1+\epsilon)A_t\big)\Big], \tag{8}$$

where $\rho_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}$ is the importance weight between the new and old policies, A_t is the advantage estimate, and ϵ is the clipping parameter. $\pi_\theta(a_t|s_t)$ denotes the probability of taking action a_t under state s_t with parameters θ. A_t is computed using the estimated state value from the value head, which guides the policy update toward actions that yield higher-than-expected rewards. The clip function truncates the policy ratio $\rho_t(\theta)$ within the range $[1-\epsilon, 1+\epsilon]$ to prevent excessively large policy updates, thereby stabilizing training and avoiding performance collapse. Only the adapter and value head are updated during PPO optimization.

3.4. Fuzz Testing-Powered Prompt Optimization

The RL-optimized prompts successfully achieve high semantic alignment with the target image and precisely capture the core visual entities, yielding concise, content-faithful subjects. Nevertheless, their ability to reconstruct stylistic and compositional modifiers is limited, largely because image captioning backbones provide weak prior knowledge of modifiers training data. To address this limitation, we introduce a fuzz optimization to refine the RL-derived prompts. As shown in Figure 2 and Algorithm 1, we initialize the seed pool with RL-optimized prompts and iteratively improve them through mutation and evaluation. At each iteration, a set of mutators, either subject- (e.g., paraphrasing, enrichment) or modifier-level (e.g., description and style generation), propose new prompt variants. Each mutated prompt is then used to synthesize an image, whose similarity to the target image determines its score. High-scoring prompts are retained in the seed pool for further mutation, enabling exploration and exploitation of the prompt space in a self-improving loop. This process progressively enriches the prompt with expressive modifiers while preserving its semantic grounding. However, applying fuzz testing to prompt stealing faces two key challenges. **First, there is a lack of high-quality seeds.** Recent studies [5, 7, 10] show that seed choice profoundly shapes the efficacy and trajectory of fuzzing, and sparse or weak seeds lead to poor coverage and premature convergence. **Second, effective mutation strategies are missing.** Existing approaches, such as traditional mutation strategies like Havoc in AFL [47] or recent LLM-based text mutation methods [5, 46], are not tailored to structured text-to-image prompts with *subject* and *modifier*.

To address these challenges, we introduce two key designs that make fuzzing both structure-aware and sample-efficient. First, we guide exploration with a compact, quality-driven seed pool that retains strong candidates and schedules the next mutation via principled selection. Second, we mutate prompts with strategies that respect the compositional structure of text-to-image inputs, separately handling *subject* and *modifier* to enrich details without drifting from content.

Seed Update and Selection. PROMPTMINER maintains a *capacity-constrained elitist* seed pool with a fixed size of candidates. After each evaluation, the new prompt and its score are inserted into the pool; the pool is kept sorted by score, and the lowest-scoring prompt is evicted when the capacity is exceeded. To choose the next seed for mutation, we adopt the Monte Carlo Tree Search (MCTS) [1] algorithm to balance exploration and exploitation. This mechanism preserves high-quality prompts, continuously explores promising neighborhoods in prompt space.

Algorithm 1 Workflow of Fuzz Optimization

Input: Target image $\boldsymbol{x}$; RL-optimized prompt p_{RL}; text-to-image generator $\mathcal{G}$; similarity function Ψ; VLM-based mutator set $\mathcal{M}$; seed pool capacity K; query budget Q.
Result: Optimized subject–modifier prompt $p^\star$.

$\mathcal{S} \leftarrow \{(p_{\text{RL}}, \Psi(\mathcal{G}(p_{\text{RL}}), \boldsymbol{x}))\}$ ▷ *Initialization*
for $t = 1$ **to** Q **do**
  $(p_{\text{seed}}, s_{\text{seed}}) \leftarrow \text{SELECTFROMPOOL}(\mathcal{S})$ ▷ *MCTS*
  $\mathcal{M}_i \leftarrow \text{SAMPLEMUTATOR}(\mathcal{M})$ ▷ *Mutation*
  $p_{\text{new}} \leftarrow \mathcal{M}_i(p_{\text{seed}}, \boldsymbol{x})$
  $\hat{\boldsymbol{x}} \leftarrow \mathcal{G}(p_{\text{new}})$ ▷ *Execution*
  $s_{\text{new}} \leftarrow \Psi(\hat{\boldsymbol{x}}, \boldsymbol{x})$
  $\mathcal{S} \leftarrow \mathcal{S} \cup \{(p_{\text{new}}, s_{\text{new}})\}$ ▷ *Update*
  keep top-K elements in $\mathcal{S}$ by score s
end for
$(p^\star, s^\star) \leftarrow \arg\max_{(p,s)\in\mathcal{S}} s$
return $p^\star$

Hybrid-Strategy Prompt Mutation. To enhance expressiveness and visual fidelity, PROMPTMINER employs a VLM mutator (e.g., Qwen2-VL-2B-Instruct [40]) with five targeted operators: (1) ***Subject-Paraphrase*** rewrites the subject for more natural phrasing without changing meaning; (2) ***Subject-Enrich*** inserts brief image-grounded details (e.g., color, count, pose) while preserving sentence structure; (3) ***Modifier-Generate*** jointly produces a new description and style from the image and subject; (4) ***Modifier-Description*** refines the existing description with spatial relations, composition, and lighting cues; and (5) ***Modifier-Style*** adjusts medium, texture, lens, and quality tokens to strengthen aesthetic control. Together, these operators expand RL-derived subjects into well-formed prompts that remain semantically grounded yet stylistically rich. See Section G for detailed descriptions and the exact prompt templates of each mutator.

4. Evaluation

4.1. Experiment Setup

Target Models. We employ the most widely used state-of-the-art models, including Stable Diffusion v1.5 [33], SDXL-Turbo [34], Stable Diffusion 3.5 Medium [4] and FLUX.1 dev [12], as our target text-to-image generative models.

Datasets. We empirically evaluate our prompt stealing pipeline on four widely used datasets: MS COCO [17], Flickr [45], Lexica [36], and DiffusionDB [43]. MS COCO

Table 1. Image similarity and textual alignment comparison across datasets.

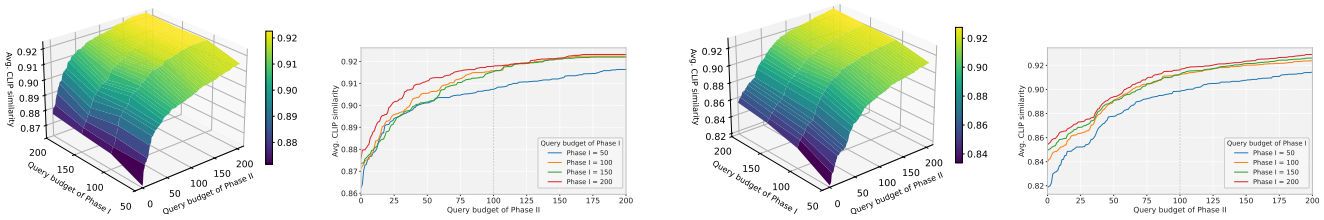
Dataset	Method	Stable Diffusion v1.5			SDXL-Turbo			FLUX.1 dev			Stable Diffusion 3.5 Medium		
		CLIP↑	LPIPS↓	SBERT↑	CLIP↑	LPIPS↓	SBERT↑	CLIP↑	LPIPS↓	SBERT↑	CLIP↑	LPIPS↓	SBERT↑
MS COCO	BLIP	0.858	0.458	0.660	0.855	0.444	0.643	0.885	0.467	**0.700**	0.881	0.409	0.718
	CLIP-IG	0.874	0.461	0.499	0.862	0.466	0.486	0.887	0.491	0.551	0.883	0.461	0.565
	VLM-as-expert	0.861	0.466	0.556	0.866	0.462	0.539	0.923	0.407	0.588	0.913	0.387	0.612
	PH2P	0.673	0.724	0.041	0.645	0.612	0.047	0.641	0.703	0.040	0.642	0.724	0.044
	PromptStealer	0.861	0.453	0.633	0.861	0.439	0.652	0.898	0.464	0.680	0.866	0.446	0.693
	VGD	0.816	0.700	0.424	0.802	0.565	0.410	0.791	0.630	0.417	0.785	0.691	0.416
	PROMPTMINER (Ours)	**0.933**	**0.342**	**0.664**	**0.934**	**0.340**	**0.673**	**0.958**	**0.345**	0.683	**0.953**	**0.303**	**0.751**
Flickr	BLIP	0.773	0.493	0.547	0.808	0.485	0.549	0.833	0.499	0.620	0.820	0.465	0.637
	CLIP-IG	0.833	0.466	0.472	0.835	0.478	0.468	0.860	0.508	0.511	0.857	0.487	0.547
	VLM-as-expert	0.817	0.469	0.514	0.827	0.473	0.501	0.895	0.447	0.571	0.874	0.431	0.578
	PH2P	0.638	0.715	0.039	0.616	0.629	0.046	0.630	0.686	0.038	0.613	0.721	0.030
	PromptStealer	0.783	0.504	0.501	0.816	0.475	0.581	0.843	0.498	0.595	0.835	0.484	0.622
	VGD	0.764	0.671	0.341	0.761	0.577	0.366	0.764	0.613	0.353	0.753	0.684	0.329
	PROMPTMINER (Ours)	**0.894**	**0.402**	**0.568**	**0.910**	**0.405**	**0.603**	**0.927**	**0.411**	**0.628**	**0.916**	**0.410**	**0.660**
Lexica	BLIP	0.699	0.590	0.319	0.749	0.507	0.397	0.760	0.554	0.335	0.746	0.560	0.406
	CLIP-IG	0.840	0.471	0.553	0.856	0.451	0.591	0.860	0.504	0.587	0.859	0.474	0.616
	VLM-as-expert	0.794	0.501	0.507	0.811	0.450	0.527	0.866	0.445	0.535	0.857	0.441	0.572
	PH2P	0.640	0.729	0.178	0.662	0.569	0.171	0.657	0.657	0.173	0.613	0.682	0.174
	PromptStealer	0.762	0.560	**0.563**	0.787	0.475	**0.594**	0.788	0.563	**0.600**	0.804	0.520	0.622
	VGD	0.753	0.701	0.466	0.766	0.530	0.446	0.760	0.612	0.481	0.762	0.643	0.485
	PROMPTMINER (Ours)	**0.875**	**0.450**	**0.563**	**0.911**	**0.420**	0.591	**0.921**	**0.435**	0.594	**0.921**	**0.407**	**0.629**

Figure 3. Visualization of generated images compared with target image.

and Flickr feature sentence-style captions that describe entities, actions, and scene layout without art-style tags, whereas Lexica and DiffusionDB contain keyword-driven prompts that follow a *"subject, modifiers"* pattern with comma-separated stylistic and technical tokens. For the first three datasets, we randomly select 50 prompts and use the target text-to-image generators to synthesize the corresponding images. For DiffusionDB, we treat it as an *in-the-wild* dataset: instead of generating new images with the target models, we directly use 50 original images from the dataset, where the underlying generative models are unknown.

