# On the Multi-modal Vulnerability of Diffusion Models

**Dingcheng Yang** [1 *] **Yang Bai** [2 *] **Xiaojun Jia** [3] **Yang Liu** [3] **Xiaochun Cao** [4] **Wenjian Yu** [1]

## Abstract

Diffusion models have been widely deployed in various image generation tasks, demonstrating an extraordinary connection between image and text modalities. Although prior studies have explored the vulnerability of diffusion models from the perspectives of text and image modalities separately, the current research landscape has not yet thoroughly investigated the vulnerabilities that arise from the integration of multiple modalities, specifically through the joint analysis of textual and visual features. In this paper, we first visualize both text and image feature space embedded by diffusion models and observe a significant difference, i.e., the prompts are embedded chaotically in the text feature space, while in the image feature space they are clustered according to their subjects. Based on this observation, we propose MMP-Attack, which leverages multimodal priors (MMP) to manipulate the generation results of diffusion models by appending a specific suffix to the original prompt. Specifically, our goal is to induce diffusion models to generate a specific object while simultaneously eliminating the original object. Our MMP-Attack shows a notable advantage over existing studies with superior manipulation capability and efficiency. Our code is publicly available at `https://github.com/ydc123/MMP-Attack`.

## 1. Introduction

In recent years, diffusion models (Ho et al., 2020; Song et al., 2020) have revolutionized the field of image generation, achieving state-of-the-art results in both the diversity and quality of generated content. The advancement of vision-language models (Radford et al., 2021) has further enhanced the capabilities of diffusion models, giving rise to novel applications in text-to-image (T2I) generation (Rombach et al., 2022; Ramesh et al., 2022; Nichol et al., 2022; Saharia et al., 2022). However, existing studies have shown that diffusion models also exhibit vulnerability issues, where minor modifications to the original prompts can manipulate diffusion models to generate completely different image content. Zhuang et al. (2023) optimized a specific suffix to conduct untargeted attack and targeted erasing, namely generating random image content unrelated to the original prompt and omitting a specific category mentioned in the original prompt respectively. Liu et al. (2024) explored white-box targeted attack, while Maus et al. (2023) explored query-based targeted attack. However, they both demand a significant number of image generations, not only making it time-consuming but also unsuitable for commercial models due to their confidentiality or substantial monetary costs.

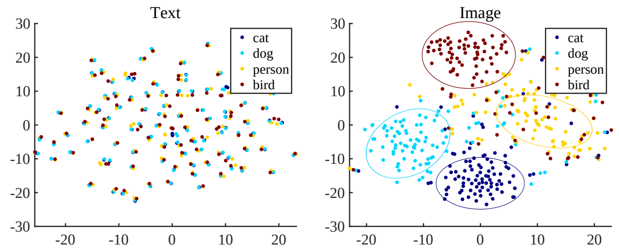


*Figure 1.* Visualization of 400 samples in text (left) and image (right) feature space embedded by Stable Diffusion v1.4 (SD v14). Text features are chaotic while image features are clustered.

Besides their unsatisfactory performance, existing studies are also limited by their designed algorithms solely on either text or image feature space. In (Zhuang et al., 2023), untargeted attack was achieved by maximizing the distance between the optimized prompt and the original prompt in the text feature space. In (Liu et al., 2024; Maus et al., 2023), the objective function was designed in the image feature space with an auxiliary image classifier, assessing whether the generated images contain objects of predefined categories. The lack of exploration across different modalities inspired us to visualize the text and image feature space within diffusion models simultaneously. A significant dif-

[*] Equal contribution [1] Dept. Computer Science & Tech., BNRist, Tsinghua University, Beijing, China [2] Tencent Technology (Beijing) Co.Ltd [3] Nanyang Technological University [4] Sun Yat-sen University, Shenzhen. Correspondence to: Wenjian Yu <yu-wj@tsinghua.edu.cn>.

*Proceedings of the 41st International Conference on Machine Learning*, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

ference between the two modalities is thus observed. As shown in Figure 1, we draw both text and image features of 400 samples embedded by Stable Diffusion v1.4 (SD v14), which are formed with 100 templates and 4 objects. Details of these samples are given in Section 3.2. As illustrated in Figure 1, it is evident that the prompts are embedded chaotically in the text feature space, while in the image feature space they are clustered according to their subjects. This phenomenon can be attributed to the fact that text features distribute their emphasis across a variety of words, consequently placing greater importance on the sentences or templates. In contrast, image features are more concentrated on the specific objects. As a result, incorporating features from both modalities is crucial for effectively manipulating diffusion models. This difference also highlights certain suboptimal alignments within existing diffusion models and the essence of utilizing multi-modal features, particularly from a robustness standpoint, which we will further analyse in Section 3.

Based on this observation, we propose an **MMP-Attack** by utilizing **M**ulti-**M**odal **P**riors. Our approach optimizes a suffix appended to the original prompt, aiming to effectively facilitates the generation of a desired target object by removing the original object, thus addressing the most challenging scenario. Specifically, we minimize the distance between the optimized prompt and the target category (to add) in both text and image feature space. The differences between MMP-Attack and existing works are summarized in Table 1.

*Table 1.* Comparison of existing methods with ours, based on the considered modality, targeted/untargeted setting, and whether image generation is required.

| Method | Modality | Targeted | Generation-free |
|---|---|---|---|
| Liu et al. (2024) | image | ✓ | × |
| Maus et al. (2023) | image | ✓ | × |
| Zhuang et al. (2023) | text | × | ✓ |
| **MMP-Attack (Ours)** | text+image | ✓ | ✓ |

The experimental results indicate that our MMP-Attack achieves a significantly higher attack success rate compared to the relevant works. Moreover, after analyzing the optimized suffix, we observed that MMP-Attack often works in a ***cheating*** way, which means that it often contains some tokens related to the target object. *It should be noted that simply appending the target object to the original prompt does not work.* Therefore, we also denote the suffix we optimize as a ***cheating suffix***.