Baselines. We compare PROMPTMINER with state-of-the-art prompt stealing and auxiliary baselines: BLIP [14], an image–captioning model producing sentence-style prompts; CLIP Interrogator (CLIP-IG) [25], which pairs CLIP with BLIP and selects modifiers from a large pool; VLM-as-expert using GPT-4o [9] for multimodal prompt generation; PromptStealer [37], which trains a subject generator and a modifier classifier on large labeled data; PH2P [19], which applies delayed discrete projection from late diffusion steps; and VGD [11], a gradient-free method that uses an LLM with CLIP-based guidance to produce coherent prompts.

Evaluation Metrics. We evaluate prompt stealing performance in terms of **image similarity** and **textual alignment**. For image similarity, we use CLIP [28] and LPIPS [48] similarity, where CLIP similarity measures high-level semantic



(a) Impact of query budget on Flick dataset.

(b) Impact of query budget on Lexica dataset.

Figure 4. Impact of query budget on *phase I* (RL-based Prompt Inversion) and *phase II* (Fuzz Testing–Powered Prompt Optimization).

Table 2. Prompt stealing performance on *in-the-wild* images.

Method	Image Similarity		Textual Alignment
	CLIP↑	LPIPS↓	SBERT↑
BLIP	0.753	0.659	0.365
CLIP-IG	0.803	0.628	0.541
VLM-as-expert	0.782	0.627	0.544
PH2P	0.648	0.706	0.154
PromptStealer	0.754	0.643	0.542
VGD	0.749	0.649	0.454
PROMPTMINER (Ours)	**0.863**	**0.622**	**0.545**

alignment in a joint vision–language space and LPIPS similarity captures low-level perceptual differences. For textual alignment, we use Sentence-BERT (SBERT) [32] to compute the cosine similarity between embeddings of the inverted and ground-truth prompts, indicating how well the recovered text preserves the original semantics.

More details about the experimental setup are provided in the supplementary material.

4.2. Experiment Result

Image Similarity. We evaluate the effectiveness of the inverted prompts generated by PROMPTMINER for image generation. As shown in Table 1, PROMPTMINER surpasses all the baselines across all datasets and text-to-image generative models in both CLIP and LPIPS similarity. Concretely, on MS COCO, PROMPTMINER attains CLIP similarity (0.958) and LPIPS similarity (0.345) with FLUX.1 dev, and CLIP Similarity (0.953) and LPIPS Similarity (0.303) with Stable Diffusion 3.5 Medium; on Flickr with FLUX.1 dev, PROMPTMINER records CLIP similarity (0.927) together with LPIPS similarity (0.411); on Lexica with Stable Diffusion 3.5 Medium, it achieves CLIP similarity (0.921) and LPIPS similarity (0.407). These gains demonstrate state-of-the-art image similarity, reflecting stronger semantic alignment and perceptual fidelity than prior methods. We visualize representative generations on FLUX.1 dev across MS COCO, Flickr, and Lexica in Figure 3. The images produced by PROMPTMINER are visibly closer to the targets than those from baselines, consistent with the quantitative gains. Additional examples are provided in the Section L.

Textual Alignment. We measure the semantic alignment between the inverted prompts and the ground-truth prompts using sentence-BERT Socre (SBERT) [32]. PROMPTMINER achieves the highest SBERT across nearly all datasets and models, showing superior text-level recovery. With Stable Diffusion 3.5 Medium, SBERT rises from 0.718 with BLIP on MS COCO to **0.751** with PROMPTMINER; on Flickr, it improves from 0.637 to **0.660**; on Lexica, PROMPTMINER reaches **0.629**, surpassing all baselines. These consistent improvements confirm that PROMPTMINER reconstructs subject–modifier semantics more faithfully, producing linguistically natural and semantically aligned prompts.

We attribute these improvements to the two-stage design of PROMPTMINER. The RL-based adapter ensures that the recovered subjects remain tightly aligned with image semantics, providing a stable content foundation. The subsequent fuzz testing-powered optimization stage systematically explores modifier space, enriching the prompt with stylistic and compositional cues that are crucial for achieving perceptual resemblance. This synergy allows PROMPTMINER to generate inverted prompts that not only reproduce the main subject accurately but also replicate fine-grained artistic attributes.

Prompt Stealing on In-The-Wild Images. We further evaluate the performance of PROMPTMINER on real-world images whose underlying generation models are unknown, demonstrating the generalization ability of our method in unconstrained scenarios. Specifically, we randomly sample 50 image–prompt pairs from the DiffusionDB [43] dataset, where each image is generated by an unknown diffusion model and accompanied by its original text prompt. As shown in Table 2, our method achieves the highest scores on all metrics. This demonstrates that even when the source model of the image is completely unknown, our method can effectively leverage a proxy text-to-image generative model to perform reliable prompt stealing. Such generalization highlights the robustness of our framework and its ability to capture transferable visual–textual correlations across heterogeneous generation models.

4.3. Ablation Study

Impact of Query Budget. Figure 4 illustrates the performance of PROMPTMINER on CLIP similarity under different query budgets. Specifically, *Phase I* (RL-based Prompt Inversion) uses query budgets of 50, 100, 150, and 200, while *Phase II* (Fuzz Testing–Powered Prompt Optimization) varies its query budget from 1 to 200. On Flickr (predominantly subject-only prompts), *Phase I* exhibits rapid con-

Table 3. Potential defenses against PROMPTMINER.

Defense	Image Similarity		Textual Alignment
	CLIP↑	LPIPS↓	SBERT↑
No defense	0.911	0.420	0.591
Random Noise Injection	0.903	0.420	0.572
Puzzle Effect	0.902	0.425	0.561
Textual Watermarking	0.887	0.445	0.583

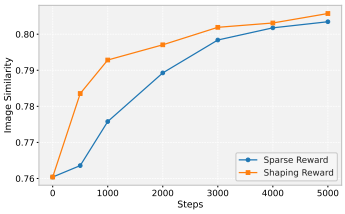


Figure 5. Impact of reward shaping.

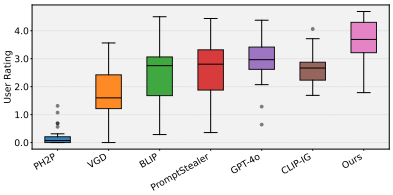


Figure 6. Ratings from the user study.

vergence: once the query budget reaches roughly one hundred, additional budget yields minimal improvement. *Phase II* attains comparable final CLIP scores across these settings—indicating limited marginal value from further *Phase I* investment for simple prompts. On Lexica (prompts combining subjects with rich modifiers), the enlarged search space slows *Phase I* convergence; allocating a larger budget produces stronger initial subjects and enables *Phase II* to reach higher final CLIP similarity. Considering both efficiency and quality, **there exists a clear trade-off between allocating queries to *Phase I* and *Phase II***. As shown in Figure 4, the optimal goal is to reach the **upper yellow region**—representing high CLIP similarity—with minimal query expenditure. Increasing the *Phase I* budget enhances initialization quality, particularly for complex prompts, whereas investing more in *Phase II* refines stylistic modifiers but quickly exhibits diminishing returns. Empirically, balanced allocations with **100** queries per phase achieve strong performance efficiently, approaching the optimal region without excessive cost.

Impact of Reward Shaping. We evaluate the impact of reward shaping by comparing training dynamics under sparse versus shaping rewards. The sparse reward provides feedback only at the end of an episode, while the shaped reward adds intermediate dense signals to accelerate learning. As shown in Figure 5, The shaped reward supplies dense intermediate feedback via a potential function, enabling faster progress in early stages and reaching a high image similarity with substantially fewer steps. In contrast, the sparse-reward variant lacks intermediate guidance and therefore converges more slowly. Overall, potential-based reward shaping accelerates convergence and improves training efficiency while maintaining stability.

For additional ablation results, please see Section I.

4.4. User Study

We conducted a user study to further evaluate the effectiveness of our method from the perspective of human perception. Participants are presented with a target image and several anonymous candidate images generated by PROMPTMINER and baselines, and are asked to select the one that appears most visually similar to the target. The similarity preference is rated on a five-point scale from 0 to 5. We randomly select a total of 30 test cases from the experimental results covering all the three datasets and four generative models. According to the results in Figure 6, PROMPTMINER consistently receives the highest preference scores, indicating that its recovered prompts generate images that better align with human visual perception.

5. Potential Defenses

Given the increasing risk of prompt stealing attacks, we further explore several existing defenses that can be applied during image dissemination to mitigate potential prompt extraction or replication. Specifically, as same as [50], we investigate three representative defenses: (1) **Random noise Injection**, where Gaussian noise is added to subtly perturb pixel-level information; (2) **Puzzle effect**, which divides the image into a 4×4 grid and applies random local translations which is applied in practice [2]; and (3) **Textual watermarking**, which overlays a visible pattern such as "@watermark" across the image to indicate ownership or provenance and also used in reality [21]. These methods differ in their balance between imperceptibility and robustness: while noise and puzzle effect are visually subtle and suitable for general protection, watermarking provides explicit attribution at the cost of stronger visual alteration. We evaluate the effectiveness of each defense in hindering prompt stealing in Table 3. It is obvious that these post-processing defenses have only a minor impact on preventing our prompt stealing attack, indicating that PROMPTMINER is inherently resilient to low-level visual perturbations due to its semantics-driven optimization process.

6. Conclusion

In this paper, we propose PROMPTMINER, a novel and effective black-box framework for prompt stealing against text-to-image generative models. Unlike prior approaches that rely on gradient access or large-scale supervised training, PROMPTMINER efficiently reconstructs prompts through a two-phase process, RL–based subject inversion and fuzz-testing–powered modifier optimization, achieving both semantic precision and stylistic fidelity in the recovered prompts. Extensive experiments across diverse datasets and diffusion backbones show that PROMPTMINER consistently outperforms existing baselines in image and textual similarity, generalizes to in-the-wild settings, and remains robust under common defensive perturbations. This

work highlights the growing need for prompt protection in generative systems and provides a foundation for future research on secure and interpretable prompt recovery.