The major contributions are summarized as follows.

- We conduct a visual analysis of both the text and image feature spaces that are embedded by diffusion models. Our work represents the first instance of observing the notable differences in features across multi-modalities. Such observations could potentially highlight a misalignment between the two modalities within diffusion models, particularly from the perspective of robustness.
- Based on the observations, we propose **MMP-Attack**, which leverages multi-modal priors to manipulate the generation results of diffusion models. This is achieved by appending a specific suffix after the original prompt, which often contains some tokens related to the target object, hence referred to as a *cheating suffix*.
- Experimental results indicate that our method achieves over 81.8% attack success rates on two open-source T2I models even with only four tokens, showcasing a notable advantage over existing works.

## 2. Related Work

### 2.1. Diffusion Models

Diffusion models have achieved remarkable success in the field of image generation through a learnable step-wise denoising process that transforms a simple Gaussian distribution into the data distribution (Ho et al., 2020). Some studies have been proposed to accelerating the image generation process (Song et al., 2020; Lu et al., 2022). Notably, by combining with the visual language model CLIP (Radford et al., 2021), the diffusion model showcases exceptional prowess in text-to-image generation (Rombach et al., 2022).

### 2.2. Manipulation in T2I Generation

Deep neural networks are known to be vulnerable (Szegedy et al., 2014; Zhao et al., 2021; Yang et al., 2023a;b; Bai et al., 2020; 2023). Recent studies have shown that the T2I generation process is vulnerable to prompts, indicating that it is possible to manipulate T2I models to generate images unrelated to the given prompt by adding a special suffix to the prompt (Liu et al., 2024; Maus et al., 2023; Zhuang et al., 2023). Liu et al. (2024) proposed a white-box method, which assumes that the diffusion model is fully known, making it unsuitable for confidential commercial models. Maus et al. (2023) performed a high-cost query-based method. The practicality of both approaches is limited. Zhuang et al. (2023) assumed that the diffusion model has a white-box CLIP model but an inaccessible and unqueryable generative model. Under this assumption, they proposed a generation-free method against T2I models, which employed a genetic algorithm to manipulate the CLIP model. However, they only considered untargeted attack and targeted erasing. In this paper, we follow the setting outlined in (Zhuang et al., 2023) but address a more challenging task: targeted manipulation, specifically by adding target objects while removing original objects in original prompts. Our experimental results demonstrate a significant improvement over (Zhuang et al., 2023).

It is important to clarify that the manipulation in T2I generation studies different topics from safety in T2I generation, which aim to construct a prompt to make the diffusion model generate inappropriate content, such as Not-Safe-For-Work (NSFW) content (Yang et al., 2024a;b; Tsai et al.) or infringing content (Zhang et al., 2023). Although both goals are to make the generated image contain specific content, we need to optimize a suffix for an original prompt, thus investigating the robustness of diffusion models. Additionally, the presence of the original prompt makes our research problem more difficult.

## 3. Observations on Multi-modal Features within Diffusion Models

### 3.1. Preliminary: Pipeline of Diffusion Model

Given that the vocabulary of candidate tokens forms a set $\mathbb{V} = \{w_1, w_2, \cdots, w_L\}$ where $L$ represents the number of tokens in the vocabulary $\mathbb{V}$, an input prompt can be expressed as $s \in \mathbb{V}^*$. A well-trained diffusion model consists of two components: a CLIP model and a generative model $G$. The CLIP model includes an image encoder $F^i$, which takes an image as input and outputs a $d_{\text{emb}}$-dimensional image embedding vector. It also includes a token embedder $E_\psi$ and a text encoder $F^t$, which together embed a text prompt into a $d_{\text{emb}}$-dimensional text embedding vector. Here, $\psi \in \mathbb{R}^{|\mathbb{V}| \times d_{\text{token}}}$ serves as an embedding codebook. For the input prompt $s$, $E_\psi(s)$ is a matrix of shape $|s| \times d_{\text{token}}$, where $E_\psi(s)_i = \psi_j$, with the condition that $w_j = s_i$. This token embedding matrix $E_\psi(s)$ is then input into the text encoder $F^t$ and embedded as a $d_{\text{emb}}$-dimensional text embedding vector. During the training stage, an image is transformed to an image embedding vector by the image encoder. Simultaneously, its caption (text data) is transformed to a text embedding vector by the token embedder and the text encoder. The distance between the two vectors is minimized to enable the CLIP model to align the image space and text space. During the T2I generation stage, the input prompt $s$ is first embedded into a text embedding vector $v$ by the token embedder and text encoder. Then, it is input into the subsequent generative model $G$ to sample $x \sim G(v)$, where $G(v)$ is a probability distribution conditioned on $v$, and $x$ represents a sampled image. Thus, the T2I generation from the input prompt $s$ can be understood as a process of sampling from the probability distribution $x \sim G(F^t(E_\psi(s)))$.

### 3.2. Multi-modal Features in Diffusion Models

Previous studies have separately investigated the vulnerability of diffusion models from the perspectives of text and image modalities (Zhuang et al., 2023; Liu et al., 2024). In contrast to their studies, we investigate the vulnerability of multi-modal features. Given a prompt $s$, we define its text embedding vector as $F^t(E_\psi(s))$, and its image embedding vector as $F^i(x)$, where $x \sim G(F^t(E_\psi(s)))$. Then, we visualize the text and image feature spaces, showcasing a marked distinction between the multi-modalities.