References

[1] Rémi Coulom. Efficient selectivity and backup operators in monte-carlo tree search. In *Computers and Games*, pages 72–83, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg. 5

[2] Dianamoran@PromptBase. Vivid bright bold graphic illustrations, 2025. 8, 3

[3] 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. 2

[4] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis, 2024. 5, 1

[5] Xueluan Gong, Mingzhe Li, Yilin Zhang, Fengyuan Ran, Chen Chen, Yanjiao Chen, Qian Wang, and Kwok-Yan Lam. {PAPILLON}: Efficient and stealthy fuzz {Testing-Powered} jailbreaks for {LLMs}. In *34th USENIX Security Symposium (USENIX Security 25)*, pages 2401–2420, 2025. 5

[6] Yaru Hao, Zewen Chi, Li Dong, and Furu Wei. Optimizing prompts for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:66923–66939, 2023. 1

[7] Adrian Herrera, Hendra Gunadi, Shane Magrath, Michael Norrish, Mathias Payer, and Antony L. Hosking. Seed selection for successful fuzzing. In *Proceedings of the 30th ACM SIGSOFT International Symposium on Software Testing and Analysis*, page 230–243, New York, NY, USA, 2021. Association for Computing Machinery. 5

[8] Todd Hester, Matej Vecerik, Olivier Pietquin, et al. Deep q-learning from demonstrations. In *AAAI*, 2018. 4

[9] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 2, 6, 1

[10] Aftab Hussain and Mohammad Amin Alipour. Removing uninteresting bytes in software fuzzing. In *2022 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, pages 301–305, 2022. 5

[11] Donghoon Kim, Minji Bae, Kyuhong Shim, and Byonghyo Shim. Visually guided decoding: Gradient-free hard prompt inversion with language models, 2025. 1, 2, 3, 6

[12] Black Forest Labs. Flux, 2024. 1, 2, 5

[13] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation, 2022. 1

[14] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022. 2, 6, 1, 8

[15] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 2

[16] Mingzhe Li, Gehao Zhang, Zhenting Wang, Shiqing Ma, Siqi Pan, Richard Cartwright, and Juan Zhai. Editor: Effective and interpretable prompt inversion for text-to-image diffusion models. *arXiv preprint arXiv:2506.03067*, 2025. 1, 3

[17] 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*, pages 740–755, Cham, 2014. Springer International Publishing. 2, 5, 1

[18] Vivian Liu and Lydia B. Chilton. Design guidelines for prompt engineering text-to-image generative models, 2023. 1, 3

[19] Shweta Mahajan, Tanzila Rahman, and Leonid Sigal Kwang Moo Yi. Prompting hard or hardly prompting: Prompt inversion for text-to-image diffusion models. In *CVPR 2024*, 2024. 1, 2, 3, 6

[20] Ron Mokady, Amir Hertz, and Amit H Bermano. Clipcap: Clip prefix for image captioning. *arXiv preprint arXiv:2111.09734*, 2021. 3

[21] Mylab@PromptBase. Summer animal relaxes, 2025. 8

[22] Ali Naseh, Katherine Thai, Mohit Iyyer, and Amir Houmansadr. Iteratively prompting multimodal llms to reproduce natural and ai-generated images. *arXiv preprint arXiv:2404.13784*, 2024. 2

[23] Andrew Y Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *Icml*, pages 278–287. Citeseer, 1999. 4, 11

[24] Long Ouyang, Jeff Wu, Xu Jiang, et al. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*, 2022. 4

[25] Pharmapsychotic. Clip interrogator, 2025. 2, 3, 6, 1

[26] PromptBase. Promptbase, 2025. 1

[27] PromptHero. Prompthero, 2025. 1

[28] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 6

[29] 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. 2

[30] Aravind Rajeswaran, Vikash Kumar, Abhishek Gupta, et al. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. In *RSS*, 2018. 4

[31] 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. 1

[32] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks, 2019. 7, 2

[33] 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 (CVPR)*, pages 10684–10695, 2022. 1, 2, 5

[34] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation, 2023. 2, 5, 1

[35] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. 4

[36] Xinyue Shen, Yiting Qu, Michael Backes, and Yang Zhang. Prompt Stealing Attacks Against Text-to-Image Generation Models. In *USENIX Security Symposium (USENIX Security)*. USENIX, 2024. 2, 5

[37] Xinyue Shen, Yiting Qu, Michael Backes, and Yang Zhang. Prompt stealing attacks against text-to-image generation models, 2024. 1, 2, 3, 6

[38] David Silver, Aja Huang, Chris J. Maddison, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 2016. 4

[39] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. A Bradford Book, Cambridge, MA, USA, 2018. 2, 3

[40] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 5, 3

[41] Zhenting Wang, Chen Chen, Lingjuan Lyu, Dimitris N Metaxas, and Shiqing Ma. Diagnosis: Detecting unauthorized data usages in text-to-image diffusion models. In *The Twelfth International Conference on Learning Representations*, 2023. 2

[42] Zhenting Wang, Chen Chen, Yi Zeng, Lingjuan Lyu, and Shiqing Ma. Where did i come from? origin attribution of ai-generated images. *Advances in neural information processing systems*, 36, 2024. 2

[43] 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:2210.14896 [cs]*, 2022. 5, 7, 2

[44] 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. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2024. Curran Associates Inc. 2, 3, 1

[45] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. 2, 5

[46] Jiahao Yu, Xingwei Lin, Zheng Yu, and Xinyu Xing. Gptfuzzer: Red teaming large language models with auto-generated jailbreak prompts, 2024. 5

[47] Michał Zalewski. American fuzzy lop, 2014. Fuzz testing framework. 5

[48] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 6, 2

[49] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert, 2020. 2

[50] Shiqian Zhao, Chong Wang, Yiming Li, Yihao Huang, Wenjie Qu, Siew-Kei Lam, Yi Xie, Kangjie Chen, Jie Zhang, and Tianwei Zhang. Towards effective prompt stealing attack against text-to-image diffusion models, 2025. 3, 8

Appendix

A. More Details of the Used Models

In this section, we provide more details about the models used in the experiments. We conduct experiments on four representative text-to-image generative models with different architectural designs, including Stable Diffusion v1.5[33], SDXL Turbo [34], Stable Diffusion 3.5 Medium [4], and FLUX.1 dev [12].

- **Stable Diffusion v1.5:** This model is initialized with the weights of the Stable-Diffusion-v1.2 checkpoint and subsequently fine-tuned on 595k steps at resolution 512x512 on "laion-aesthetics v2.5+" and 10% dropping of the text-conditioning to improve classifier-free guidance sampling. The text encoder of this model is CLIP-ViT/L.
- **SDXL Turbo:** This model is a distilled version of SDXL 1.0, trained for real-time synthesis. SDXL-Turbo is based on a novel training method called Adversarial Diffusion Distillation (ADD), which allows sampling large-scale foundational image diffusion models in 1 to 4 steps at high image quality. This approach uses score distillation to leverage large-scale off-the-shelf image diffusion models as a teacher signal and combines this with an adversarial loss to ensure high image fidelity even in the low-step regime of one or two sampling steps. The text encoders of this model are OpenCLIP-ViT/G and CLIP-ViT/L.
- **Stable Diffusion 3.5 Medium:** This model is a Multimodal Diffusion Transformer with improvements (MMDiT-X) text-to-image model that features improved performance in image quality, typography, complex prompt understanding, and resource-efficiency. The text encoders of this model are OpenCLIP-ViT/G, CLIP-ViT/L and T5-xxl.
- **FLUX.1 dev:** his model is built upon a 12-billion-parameter *rectified flow Transformer* architecture, a diffusion-based framework reformulated as a continuous flow matching process for improved stability and efficiency. The model employs a dual text-encoder design (T5-xxl and CLIP-ViT/L) for robust semantic conditioning and integrates a high-capacity autoencoder for latent-space compression.

B. More Details of Baselines

Table 4. Comparison of existing prompt stealing and prompt inversion Methods.

Method	Modifier	Optimization	Black-box
PEZ [44]	✗	✓	✗
PH2P [19]	✗	✓	✗
BLIP [14]	✗	✗	✓
CLIP-IG [25]	✓	✗	✓
VGD [11]	✗	✗	✓
PromptStealer [37]	✓	✗	✓
PROMPTMINER (Ours)	✓	✓	✓

We compare our approach with state-of-the-art prompt inversion and prompt stealing methods as summarized in Table 4. In addition, we include auxiliary baselines that derive prompts from image captioning and from vision–language models (VLMs).

- **BLIP [13]:** This is an image captioning model that learns from image-caption pairs [17] to connect vision and language.
- **CLIP Interrogator [25]:** This is a tool that uses CLIP and BLIP together to analyze an image and suggest text prompts that best describe it. In addition, it chooses extra descriptive modifiers from a predefined large-scale modifier pool.
- **VLM-as-an-expert:** The vision-language model provides strong multimodal reasoning ability, allowing it to interpret complex visual content, align it with linguistic context, and generate accurate, coherent prompts. Here we use GPT-4o [9] for this purpose.
- **PromptStealer [37]:** This is a prompt stealing attack that attempts to recover the prompt. It consists of a subject generator to infer the main subject and a modifier detector to identify descriptive modifiers.
- **PH2P [19]:** This method focuses on the later, more noisy timesteps and uses delayed projection into the discrete token vocabulary to recover human-readable prompts that reflect the visual content.

- **VGD [11]:** This is a gradient-free hard prompt inversion method that uses a large language model and CLIP guidance to generate human-readable prompts aligned with visual content, without extra training.

C. More Details of Datasets

We empirically evaluate our prompt stealing pipeline on four widely used datasets: MS COCO [17], Flickr [45], Lexica [36], and DiffusionDB [43]. For the first three datasets, we randomly select 50 prompts and use the target text-to-image generative models to synthesize the corresponding images. For DiffusionDB, we treat it as an *in-the-wild* dataset: instead of generating new images with the target models, we directly use 50 original images from the dataset, where the underlying generative models are unknown.