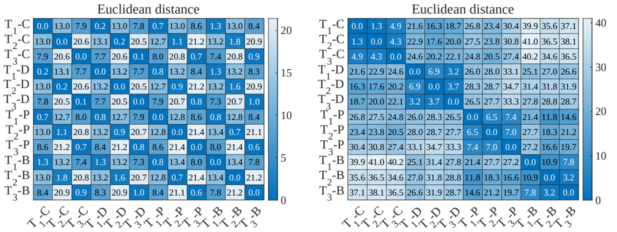


*Figure 2.* Euclidean distances between 12 different prompts in the text (left) and image (right) feature spaces. The prompts are generated from 3 different templates: ‘a {noun} is sitting on a bench in a park’, ‘a {noun} is peeking out from behind a curtain’, and ‘a {noun} is standing at the edge of a cliff’, denoted as $T_1$, $T_2$, and $T_3$, respectively. ‘-C’, ‘-D’, ‘-P’, and ‘-B’ represent the {noun} being cat, dog, person, and bird respectively.

**Chaos Effect of Features in Text Space.** We first visualize the text feature space. We instructed ChatGPT to generate 100 prompt templates, and then sequentially filled in ‘`cat`’, ‘`dog`’, ‘`bird`’, and ‘`person`’ sequentially as subjects to form 400 prompts. Then, we embedded these 400 prompts into text embedding vectors by the SD v14 and visualized them in the text feature space using t-SNE. The visualization results is shown in Figure 1(a), illustrating that prompts associated with different subjects are mixed together. This is because both the subjects and other tokens are considered important by the text encoder. Thus, in the text feature space, prompts with different subjects but originating from the same template can be embedded close together. This implies that even if two prompts are close in the text feature space, they may have different subjects. To illustrate this phenomenon more clearly, we chose 12 prompts originating from 3 templates and calculated their Euclidean distances from each other, as shown in Figure 2(a).

**Clustering Effect of Features in Image Space.** Then, we use SD v14 to visualize these 400 prompts on the image feature space, as shown in Figure 1(b). It can be observed that the embedding vectors of these prompts have remarkably different distributions on the image feature space than on the text feature space, revealing the potential misalignment between text feature space and image feature space for diffusion models. Specifically, the prompts with the same subject are clustered together in the image feature space, while prompts with different subjects are distinguished from each other. This difference arises because the text encoder extracts features of all tokens in the prompts, while an image encoder primarily extracts features of the key object (subject) in the images. Thus, the prompts that are close in the image feature space often share the same subject. This can be evidenced in Figure 2(b), where the distances of prompt pairs with the same subject are significantly lower (up to 10.9) in the image feature space.

## 4. Methodology

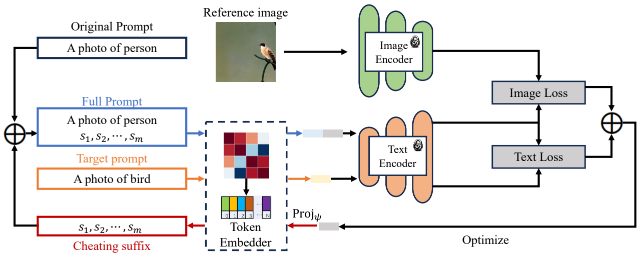


*Figure 3.* An illustration of the proposed MMP-Attack flow.

In this part, we propose MMP-Attack, which leverages multi-modal priors to targeted manipulate the T2I generation. We begin by formulating the targeted manipulation problem for T2I models. Then, motivated by the misalignment phenomenon observed in Section 3, we propose an optimization objective that simultaneously considers both the image and text modalities. Finally, we present the corresponding optimization approach. An illustration of our MMP-Attack is shown in Figure 3.

### 4.1. Problem Formulation

Let $s_o \in \mathbb{V}^n$ be the original prompt containing $n$ tokens, and $m$ be the number of tokens in the cheating suffix. The cheating suffix to be optimized can be represented as $s_a \in \mathbb{V}^m$, which will be concatenated with $s_o$ to get the full prompt $s_o \oplus s_a \in \mathbb{V}^{n+m}$, where the operator $\oplus$ denotes concatenation operator. For conducting targeted manipulation, we assume that there is a target category $t \in \mathbb{V}$ (e.g., `dog`, `bird`), which is irrelevant to $s_o$. We need to search for a cheating suffix that, when concatenated with the original prompt $s_o$, guides the T2I diffusion model to generate an image containing the target category but is unrelated to $s_o$. The optimization objective is as follows:

$$\text{argmax}_{s_a} \mathbb{E}_{x \sim G(F^t(E_\psi(s_o \oplus s_a)))} \mathcal{A}(x, t, s_o) \ , \tag{1}$$

The $\mathcal{A}(x, t, s_o)$ is an evaluation metric to assess the manipulation performance. Following the assumptions of relevant work (Zhuang et al., 2023), we have access only to the CLIP model and are blind to the generative model $G$.

### 4.2. Optimization Approach

Directly solving (1) is infeasible, because it involves a generative model $G$ that is unknown in our assumption. An alternative approach is to first construct a target vector $v_t$ that provides a favorable solution to the following optimization objective:

$$\text{argmax}_{v_t} \mathbb{E}_{x \sim G(v_t)} \mathcal{A}(x, t, s_o) \ . \tag{2}$$

Assuming such a $v_t$ exists, we can achieve a favorable solution to (1) by maximizing the similarity between the text embedding vectors of $s_o \oplus s_a$ and target vector $v_t$. Consequently, the optimization objective (1) is transformed into a simplified problem involving only $F^t$ and $E_\psi$:

$$\text{argmax}_{s_a} \cos(F^t(E_\psi(s_o \oplus s_a)), v_t). \tag{3}$$

Although $G$ is unknown, constructing a favorable solution for problem (2) is not difficult, since we can use some heuristics solutions. For example, the images generated by a manually crafted prompt $s'$ =‘`a photo of` $t$’ will undoubtedly satisfy the requirements of our targeted manipulation. Thus, we can utilize its text embedding vector $v_t^{text} = F^t(E_\psi(s'))$ as a target vector to guide the optimization.