- **MS COCO and Flickr:** MS COCO is a large benchmark of everyday scenes with natural photographs. Each image has multiple human-written captions. The captions are full sentences that name visible objects, actions, and simple scene context in plain language. Flickr is a captioning dataset of real-world photos collected from Flickr with five human-written sentences per image. These two datasets use sentence-style prompts that describe entities, actions, and scene layout without art-style tags.
- **Lexica and DiffusionDB:** Lexica is a collection of user-entered prompts from an online prompt search service for text-to-image models. Prompts are keyword-focused and often include long chains of style, medium, camera, lighting, and quality terms; some entries also include negative phrases. DiffusionDB is the first large-scale text-to-image prompt dataset. It contains 14 million images generated by Stable Diffusion using prompts and hyperparameters specified by real users. Prompts are typically keyword lists with stylistic and technical modifiers rather than full sentences about photographic scenes. Prompts of these two datasets follow a *"subject, modifiers"* pattern, where modifiers are comma-separated terms for style, medium, rendering setup, lighting, and quality.

D. More Details of Metrics

We employ both image similarity and textual alignment metrics to quantitatively evaluate the quality of generated images and the stolen prompts. Specifically, we use CLIP Similarity and LPIPS Similarity for image comparison, and BERTScore and SBERT for textual alignment assessment.

- **CLIP Similarity:** This metric measures the semantic similarity between the generated image $\hat{x}$ and the target image x. It leverages the image encoder f_{img} from CLIP [29] to compute cosine similarity in the joint vision-language embedding space:

$$CLIP(\hat{x}, x) = \frac{f_{\text{img}}(\hat{x}) \cdot f_{\text{img}}(x)}{\|f_{\text{img}}(\hat{x})\| \|f_{\text{img}}(x)\|}, \tag{9}$$

  where a higher value indicates stronger semantic alignment between $\hat{x}$ and x.

- **LPIPS Similarity:** The Learned Perceptual Image Patch Similarity (LPIPS) [48] metric evaluates perceptual similarity between two images by comparing deep features extracted from a pretrained convolutional neural network (e.g., AlexNet or VGG). LPIPS computes the L2 distance between normalized features across multiple layers, reflecting human perceptual judgments of image quality. A lower LPIPS score indicates higher perceptual similarity.

- **BERTScore:** To measure token-level textual alignment, we use BERTScore [49] with a token encoder f_{bert}. For candidate tokens $\{\hat{p}_i\}_{i=1}^{|\hat{p}|}$ and reference tokens $\{p_j\}_{j=1}^{|p|}$, define cosine similarity $c_{ij} = \frac{f_{\text{bert}}(\hat{p}_i) \cdot f_{\text{bert}}(p_j)}{\|f_{\text{bert}}(\hat{p}_i)\| \, \|f_{\text{bert}}(p_j)\|}$. Precision and recall are

$$\mathrm{P} = \frac{1}{|\hat{p}|} \sum_i \max_j c_{ij}, \qquad \mathrm{R} = \frac{1}{|p|} \sum_j \max_i c_{ij}, \tag{10}$$

  and the final BERTScore is the F1:

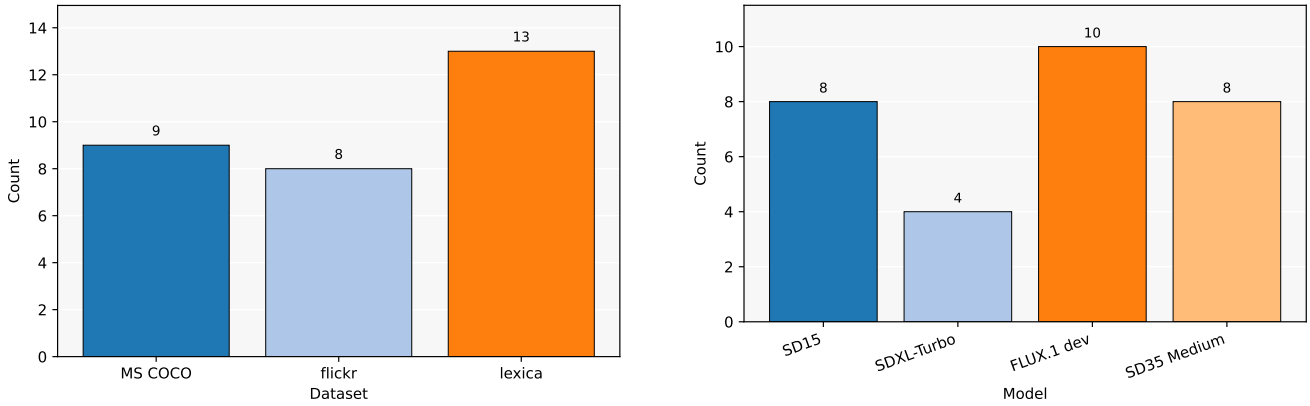
$$\mathrm{BERTScore}(\hat{p}, p) = \frac{2\mathrm{PR}}{\mathrm{P} + \mathrm{R}}. \tag{11}$$

- **SBERT:** Sentence-level textual alignment is computed with Sentence-BERT [32] using a sentence encoder f_{sbert}. Given a generated prompt $\hat{p}$ and a target prompt p, we take the cosine similarity of their sentence embeddings:

$$\mathrm{SBERT}(\hat{p}, p) = \frac{f_{\text{sbert}}(\hat{p}) \cdot f_{\text{sbert}}(p)}{\|f_{\text{sbert}}(\hat{p})\| \, \|f_{\text{sbert}}(p)\|}, \tag{12}$$

  where a higher score indicates stronger semantic equivalence.

E. More details of User Study



(a) Samples per dataset

(b) Samples per model

Figure 7. Distributions of the sampled test cases in the user study. **SD15** is Stable Diffusion v1.5, and **SD35 Medium** means Stable Diffusion 3.5 Medium.

We conduct a user study to assess the perceptual quality and fidelity of prompts recovered by different methods. We randomly select 30 image sets from three datasets and four generative models. Each set includes one target image and seven generated images produced by different prompt stealing or prompt inversion methods. During evaluation, participants are blinded to both the methods and the corresponding prompts, and the order of images within each set is randomly shuffled.

Participants rate the similarity between each generated image and its target on a 6-point Likert scale from 0 to 5, where 0 indicates *not similar at all* and 5 indicates *almost identical*. We compute the final score for each method as the average rating across all related images. The distribution of sampled test cases is illustrated in Figure 7.

F. More details of Defenses

As shown in Figure 8, we investigate three representative defenses: (1) **Random noise Injection**, where Gaussian noise with a mean of 0 and a standard deviation of approximately 25 is added to subtly perturb pixel-level information; (2) **Puzzle effect**, which divides the image into a 4×4 grid and applies random local translations with a variability parameter of 3, which is applied in practice [2]; and (3) **Textual watermarking**, which overlays a visible pattern such as "@watermark" across the image with a font size of 20 and row/column spacing of 20/30 pixels.

G. More details of VLM-Based Mutators

To enable structured, targeted prompt refinement, we design a suite of *VLM-based mutators* that operate on both subjects and modifiers. Each mutator leverages the vision-language reasoning ability of a powerful VLM (e.g., Qwen2-VL-2B-Instruct [40]) to generate linguistically natural and visually grounded edits. Specifically, these mutators either paraphrase or enrich the subject for clarity and detail, or modify the descriptive and stylistic aspects of the prompt to enhance expressiveness and visual fidelity. Below, we outline the five mutator types and their respective functions.

- *Subject-Paraphrase:* Designed to rewrite the subject without altering its semantic meaning, this operator restructures the sentence to achieve higher linguistic diversity and naturalness while maintaining content fidelity to the original description.

- *Subject-Enrich:* This operator augments the subject by inserting concise, image-grounded details such as color, quantity, or pose. The enrichment strengthens the visual grounding of the prompt while preserving its syntactic structure and readability.

- *Modifier-Generate:* Designed to synthesize **a total new *description* and *style*** jointly from the image and the subject, this operator produces a compact, comma-separated tag string for scene facts together with a short style tag, ensuring strong visual grounding and aesthetic control.

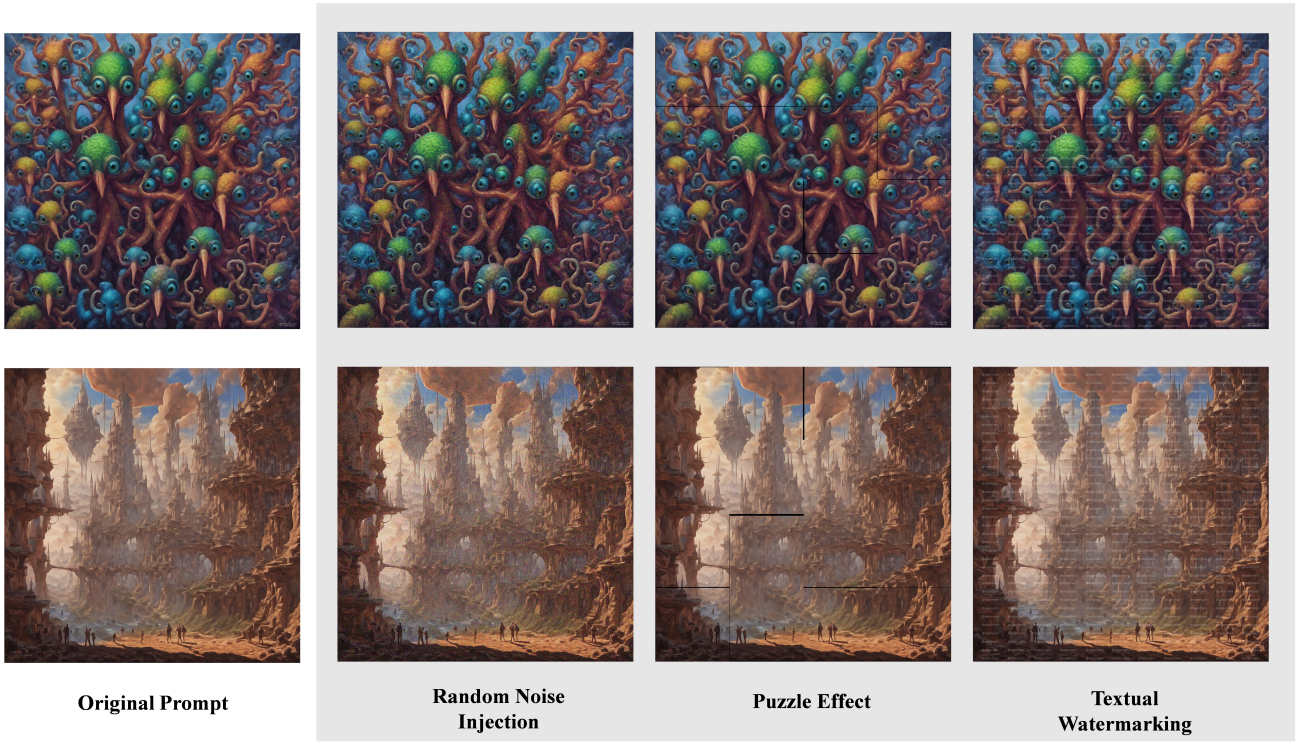


Figure 8. Visualization of images after applying three representative defenses.

- *Modifier-Description:* Designed to enhance descriptive richness, this operator refines or extends the **existing prompt's description** by incorporating spatial relations, compositional layouts, and lighting attributes observed in the image. It effectively bridges literal captions and visually rich scene descriptions.
- *Modifier-Style:* Starting from the **current prompt's style**, this operator introduces or adjusts stylistic tokens such as medium, texture, camera lens, or rendering quality. It allows the prompt to better control the generative behavior of the text-to-image model and produce outputs with stronger artistic fidelity.

Besides these five primary operators, we also include an auxiliary mutator, *Subject-Fix-Grammar*, to remedy artifacts introduced by RL optimization (e.g., token repetition, minor grammatical errors, punctuation/spacing issues). This lightweight cleanup pass preserves the original semantics and word order while improving readability and coherence. We show the exact prompts for each mutation below.

G.1. Shared System Prompt

System Prompt (shared across mutators)

you are a prompt mutator for text-to-image diffusion models.
given a base prompt and an input image, you must return EXACTLY ONE SINGLE-LINE JSON object.
- lowercase english only; no markdown; no code fences; no trailing commas; no extra text.
- never insert line breaks inside values.
- if the base prompt conflicts with the image, trust the image.
- be concrete and visual; do not invent invisible objects.
- field-specific rules:
description: 15-35 words, write as a compact comma-separated tag string (not full sentences). include: subject and key attributes or pose, setting/location, composition/shot/angle, lighting, overall color tendency, one brief quality token (e.g., highly detailed or sharp focus), optional material/texture cue, artist, plus up to 2 simple negatives (e.g., no watermark, no text). use only visible facts.
style: <= 12 words, a short comma-separated tag string of medium/movement/lens/quality only (e.g., digital painting,

photorealistic, vector art, isometric, 35mm lens, 85mm lens, film grain, clean render). do not include scene facts or lighting.
base_prompt: <= 15 words, preserve the original meaning, clearer phrasing, avoid style or lighting tokens.
examples:
input(base): 'two samurai duel in a bamboo forest'
output(desc+style): "description":"bamboo grove, two samurai facing between tall stalks, medium shot, eye-level, dappled sunlight, green tones, foreground leaves, no text","style":"ink illustration, ukiyo-e inspired, paper texture"
input(base): 'astronaut and robot on mars at dawn'
output(modify-desc): "description":"red dunes, astronaut left of small robot, wide shot, low angle, soft dawn light, cool shadows, distant mountains, no watermark"
input(base): 'portrait of an old musician in neon city'
output(modify-style): "style":"photorealistic, 85mm lens, cinematic still"
input(base): 'a child reading a book under a tree'
output(paraphrase-base): "base_prompt":"a child reading beneath a tree"

G.2. Subject Mutators

Subject-Enrich (user prompt)

task: ENRICH the base prompt by inserting concise, image-grounded modifiers WITHOUT changing its syntactic skeleton or word order.
base prompt: 'base_prompt'
you will receive the IMAGE together with this message. use ONLY details that are VISIBLE in the image.
output schema (single line json): "base_prompt":"..."
constraints:
- preserve the original tokens as an ordered subsequence: do not delete, replace, or reorder existing words; no synonym substitution.
- keep the subject–verb–object–prepositional structure intact.
- INSERT 2–6 words total, placed immediately AFTER the nouns/verbs they modify (adjectives/appositives for nouns; short adverbs/adjuncts for verbs).
- allowed insertions: count/quantity (one/two), object attributes (color, size, material), pose/state, simple spatial cues relative to BACKGROUND (e.g., near the fence, in front of the gate), and other concrete scene facts visible in the image.
- forbid insertions about style/lighting/lens/artist or abstract aesthetics (these belong to style).
- if a detail is uncertain or not visible, DO NOT add it; trust the image over the text.
- lowercase only; no quotes; no extra commentary.
examples:
base: 'a child reading a book under a tree'
enriched: 'base_prompt':'a small child quietly reading a worn book under a shady tree'
base: 'a dog runs across a field'
enriched: 'base_prompt':'a brown dog runs swiftly across a grassy field'
return ONLY the json line.

Subject-Fix-Grammar (user prompt)

task: CLEANUP the base prompt by fixing grammar/spelling, removing duplicated words/phrases, and correcting spacing/punctuation ONLY.
base prompt: 'base_prompt'
output schema (single line json): "base_prompt":"..."
constraints:
- do NOT add any new content or modifiers; do NOT introduce synonyms; do NOT reorder clauses.

- preserve the original subject–verb–object–prepositional order and overall sentence structure.
- only remove repeated tokens/phrases, fix typos, collapse multiple spaces, and standardize minimal punctuation.
- keep length approximately unchanged (within ±2 words of the original).
- lowercase only; no quotes; no extra commentary.
return ONLY the json line.

Subject-Paraphrase (user prompt)

task: PARAPHRASE the base prompt WITHOUT changing its meaning, and enforce the structure:
WHO/WHAT + is doing + WHERE.
base prompt: 'base_prompt'
output schema (single line json): "base_prompt":"..."
constraints:
- <= 15 words total.
- structure must be strictly: <who/what> + 'is/are' + <present participle> + <where-phrase>.
- examples: 'a child is reading under a tree'; 'two samurai are dueling in a bamboo forest'; 'a red car is driving along a rainy street'.
- no style/lighting/lens/artist tokens.
- preserve entities and relations; trust the image if there is a conflict.
- lowercase only; no quotes; no extra commentary.

return ONLY the json line.

G.3. Modifier-Generate

GEN_DESC_STYLE (user prompt)

task: generate a NEW description and a NEW style from the base prompt and the image. base prompt: 'base_prompt'
output schema (single line json): "description":"...","style":"..."
constraints:
- description: 15–35 words, compact comma-separated tag string (not full sentences).
- include, in order when possible: subject+attributes/pose, setting, composition/shot/angle, lighting, overall color tendency, one quality token, optional material/texture, artist, up to 2 simple negatives.
- include one explicit spatial or depth cue (e.g., foreground, in front of distant hills).
- style: <= 12 words; medium/movement/lens/quality only; no scene facts; no lighting.
return ONLY the json line.

Modifier-Style (user prompt)

task: CHANGE STYLE ONLY while preserving entities and relations in the current description.
current style: style-from-parent
current description: description-from-parent
base prompt: 'base_prompt'
output schema (single line json): "style":"..."
constraints:
- concise comma-separated tag string (<=12 words) of medium/movement/lens/quality.
- do NOT include scene facts or lighting.
- keep the aesthetic consistent; improve clarity or fidelity.
return ONLY the json line.

Modifier-Description (user prompt)

task: CHANGE DESCRIPTION ONLY while preserving the subject meaning and current style.
current description: description-from-parent
current style: style-from-parent
base prompt: 'base_prompt'
output schema (single line json): "description":"..."
constraints:
- compact comma-separated tag string (15–35 words).
- include: subject+attributes/pose, setting, composition/shot/angle, lighting, overall color tendency, one quality token, optional material/texture, artist, up to 2 negatives.
- include one explicit spatial or depth cue.
- concrete visible details only; avoid abstractions; do not change the style semantics.
return ONLY the json line.

H. Token-Level Textual Alignment

Table 5. Token-level textual alignment comparison across datasets.

Dataset	Method	Stable Diffusion v1.5			SDXL-Turbo			FLUX.1 dev			Stable Diffusion 3.5 Medium		
		P↑	R↑	F1↑	P↑	R↑	F1↑	P↑	R↑	F1↑	P↑	R↑	F1↑
MS COCO	BLIP	0.905	0.911	0.908	0.905	0.912	0.909	0.907	0.917	0.912	0.912	0.921	0.916
	CLIP-IG	0.803	0.894	0.846	0.793	0.896	0.841	0.797	0.898	0.844	0.797	0.900	0.845
	VLM-as-expert	0.826	0.898	0.860	0.826	0.898	0.861	0.826	0.904	0.863	0.831	0.908	0.868
	PH2P	0.761	0.827	0.793	0.768	0.827	0.797	0.765	0.827	0.795	0.757	0.827	0.790
	PromptStealer	0.905	0.919	0.912	0.915	0.922	0.918	0.911	0.921	0.916	0.917	0.926	0.922
	VGD	0.822	0.874	0.847	0.829	0.880	0.853	0.827	0.874	0.850	0.822	0.874	0.847
	PROMPTMINER (Ours)	0.904	0.901	0.902	0.903	0.906	0.904	0.903	0.919	0.911	0.904	0.915	0.909
Flickr	BLIP	0.901	0.887	0.894	0.907	0.888	0.897	0.901	0.891	0.896	0.903	0.889	0.896
	CLIP-IG	0.809	0.876	0.841	0.800	0.878	0.837	0.803	0.880	0.839	0.802	0.880	0.839
	VLM-as-expert	0.835	0.890	0.861	0.832	0.887	0.859	0.838	0.900	0.867	0.840	0.901	0.869
	PH2P	0.767	0.814	0.789	0.762	0.813	0.786	0.758	0.813	0.784	0.763	0.813	0.787
	PromptStealer	0.895	0.889	0.892	0.911	0.898	0.904	0.910	0.895	0.902	0.911	0.899	0.905
	VGD	0.827	0.858	0.842	0.826	0.860	0.842	0.824	0.856	0.839	0.824	0.856	0.839
	PROMPTMINER (Ours)	0.906	0.904	0.905	0.917	0.907	0.912	0.914	0.901	0.907	0.913	0.905	0.909
Lexica	BLIP	0.855	0.783	0.817	0.855	0.782	0.816	0.853	0.782	0.815	0.856	0.784	0.818
	CLIP-IG	0.828	0.823	0.825	0.827	0.822	0.825	0.827	0.824	0.826	0.822	0.825	0.823
	VLM-as-expert	0.822	0.802	0.812	0.825	0.803	0.814	0.823	0.803	0.813	0.826	0.806	0.816
	PH2P	0.761	0.774	0.767	0.769	0.775	0.772	0.761	0.774	0.768	0.760	0.775	0.767
	PromptStealer	0.865	0.820	0.842	0.869	0.817	0.842	0.871	0.818	0.843	0.872	0.819	0.844
	VGD	0.822	0.791	0.806	0.825	0.790	0.807	0.790	0.816	0.803	0.814	0.789	0.801
	PROMPTMINER (Ours)	0.844	0.810	0.826	0.846	0.809	0.827	0.850	0.820	0.835	0.856	0.823	0.839

Table 6. Token-level textual alignment comparison on *in-the-wild* datasets with SDXL-Turbo.