However, as demonstrated in Section 3, even though prompts have relatively close distances in the text feature space, the resulting images could be far apart in the image feature space, indicating differences in key objects present in the images. Thus, we integrate image modal information with text modal to guide the optimization process. Since the generative model $G$ is unknown, we can not compute loss terms for generated images, as done in prior works (Liu et al., 2024; Maus et al., 2023). Instead, we propose a target vector based on image modality. This approach also avoids the costly image generation required in prior work that utilized the image modality. Specifically, given a reference image $x_t$ containing the target category, we calculate its image embedding vector $v_t^{image} = F^i(x_t)$, where $F^i$ is the image encoder of the CLIP model. The CLIP model possesses the characteristic that image-text pairs with higher correlation exhibit larger cosine similarities in their embedding vectors. Therefore, $v_t^{image}$ is also a favorable solution for (2). Finally, we concurrently optimize in both the image and text modalities. The optimization objective is as follows:

$$\begin{aligned} &\text{argmax}_{s_a} \cos(v, v_t^{image}) + \lambda \cos(v, v_t^{text}), \\ &\quad \text{s.t.} \quad v = F^t(E_\psi(s_o \oplus s_a)), \end{aligned} \tag{4}$$

where $\lambda$ is a weighting factor to balance the loss terms between the image and text modalities.

The remaining challenge lies in solving (4), which is a non-differentiable optimization problem. To address this issue, a commonly used technique is Straight-Through Estimation (STE) technique (Bengio et al., 2013), which introduces a differentiable function $\text{sg}(\cdot)$ that is defined as the identity function during forward propagation and has zero partial derivatives. We leverage the $\mathrm{sg}(\cdot)$ function to solve (4). Specifically, we optimize the token embedding matrix $Z \in \mathbb{R}^{m \times d_{\text{token}}}$ of the cheating suffix, and define a differentiable function $\text{Proj}_{\psi} : \mathbb{R}^{m \times d_{\text{token}}} \to \mathbb{R}^{m \times d_{\text{token}}}$, where $\text{Proj}_{\psi}(Z)_i = Z_i + \mathrm{sg}(\psi_j - Z_i)$ such that $j = \mathrm{argmin}_{j'} \|\psi_{j'} - Z_i\|_2^2$. Notice that each row in matrix $\text{Proj}_{\psi}(Z)$ corresponds to an entry in the codebook $\psi$, therefore we can decode the cheating suffix $s_a = E_{\psi}^{-1}(\text{Proj}_{\psi}(Z))$. Moreover, due to the property $E_{\psi}(s_o \oplus s_a) = E_{\psi}(s_o) \oplus E_{\psi}(s_a)$, (4) can be reformulated into the following optimization problem:

$$\begin{aligned} &\mathrm{argmax}_Z \cos(v, v_t^{image}) + \lambda \cos(v, v_t^{text}) \\ &\quad \text{s.t.} \quad v = F^t(E_{\psi}(s_o \oplus s_a)) \\ &\qquad\qquad = F^t(E_{\psi}(s_o \oplus E_{\psi}^{-1}(\text{Proj}_{\psi}(Z)))) \\ &\qquad\qquad = F^t(E_{\psi}(s_o) \oplus \text{Proj}_{\psi}(Z)). \end{aligned} \tag{5}$$

Because the Proj function is differentiable, (5) can be solved using a gradient-based optimizer, providing better performance compared to prior work (Zhuang et al., 2023) that employs an zero-order optimizer.

We summarized the optimization approach in Algorithm 1. The target conditional vectors are first calculated in Step 1-3. Then, the optimization variable $Z$ is initialized and optimized by a gradient descent algorithm (Step 4-13). Finally, the cheating suffix is decoded based on $Z$ (Step 14).

**Algorithm 1** MMP-Attack

**Input:** token embedder $E_{\psi}$, dimension of the token embedding vector $d_{\text{token}}$, text encoder $F^t$, image encoder $F^i$, learning rate $\eta$, number of iterations $N$, original prompt $s_o$, number of tokens in cheating suffix $m$, target category $t \in \mathbb{V}$, weighting factor $\lambda$, a reference image $x_t$ containing the target category $t$ and unrelated to original prompt $s_o$.
**Output:** Cheating suffix $s_a$.

```
 1: $v_t^{image} \leftarrow F^i(x_t)$.
 2: $s' \leftarrow$ ‘a photo of $t$’.
 3: $v_t^{text} = F^t(E_{\psi}(s'))$
 4: Initialize $Z \in \mathbb{R}^{m \times d_{\text{token}}}$.
 5: $bestloss \leftarrow \infty, bestZ \leftarrow Z$
 6: for $i \leftarrow 1$ to $N$ do
 7:     $v \leftarrow F^t(E_{\psi}(s_o) \oplus \text{Proj}_{\psi}(Z))$.
 8:     $\mathcal{L} = -\cos(v, v_t^{image}) - \lambda \cos(v, v_t^{text})$.
 9:     if $bestloss > \mathcal{L}$ then
10:         $bestloss \leftarrow \mathcal{L}, bestZ \leftarrow Z$.
11:     end if
12:     $Z \leftarrow Z - \eta \nabla_Z \mathcal{L}$.
13: end for
14: $s_a \leftarrow E_{\psi}^{-1}(\text{Proj}_{\psi}(bestZ))$.
```

# 5. Experiments

## 5.1. Settings

**Dataset.** Five object categories are selected from the Microsoft COCO dataset (Lin et al., 2014), namely `car`, `dog`, `person`, `bird`, and `knife`. They are considered as both the original and target categories, forming a total of $5 \times 4 = 20$ distinct category pairs. For each category pair, a cheating suffix is generated. Each cheating suffix is then used to generate 100 images to evaluate the manipulation performance metrics. The final performance metrics are obtained by averaging across all categories, which means that for a given method, its performance metrics are calculated over $5 \times 4 \times 100 = 2000$ images.

**Models.** Following the setting in relevant work (Zhuang et al., 2023), we initially employ Stable Diffusion v1.4 (SD v14)[1] as the diffusion model for image generation and performance evaluation. This model utilizes a pretrained CLIP model[2], which is trained on a dataset containing text-image pairs (Thomee et al., 2016). Furthermore, we also consider an additional model, Stable Diffusion v2.1 (SD v21)[3], which has a distinct CLIP model[4] compared to SD v14. Finally, we also consider a commercial T2I service, i.e., DALL-E 3 (Betker et al., 2023) and Imagine Art[5].

**Evaluation Metrics.** The following metrics are considered to evaluate the manipulation performance: 1) **CLIP score**: We use the CLIP (Radford et al., 2021) model to calculate the embedding vectors for the generated image and the prompt (‘`a photo of` $t$’), subsequently determining their matching score based on cosine similarity. 2) **BLIP score**: BLIP (Li et al., 2022) is a better visual-language model. We use it to compute the image-text matching score. 3) **Original Category Non-Detection Rate (OCNDR)**: A binary metric where we employ an object detection model to examine if the generated image fails to detect objects of the original category, indicating an untargeted manipulation. 4) **Target Category Detection Rate (TCDR)**: Similar to OCNDR, it is a binary metric where we use an object detection model to check if the generated image contains objects of the target category. 5) **BOTH**: A binary metric where the value is 1 if and only if both OCNDS and TCDR are 1. A pretrained faster R-CNN model (Ren et al., 2015) with a ResNet-50-FPN backbone (Lin et al., 2017) is utilized as the object detection model to evaluate OCNDR, TCDR and BOTH, which is publicly available at torchvision[6].

[1] https://huggingface.co/CompVis/stable-diffusion-v1-4

[2] https://huggingface.co/openai/clip-vit-large-patch14

[3] https://huggingface.co/stabilityai/stable-diffusion-2-1

[4] https://huggingface.co/laion/CLIP-ViT-H-14-laion2B-s32B-b79K

[5] https://www.imagine.art/

[6] https://download.pytorch.org/models/fasterrcnn_resnet50_fpn_v2_coco-dd69338a.pth

### 5.2. Main Results

We utilize our MMP-Attack to optimize cheating suffixes, each comprising four tokens ($m = 4$). For comparative purposes, we consider three baseline methods : 1) **No attack**, meaning no cheating suffix is added; 2) **Random**, where four tokens are randomly chosen to form the cheating suffix; 3) **Genetic** (Zhuang et al., 2023), a method that proposed a genetic algorithm for untargeted attack, aiming to maximize the distance in text feature space from the original prompt. It can be directly extended as a baseline for targeted attack by minimizing the distance in text feature space from the target prompt $s'$ =‘a photo of $t$’. The hyper-parameters for the attack methods are presented in the Appendix A, while ablation studies provided in Appendix D.

Attacking results against SD v14 and SD v21 are listed in Table 2, which first shows that all baselines are relatively weak. Then, Table 2 also demonstrates a substantial superiority of MMP-Attack over the baselines. Specifically, for BOTH score, MMP-Attack surpasses the strongest baseline Genetic by 67.6% and 80.9% on SD v14 and SD v21, respectively. This metric offers an intuitive reflection of attack success rates, requiring the generated images not only exclude the original category but also contain the target category.

*Table 2.* Results of different methods against SD v14/v21. The metrics are defined in Sec. 5.1. **Best results are boldfaced.**

| Model | Method | CLIP | BLIP | OCNDR | TCDR | BOTH |
|---|---|---|---|---|---|---|
| SD v14 | No attack | 0.200 | 0.014 | 1.6% | 0.9% | 0.0% |
| | Random | 0.202 | 0.013 | 2.4% | 1.3% | 0.1% |
| | Genetic | 0.223 | 0.066 | 19.7% | 27.4% | 14.2% |
| | MMP-Attack (ours) | **0.265** | **0.414** | **92.0%** | **87.2%** | **81.8%** |
| SD v21 | No attack | 0.204 | 0.019 | 5.0% | 1.6% | 0.1% |
| | Random | 0.203 | 0.015 | 5.4% | 1.9% | 0.6% |
| | Genetic | 0.206 | 0.021 | 18.7% | 11.1% | 5.5% |
| | MMP-Attack (ours) | **0.270** | **0.429** | **95.2%** | **91.0%** | **86.4%** |

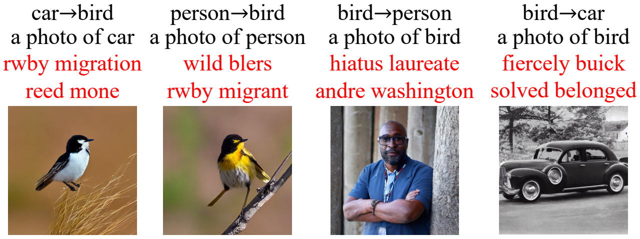


*Figure 4.* Examples of optimized cheating suffixes (marked in red) and their corresponding generated images on SD v14.