Method	**Textual Alignment**		
	P↑	R↑	F1↑
BLIP	0.857	0.815	0.835
CLIP-IG	0.819	0.841	0.830
VLM-as-expert	0.820	0.823	0.821
PH2P	0.770	0.791	0.780
PromptStealer	0.860	0.841	0.850
VGD	0.814	0.806	0.810
PROMPTMINER (Ours)	0.838	0.825	0.821

We evaluate token-level textual alignment on MS COCO, Flickr, and Lexica using four T2I backbones (Stable Diffusion v1.5, SDXL-Turbo, FLUX.1 dev, and Stable Diffusion 3.5 Medium). As shown in Table 5, PROMPTMINER delivers the highest or near-highest scores on Precision(P), Recall(R), and F1 Score(F1) across datasets and models, indicating that our

recovered prompts are both precise and comprehensive with respect to image content. We attribute these gains to the two-stage design: (i) *RL-Based Prompt Inversion* improves subject fidelity and structural consistency; (ii) *Fuzz Testing–Powered Prompt Optimization* introduces stylistic and compositional modifiers without sacrificing semantic grounding. Notably on Lexica, where prompts contain diverse, artist-crafted modifiers, PROMPTMINER maintains strong F1, demonstrating robustness to varied prompt distributions.

In-the-Wild Setting. We further assess generalization on images whose source generators are unknown. We use SDXL-Turbo as the text-to-image generative model. As reported in Table 6, PROMPTMINER attains competitive textual alignment, demonstrating that our black-box, semantics-driven optimization transfers effectively to unconstrained scenarios.

I. Ablation Study

Table 7. Impact of different VLMs used as mutators in the Fuzz Testing–Powered Optimization stage.

Mutator Model	Image Similarity		Textual Alignment
	CLIP↑	LPIPS↓	SBERT↑
Qwen2-VL-2B-Instruct	0.911	0.420	0.591
Qwen2-VL-7B-Instruct	0.910	0.447	0.597
GPT-4o	0.917	0.438	0.568

Impact of Different Mutators. We investigate the impact of using different vision–language models (VLMs) as mutators in the Fuzz Testing–Powered Optimization stage. Specifically, we select three representative VLMs with varying model capacities and architectures: **Qwen2-VL-2B-Instruct**, **Qwen2-VL-7B-Instruct**, and **GPT-4o**. The Qwen2-VL family represents open-source VLMs with strong multimodal grounding and scalable size variants, while GPT-4o serves as a powerful proprietary model that excels in multimodal reasoning and image–text alignment. As shown in Table 7, all three models enable effective prompt refinement, demonstrating the general applicability of our fuzz-testing optimization framework. Smaller open-source models such as Qwen2-VL-2B-Instruct already achieve competitive performance, indicating that our method does not rely on large-parameter models or extensive computational resources to generate high-quality modifiers and achieve strong inversion performance.

Table 8. Impact of *Phase I* (RL-based Prompt Inversion) and *Phase II* (Fuzz testing-Powered Optimization).

Dataset	Method	Image Similarity		Textual Alignment
		CLIP↑	LPIPS↓	SBERT↑
Flickr	CLIP-IG	0.835	0.478	0.468
	Phase I only	0.859	0.440	0.559
	Phase II only	0.891	0.433	0.541
	PROMPTMINER	0.910	0.405	0.603
Lexica	CLIP-IG	0.856	0.451	0.591
	Phase I only	0.832	0.475	0.440
	Phase II only	0.889	0.434	0.559
	PROMPTMINER	0.911	0.420	0.591

Impact of the Two Phase Design. We assess each stage separately and together using the ablation in Table 8. First, we validate *Phase I* by replacing its outputs with BLIP [14] captions that are then used to initialize *Phase II*. Second, we disable *Phase II* and directly evaluate prompts from *Phase I*. On Flickr dataset, the prompts mainly describe the subject. *Phase I* already surpasses the best baselines in both image similarity and semantic alignment, and adding *Phase II* yields additional improvements that produce the best overall results. On Lexica dataset, the prompts include the subject and diverse modifiers. *Phase I* alone is not sufficient, while *Phase II* brings clear gains by exploring and refining modifiers. Using only *Phase II* with BLIP initialization improves over the baseline but remains below the full two-phase system. These results show that the two stages are complementary and both are necessary, consistent with our query budget analysis. *Phase I* provides strong subject recovery that offers a high quality initialization for *Phase II* and enables the second stage to converge to a higher final value. *Phase II* in turn refines the subject inferred by *Phase I* and systematically augments it with appropriate modifiers, leading to the highest image similarity and semantic alignment across datasets.

J. Implementation Details

J.1. RL-Based Prompt Inversion

During imitation learning (IL), we freeze the BLIP backbone and train a lightweight adapter MLP to imitate expert next-token decisions. Each expert trajectory consists of (h_t, y_{t+1}) pairs, where h_t is the decoder hidden state of the partial prompt and y_{t+1} is the next ground-truth token sampled from BLIP's caption generation. In practice, we directly use BLIP to generate 10 expert trajectories per image. We train the adapter using cross-entropy loss on the decoder logits. Training is conducted for 2000 epochs with the Adam optimizer, a learning rate of 3×10^{-4}, and a batch size of 8. The input hidden dimension is 768, and the adapter MLP expands it to 1536 and projects it back to 768 dimensions using a two-layer architecture with ReLU activation.

After imitation pretraining, we fine-tune the model using the Proximal Policy Optimization (PPO) algorithm within the same prompt-generation environment. We initialize the actor and critic networks from the pretrained adapter checkpoint. The PPO hyperparameters are discount factor $\gamma = 0.98$ and clipping threshold $\epsilon = 0.2$. The actor and critic learning rates are 1×10^{-4} and 5×10^{-4}, respectively. We update the PPO agent every 150 environment steps, and perform $K = 4$ epochs of policy and value updates for each batch of collected trajectories. We train for up to 100 total epochs. For the shaped reward, we set the scaling coefficient β to 10.

J.2. Fuzz Testing-Powered Prompt Optimization

We adopt a query-budgeted fuzz testing optimization to explore the prompt space efficiently. The total query budget is set to 100 image generations per optimization run. During the first 30 queries, only the subject-related part of the prompt is mutated to refine the core semantic content. In the remaining 70 queries, both the subject and modifiers are jointly optimized to improve compositional richness and visual fidelity. For experiments on the MS COCO and Flickr datasets, we restrict all 100 queries to optimizing the subject component. The mutation process maintains a seed pool of 5 candidate prompts, which are iteratively sampled, expanded, and replaced based on their CLIP Similarity until the query budget is exhausted.

PROMPTMINER is implemented with Python 3.10 and PyTorch 2.6.0. We conducted all experiments on a Ubuntu 20.04 server equipped with 1 A100 SXM4 GPU.

K. Potential-Based Reward Shaping

We provide a formal analysis of the *potential-based reward shaping* mechanism used in the Section 3. We first show that under mild assumptions, the potential-based transformation preserves the optimal policy of the original MDP. We then derive how such shaping influences the learning dynamics by improving the reward density, gradient variance, and initialization of value estimates.

K.1. Setup and Definitions

We recall the MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, r, \gamma)$ defined in Section 3, where $\mathcal{S}$ denotes the state space, $\mathcal{A}$ the action space, $P(s' \mid s, a)$ the transition kernel, $r(s, a)$ the reward function, and $\gamma \in [0, 1)$ the discount factor. For the sake of theoretical completeness, we slightly generalize the reward function here to the form $r(s, a, s')$, which makes explicit its possible dependence on both the action a and the resulting next state s'. This modification is purely notational—conceptually equivalent to $r(s, a)$ used in Section 3—and serves to accommodate the upcoming definition of state-dependent shaping functions $F(s, a, s')$. Under this general form, the value and action-value functions of a policy π are expressed as

$$V_\pi(s) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t \, r(s_t, a_t, s_{t+1}) \;\middle|\; s_0 = s \right], \tag{13}$$

$$Q_\pi(s, a) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t \, r(s_t, a_t, s_{t+1}) \;\middle|\; s_0 = s,\ a_0 = a \right]. \tag{14}$$

To introduce reward shaping, we define a bounded *potential function* $\Phi : \mathcal{S} \to \mathbb{R}$, which assigns a scalar potential to each state, and construct the shaped reward

$$r'(s, a, s') = r(s, a, s') + F(s, a, s'), \quad \text{where} \quad F(s, a, s') = \gamma \Phi(s') - \Phi(s). \tag{15}$$

Intuitively, the term $F(s, a, s')$ measures the potential difference between successive states, scaled by the discount factor. Because this term depends solely on the states and not on the agent's policy, it modifies the learning dynamics without altering the underlying optimization objective, making it particularly suitable for theoretical analysis.