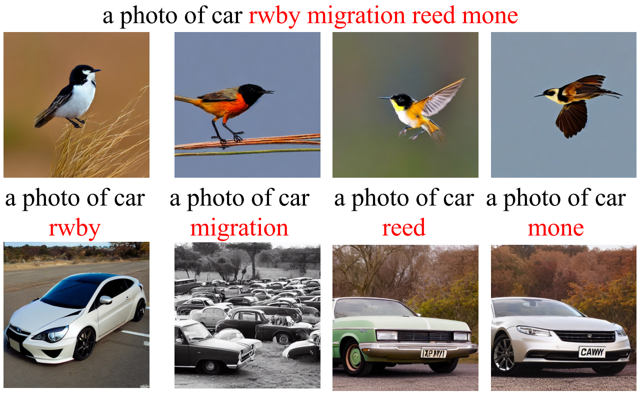


*Figure 5.* The images generated by SD v14 using different cheating suffixes (marked in red). The top four images are generated using the cheating suffix we optimized. The bottom four images are respectively generated using each of the four individual tokens as the cheating suffix.

Then, we present some results in Figure 4. More results are presented in Appendix B. By analyzing these suffixes, we observe that MMP-Attack automatically identifies specific tokens to achieve the manipulation goal. The identified tokens could be relevant words associated with the target object. For example, when the target category is `car`, MMP-Attack can automatically identify `buick`. The resulting cheating suffixes not only guide the T2I model to generate the desired objects but also lead it to ignore the original prompt. Moreover, in the task of targeting `car` to `bird`, all four tokens are unrelated to birds. Thus, when using `rwby`, `migration`, `reed` and `mone` as cheating suffixes separately, the T2I model generates images of cars (see Figure 5). However, when using all four tokens simultaneously, it generates images of birds. This constitutes a more imperceptible form of manipulation, thus bypassing simple filtering-based defense methods. Furthermore, we demonstrate that the optimized cheating suffixes exhibit universality and transferability in Appendix C.

## 6. Conclusions

In this paper, we analyze the vulnerability of diffusion models from a novel perspective of multi-modality. We are the first to observe a significant misalignment between the two modalities, particularly from the perspective of robustness. We further find that the text encoder spreads its attention across different words within a sentence and is therefore less sensitive to the main object. In contrast, image features are clearly clustered with their objects, showing a clear focus on words related to the objects. Motivated by this observation, we propose **MMP-Attack**, which leverages multi-modal priors (MMP) to targeted manipulate the generation results of diffusion models. The proposed MMP-Attack exhibits extraordinary performance, demonstrating not only high attack success rates but also superior universality and transferability. Our work contributes to a deeper understanding of T2I generation and establishes a novel paradigm for adversarial studies in AI-generated content (AIGC).

## 7. Social Impact Statements

This work first observed a modality misalignment phenomenon in text-to-image (T2I) diffusion models and, based on this, proposed a method for targeted manipulating T2I generations. It not only contributes to the understanding of vulnerabilities in T2I generations but also provides insights into boosting these systems against potential attacks.

## References

Bai, Y., Zeng, Y., Jiang, Y., Wang, Y., Xia, S.-T., and Guo, W. Improving query efficiency of black-box adversarial attack. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*, pp. 101–116. Springer, 2020.

Bai, Y., Wang, Y., Zeng, Y., Jiang, Y., and Xia, S.-T. Query efficient black-box adversarial attack on deep neural networks. *Pattern Recognition*, 133:109037, 2023.

Bengio, Y., Léonard, N., and Courville, A. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

Betker, J., Goh, G., Jing, L., Brooks, T., Wang, J., Li, L., Ouyang, L., Zhuang, J., Lee, J., Guo, Y., et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2023.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Li, J., Li, D., Xiong, C., and Hoi, S. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pp. 12888–12900. PMLR, 2022.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 740–755. Springer, 2014.

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117–2125, 2017.

Liu, Q., Kortylewski, A., Bai, Y., Bai, S., and Yuille, A. Discovering failure modes of text-guided diffusion models via adversarial search. In *The Twelfth International Conference on Learning Representations*, 2024.

Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C., and Zhu, J. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35:5775–5787, 2022.

Maus, N., Chao, P., Wong, E., and Gardner, J. R. Black box adversarial prompting for foundation models. In *The Second Workshop on New Frontiers in Adversarial Machine Learning*, 2023.

Nichol, A. Q., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., Mcgrew, B., Sutskever, I., and Chen, M. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *International Conference on Machine Learning*, pp. 16784–16804. PMLR, 2022.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

Ren, S., He, K., Girshick, R., and Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35: 36479–36494, 2022.

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2020.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *ICLR*, 2014.

Tashiro, Y., Song, Y., and Ermon, S. Diversity can be transferred: Output diversification for white-and black-box attacks. *Advances in neural information processing systems*, 33:4536–4548, 2020.

Thomee, B., Shamma, D. A., Friedland, G., Elizalde, B., Ni, K., Poland, D., Borth, D., and Li, L.-J. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016.

Tsai, Y.-L., Hsu, C.-Y., Xie, C., Lin, C.-H., Chen, J. Y., Li, B., Chen, P.-Y., Yu, C.-M., and Huang, C.-Y. Ring-a-bell! how reliable are concept removal methods for diffusion models? In *The Twelfth International Conference on Learning Representations*.