K.2. Theorem: Policy Invariance under Potential-Based Shaping

Theorem K.1. *Let V_π, Q_π be the original value functions and V'_π, Q'_π those under the shaped reward r'. Then for any bounded Φ, the following holds:*

$$V'_\pi(s) = V_\pi(s) - \Phi(s), \tag{16}$$
$$Q'_\pi(s,a) = Q_\pi(s,a) - \Phi(s), \tag{17}$$

and therefore

$$A'_\pi(s,a) = Q'_\pi(s,a) - V'_\pi(s) = A_\pi(s,a), \tag{18}$$

implying that $\arg\max_a Q'^*(s,a) = \arg\max_a Q^*(s,a)$*, i.e., the optimal policy is preserved.*

Proof. Starting from the shaped return:

$$G'_t = \sum_{k=0}^{\infty} \gamma^k \big[r_{t+k} + \gamma\Phi(s_{t+k+1}) - \Phi(s_{t+k})\big] \tag{19}$$
$$= \underbrace{\sum_{k=0}^{\infty} \gamma^k r_{t+k}}_{G_t} + \sum_{k=0}^{\infty} \big(\gamma^{k+1}\Phi(s_{t+k+1}) - \gamma^k\Phi(s_{t+k})\big). \tag{20}$$

The second summation is a telescoping series:

$$\sum_{k=0}^{n} (\gamma^{k+1}\Phi_{k+1} - \gamma^k\Phi_k) = -\Phi(s_t) + \gamma^{n+1}\Phi(s_{t+n+1}), \tag{21}$$

and taking the limit $n \to \infty$, the terminal term vanishes because Φ is bounded and $\gamma < 1$. Thus

$$G'_t = G_t - \Phi(s_t). \tag{22}$$

Taking expectations under policy π gives:

$$V'_\pi(s) = \mathbb{E}_\pi[G'_t \mid s_t = s] = V_\pi(s) - \Phi(s), \tag{23}$$

which proves (16). For Q'_π, by the Bellman definition under r':

$$Q'_\pi(s,a) = \mathbb{E}[r'(s,a,s') + \gamma V'_\pi(s')] \tag{24}$$
$$= \mathbb{E}[r(s,a,s') + \gamma\Phi(s') - \Phi(s) + \gamma(V_\pi(s') - \Phi(s'))] \tag{25}$$
$$= \mathbb{E}[r(s,a,s') + \gamma V_\pi(s')] - \Phi(s) = Q_\pi(s,a) - \Phi(s), \tag{26}$$

proving (17). Since subtracting $\Phi(s)$ does not depend on a, the greedy policy w.r.t. Q'_π is identical to that of Q_π.

K.3. Bellman Operator View and Corollary

Let $\mathcal{T}_\pi$ and $\mathcal{T}'_\pi$ denote the Bellman operators:

$$(\mathcal{T}_\pi V)(s) = \mathbb{E}_{a\sim\pi}[r(s,a,s') + \gamma V(s')]\,, \tag{27}$$
$$(\mathcal{T}'_\pi V)(s) = \mathbb{E}_{a\sim\pi}[r'(s,a,s') + \gamma V(s')]\,. \tag{28}$$

Then $\mathcal{T}'_\pi(V - \Phi) = \mathcal{T}_\pi(V) - \Phi$. Therefore, if V_π is a fixed point of $\mathcal{T}_\pi$, then $V_\pi - \Phi$ is a fixed point of $\mathcal{T}'_\pi$. The same holds for the optimal Bellman operator $\mathcal{T}$, so $V'^* = V^* - \Phi$ and $Q'^* = Q^* - \Phi$. This formalizes that potential shaping merely shifts the value function manifold by a state-dependent bias but leaves the contraction and optimality conditions invariant.

K.4. Potential-Based Reward Shaping under PPO Training

Proposition A.3 (PPO + GAE invariance under potential-based shaping). Let the shaped reward be

$$r'(s_t, a_t, s_{t+1}) \;=\; r(s_t, a_t, s_{t+1}) \;+\; \gamma\,\Phi(s_{t+1}) - \Phi(s_t), \tag{29}$$

with the same discount factor γ as the training objective. PPO employs a critic $\hat{V}_\theta$ and uses generalized advantage estimation (GAE) with temporal-difference residuals

$$\delta_t \;=\; r_t + \gamma\,\hat{V}_\theta(s_{t+1}) - \hat{V}_\theta(s_t), \qquad \widehat{A}_t \;=\; \sum_{l\geq 0} (\gamma\lambda)^l\,\delta_{t+l}. \tag{30}$$

For the shaped MDP, define the *shaped critic*

$$\hat{V}'_\theta(s) \;=\; \hat{V}_\theta(s) - \Phi(s), \tag{31}$$

and compute

$$\delta'_t \;=\; r'_t + \gamma\,\hat{V}'_\theta(s_{t+1}) - \hat{V}'_\theta(s_t), \qquad \widehat{A}'_t \;=\; \sum_{l\geq 0} (\gamma\lambda)^l\,\delta'_{t+l}. \tag{32}$$

Claim. For every trajectory and $\lambda \in [0, 1]$,

$$\delta'_t \equiv \delta_t, \qquad \widehat{A}'_t \equiv \widehat{A}_t.$$

Proof. Substitute $r'_t = r_t + \gamma\Phi(s_{t+1}) - \Phi(s_t)$ and $\hat{V}'_\theta = \hat{V}_\theta - \Phi$:

$$\begin{aligned}\delta'_t &= \underbrace{r_t + \gamma\Phi(s_{t+1}) - \Phi(s_t)}_{r'_t} + \gamma(\hat{V}_\theta(s_{t+1}) - \Phi(s_{t+1})) - (\hat{V}_\theta(s_t) - \Phi(s_t)) \\ &= r_t + \gamma\hat{V}_\theta(s_{t+1}) - \hat{V}_\theta(s_t) = \delta_t.\end{aligned}$$

Since GAE is a linear operator on the TD errors, $\widehat{A}'_t = \widehat{A}_t$ follows directly. □

Corollary A.4 (PPO surrogate and value loss are unchanged). The PPO clipped surrogate objective

$$\mathcal{L}^{\text{CLIP}}(\theta) = \mathbb{E}\Big[\min\Big(r_t(\theta)\,\widehat{A}'_t,\; \text{clip}\big(r_t(\theta), 1\pm\epsilon\big)\,\widehat{A}'_t\Big)\Big], \quad r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}, \tag{33}$$

is identical to the unshaped objective since $\widehat{A}'_t \equiv \widehat{A}_t$. The value regression loss is likewise invariant if both targets and predictions are shifted consistently:

$$(R'_t - \hat{V}'_\theta(s_t))^2 = ((R_t - \Phi(s_t)) - (\hat{V}_\theta(s_t) - \Phi(s_t)))^2 = (R_t - \hat{V}_\theta(s_t))^2. \tag{34}$$

Entropy regularization is unaffected. Hence, PPO updates on shaped data are mathematically identical to unshaped updates when the critic is shifted by $-\Phi$.

Remark. This proposition is the PPO/GAE analogue of the classical policy-invariance theorem of Ng, Harada, and Russell [23]. Potential-based shaping leaves the true advantages $A'_\pi = A_\pi$ unchanged, and with the critic shift $\hat{V}' = \hat{V} - \Phi$, it also leaves the *estimated* advantages $\widehat{A}'_t$ identical sample-by-sample.

K.5. How Shaping Can Accelerate PPO Training Without Changing What Is Optimal

If shaping is implemented exactly as above, the PPO actor and critic updates are algebraically identical to those on the unshaped MDP. Any speed-up therefore arises from improved learning dynamics rather than a change in the underlying objective. Below we summarize the PPO-specific mechanisms through which shaping can still help.

(1) Variance reduction via informed baselines. In actor–critic methods, any state-dependent baseline can be subtracted from returns without biasing the gradient. Choosing Φ correlated with returns provides an additional control-variate that reduces the variance of $\widehat{A}_t$, especially during early training when $\hat{V}_\theta$ is inaccurate. This can be realized by using the shaped critic $\hat{V}' = \hat{V} - \Phi$, or equivalently by adding $\Phi(s)$ to the baseline term in advantage computation.

(2) Finite-horizon credit assignment. PPO computes advantages over finite rollout segments. When partial-episode bootstrapping is used, the Φ terms cancel exactly (Proposition A.3). If truncated episodes are instead treated as terminal, shaping introduces a boundary term, providing denser feedback near segment boundaries and potentially improving early credit assignment—while still leaving the optimal policy unchanged.

(3) Critic warm-start and architectural priors. Setting the initial critic to $\hat{V}_0(s) \approx \Phi(s)$ or embedding $\Phi(s)$ as a fixed residual in the value head supplies a useful prior. Although the PPO gradients remain unbiased and unchanged in theory, better initialization of the critic often stabilizes optimization and accelerates the convergence of the actor–critic loop.

(4) Intentional partial shaping. Some implementations simply replace r with r' while keeping the critic unshifted. Then $\widehat{A}'_t \neq \widehat{A}_t$ early on, producing progress-aligned advantages that accelerate early learning. While this breaks the exact gradient equivalence, the optimal policy is still preserved by the theoretical invariance of potential-based shaping.

L. More Qualitative Results

Figure 9. Visualization of images generated by Stable Diffusion v1.5 compared with target image.

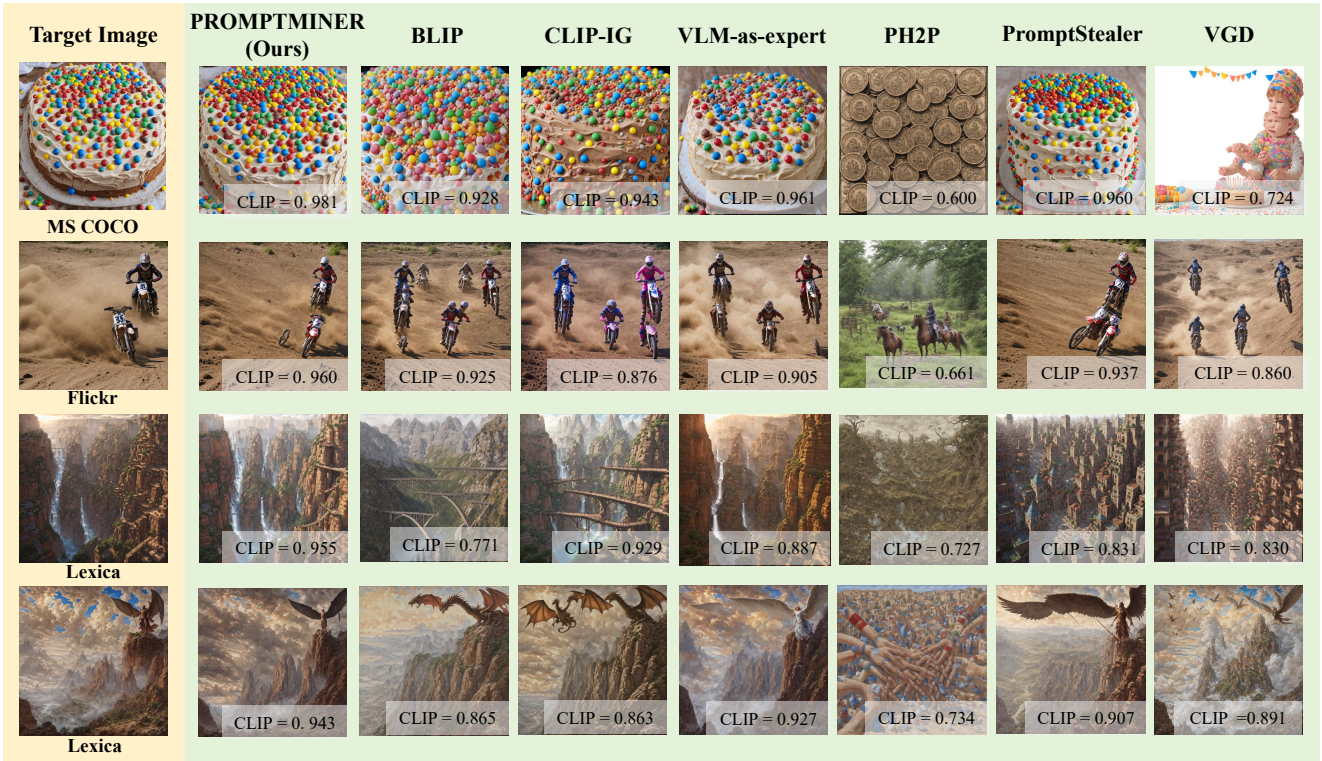


Figure 10. Visualization of images generated by SDXL Turbo compared with target image.