Yang, D., Xiao, Z., and Yu, W. Boosting the adversarial transferability of surrogate models with dark knowledge. In *2023 IEEE 35th International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 627–635. IEEE, 2023a.

Yang, D., Yu, W., Xiao, Z., and Luo, J. Generating adversarial examples with better transferability via masking unimportant parameters of surrogate model. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pp. 01–08, 2023b.

Yang, Y., Gao, R., Wang, X., Ho, T.-Y., Xu, N., and Xu, Q. Mma-diffusion: Multimodal attack on diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7737–7746, 2024a.

Yang, Y., Hui, B., Yuan, H., Gong, N., and Cao, Y. Sneakyprompt: Jailbreaking text-to-image generative models. In *2024 IEEE Symposium on Security and Privacy (SP)*, pp. 123–123. IEEE Computer Society, 2024b.

Zhang, Y., Tzun, T. T., Hern, L. W., Wang, H., and Kawaguchi, K. On copyright risks of text-to-image diffusion models. *arXiv preprint arXiv:2311.12803*, 2023.

Zhao, Z., Liu, Z., and Larson, M. On success and simplicity: A second look at transferable targeted attacks. *Advances in Neural Information Processing Systems*, 34: 6115–6128, 2021.

Zhuang, H., Zhang, Y., and Liu, S. A pilot study of query-free adversarial attack against stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2384–2391, 2023.

# A. Implementation Details

*Table 3.* List of filtered words. The first row represents the target category, followed by the next 20 rows representing the corresponding filtered words.

| car | dog | person | bird | knife |
|---|---|---|---|---|
| car | dog | person | bird | knife |
| cars | dogs | people | birds | knives |
| vehicle | cat | persons | birdie | fork |
| vehicles | dawg | woman | birdies | sword |
| dog | doggy | ppl | phone | blade |
| bus | puppy | guy | fish | wrench |
| boat | dogg | peoples | cat | gun |
| automobile | doggo | someone | bee | tool |
| train | doggie | adult | eagle | guns |
| van | cats | individual | flight | snakes |
| bike | horse | thing | birding | inmate |
| coach | animal | player | horse | weapons |
| er | pooch | man | birdman | pistol |
| sedan | car | member | crow | stabbing |
| i | dawgs | girl | dot | chair |
| plane | pup | personal | wildlife | spoon |
| cat | dad | personality | birdwatching | goalie |
| phone | adog | somebody | knowledge | bike |
| road | pet | members | plant | stab |
| suv | hotdog | child | lizard | skateboard |

The Adam optimizer is employed for searching cheating suffix, which are composed of four tokens ($m = 4$). The learning rate is set to 0.001 and the number of optimization iterations is set to 10000. For a single category pair, MMP-Attack takes approximately 6 minutes to run on a single Nvidia RTX 4090 GPU. The synonym initialization method is employed by default, with $\lambda$ set to 0.1 as the default weighting factor. The reference images used to calculate the loss term for image modality are presented in Figure 6.

To ensure the naturalness of the cheating suffix, we refined the vocabulary $\mathbb{V}$ to include only English words that end with the '</w>' symbol, indicating a white-space. This step was necessary because the CLIP vocabulary includes tokens representing prefixes that do not end with '</w>', and the concatenation of such tokens could result in the optimized cheating suffix containing non-existent words, thereby reducing the naturalness of the prompt. Furthermore, we additionally filtered out the top-20 synonyms of the target category from the vocabulary, to simulate real-world systems that block sensitive words. Specifically, the embedding codebook $\psi$ was employed to define the similarity between two tokens $w_i, w_j$ as $\cos(\psi_i, \psi_j)$, where $\cos(a, b) = \frac{a^T b}{\|a\|\|b\|}$ represents the cosine similarity between two vectors. Table 3 lists the words that are filtered out for each target category. These filtered words are mostly synonyms of the target category, or otherwise words with strong relevance. This filtering process mimics the use of a sensitive word filtering system commonly employed in real-world application.

A good initialization (Step 4 of Algorithm 1) often helps reduce the complexity of the optimization problem, leading to better solutions (Tashiro et al., 2020). To solve (5), we consider three initialization methods:

1. **EOS**: Initialize all $Z_i$ as the token embedding for `[eos]`, where `[eos]` is a special token in CLIP vocabulary representing the end of string.

2. **Random**: Randomly sample $m$ tokens from the filtered vocabulary and use their embeddings as the initial values for $Z$.

3. **Synonym**: Select the token with the highest cosine similarity to the target category $t$ in the filtered vocabulary, and use its token embedding as the initial values for all $Z_i$.

In (Zhuang et al., 2023), the number of generation step is set to 50, the number of candidates per step is set to 20, and the length of the cheating suffix is only set to 5 characters. Since this paper focuses on the more challenging targeted attack task, we set the number of generation step to 500. This implies a total of $500 \times 20 = 10000$ forward propagations, which also ensures fairness in computational cost comparison with MMP-Attack. Considering that the cheating suffix in (Zhuang et al., 2023) has a length of only 5 characters, which is usually shorter than the length of four tokens we used. To be fair, we employ the genetic algorithm to search for cheating suffix of length 32. This length exceeds the average character length of cheating suffixes searched by MMP-Attack.

# B. Display of all searched cheating suffixes

We present all the discovered cheating suffixes on SD v14 and SD v21 in Table 4.