Figure 11. Visualization of images generated by Stable Diffusion 3.5 Medium compared with target image.

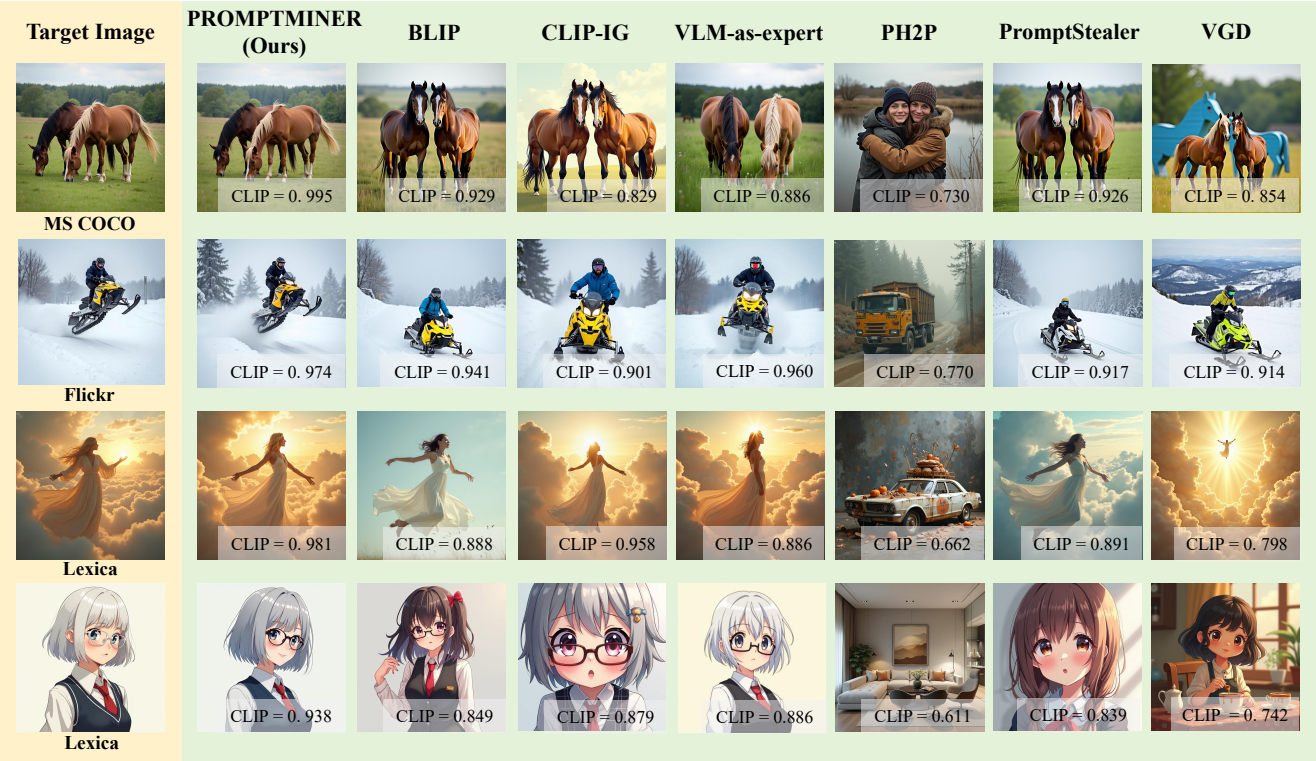


Figure 12. Visualization of mages generated by FLUX.1 dev compared with target image.

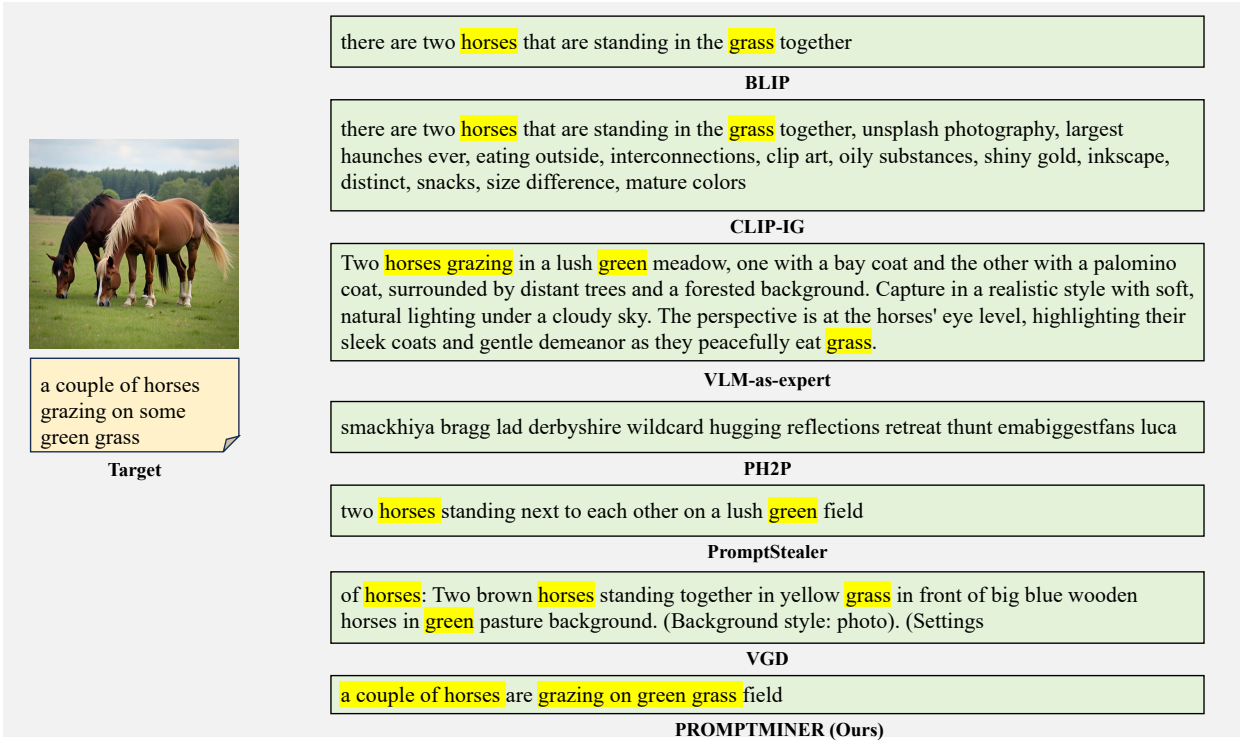


Figure 13. Qualitative Results of stolen prompts compared with target prompt on MS COCO.

A man wearing glasses sits with a drink at a table whilst reading the newspaper.

Target

man reading a newspaper and drinking a glass of beer

BLIP

man reading a newspaper and drinking a glass of beer, high quality fantasy stock photo, coffee beans, short dark blond beard, short brown hair with bangs, news broadcast, attractive man drinking coffee, graying hair, man standing, masculine, interconnections, editors choice, suspense

CLIP-IG

A man with glasses and a beard sipping tea while reading a newspaper, seated at a wooden table in a cozy cafe. The environment features warm, ambient lighting with subtle bokeh from hanging lights, creating a relaxed and intimate atmosphere. The man wears a casual denim shirt, and the perspective is a mid-close shot focusing on his contemplative expression.

VLM-as-expert

sheet thenburg impermilancutlery smile webdevelopment omd gfriend tengtengpurepure

PH2P

a man reading a newspaper and drinking a pint of beer

PromptStealer

about newspapers involving people reading holding or sitting with newsprint backgrounds People holding newspapers sitting reading news articles in coffee shop backgrounds overcast gray

VGD

a man wearing glasses sitting at a table, drinking coffee, reading the newspaper

PROMPTMINER (Ours)

Figure 14. Qualitative Results of stolen prompts compared with target prompt on Flickr.

portrait of a beautiful young fox female with wearing pirate attire. in style of yoji shinkawa and hyung - tae kim, trending on artstation, dark fantasy, great composition, concept art, highly detailed, dynamic pose, vibrant colours.

Target

there is a drawing of a fox with a pirate hat on

BLIP

there is a drawing of a fox with a pirate hat on, cgsociety portrait, krenzcushart, gritty coloring, inspired by Ruth Sanderson, featured on artstattion, by Nōami, by Eva Švankmajerová, rendered image, highly rendered, unknown artstyle, pirate woman, illustration]

CLIP-IG

Fox humanoid character with striking orange fur and white accents, wearing a black pirate hat and detailed historical attire with brown straps and a white shirt; the character stands confidently with fiery red-orange hair flowing behind; detailed digital art style with dramatic lighting emphasizing the vibrant colors, set against a clean, minimalist background; front-facing portrait perspective capturing the captivating eyes and intricate details of the costume.

VLM-as-expert

ssions tathumb italia pb ێssen expressing beater halifax hairdresser image

PH2P

portrait of a female fox pirate, intricate, sharp focus, digital painting, masterpiece

PromptStealer

of Fox style artwork featuring Foxes with vivid orange fur among some tall emerald cyberpunk 90s tinted misty

VGD

portrait of a beautiful female fox pirate artist rendering, a stylized, fox-like figure, with wearing a pirate hat and intricate attire, vibrant colors, highly detailed, dark fantasy, digital painting, concept art

PROMPTMINER (Ours)

Figure 15. Qualitative Results of stolen prompts compared with target prompt on Lexica.