# C. Universality and Transferability

We have shown that, the optimized cheating suffix $s_a$ can overwrite the content of original prompt $s_o$ and generates an image of the target category $t$. By observing Figure 4, it can be noticed that the two cheating suffixes discovered for the target category `bird` contain similar tokens, namely both include `rwby`, and one includes `migrant` while the other includes `migration`. This inspires us to explore whether the cheating suffix optimized for one category may be effective for other category pairs within the same target category, referred to as universality. We first attempt to append the

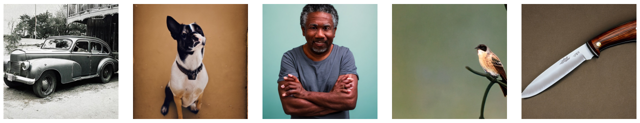


*Figure 6.* Reference images.

*Table 4.* All results of searched cheating suffixes. 'Ori. Cat.' means 'Original Category'.

| Model | Ori. Cat. | car | person | bird | dog | knife |
|---|---|---|---|---|---|---|
| SD v14 | car | - | physician qualified darryl atf | rwby migration reed mone | mutt portrait scout lao | skinner buck durable dagger |
| | person | transmission solved belonged coupe | - | wild blers rwby migrant | analog mutt pocket wilbur | crafted smoked durable gerber |
| | bird | fiercely buick solved belonged | hiatus laureate andre washington | - | since kiddo chihuahua gge | gerber outdoor laminated dagger |
| | dog | lewes automotive deluxe survives | hall actor transitions denzel | moth frid rwby tit | - | gazaunderattack rosewood transitional gerber |
| | knife | wartime neglected automotive wagon | denzel bipolar libertarian peterson | favorable bul reed tit | terriers staffers portrait django | - |
| SD v21 | car | - | dialogue resident ronald coleman | brian cumin tern hummingbird | tongue nose pied terrier | dmitry authentic pland bowie |
| | person | creole dub oldsmobile extinct | - | jharkhand tern finch migration | chihuahua shout merit terrier | pioneer hunter finn cutlery |
| | bird | unsolved creole forged automotive | tions founder willie rence | - | boston chihuahua photography shout | hunter bur exam bowie |
| | dog | lyle pontiac creole automotive | voices fellows melvin browne | vo tern detached finch | - | authentic topaz hunter petty |
| | knife | protected creole oldsmobile abroad | african equity veterans actor | flax programme tree finch | tongue pied chihuahua terrier | - |

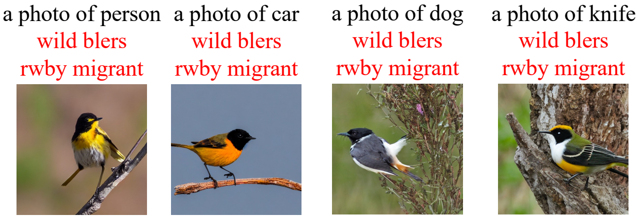


*Figure 7.* Examples of universality on SD v14. The cheating suffix marked in red is optimized with the original category `person` and the target category `bird`. It works well on different original prompts.

*Table 5.* Universal attack success rates of MMP-Attack against SD v14. The value in each cell is obtained by averaging BOTH score across the other three categories, excluding the original category (corresponding to the row) and the target category (corresponding to the column), over a total of 3 × 100 generated images.

| | car | person | bird | dog | knife |
|---|---|---|---|---|---|
| car | - | 66.0% | 54.7% | 52.3% | 88.7% |
| person | 58.3% | - | 93.3% | 41.3% | 89.7% |
| bird | 66.0% | 76.7% | - | 62.0% | 80.7% |
| dog | 39.7% | 99.0% | 69.3% | - | 68.0% |
| knife | 34.0% | 63.0% | 81.3% | 86.3% | - |

cheating suffix '`wild blers rwby migrant`' to `car`, `dog`, and `knife`, and show the generated results in Figure 7. Surprisingly, even though the original categories are not considered during the optimization process, we find out that the targeted manipulation still succeeded. Then, we systematically evaluate the universality of 20 cheating suffixes optimized for SD v14. We evaluate their effectiveness in targeted manipulation on the other three categories and present the BOTH score in Table 5. All cases exhibit a certain degree of universality, with the highest reaching up to 99%.

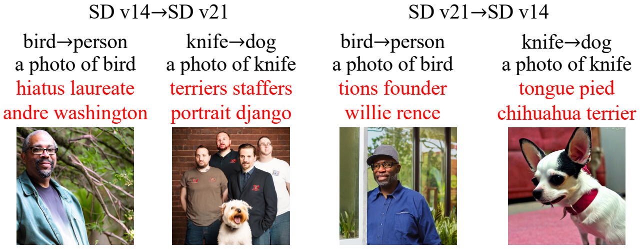


*Figure 8.* Examples of **black-box** transferability. 'SD v14 → SD v21' indicates manipulating SD v21 using the cheating suffix obtained for manipulating SD v14, and vice versa for 'SD v21 → SD v14'.

*Table 6.* **Black-box** targeted attack results. 'SD v14 → SD v21' indicates manipulating SD v21 using the cheating suffix obtained for manipulating SD v14, and vice versa for 'SD v21 → SD v14'. The metrics are defined in Section 5.1.

| Setting | CLIP | BLIP | OCNDR | TCDR | BOTH |
|---|---|---|---|---|---|
| SD v14 → SD v21 | 0.243 | 0.231 | 72.3% | 62.2% | 50.4% |
| SD v21 → SD v14 | 0.247 | 0.235 | 71.3% | 74.9% | 66.8% |

Next, we will demonstrate that our cheating suffixes exhibit transferability, meaning that cheating suffixes crafted to manipulate one diffusion model can also be effective against another diffusion model. This phenomenon has given rise to transfer-based **black-box** manipulation. Below, we use cheating suffixes generated from SD v14 to manipulate SD v21, and vice versa, use cheating suffixes generated from SD v21 to manipulate SD v14. The experimental results are listed in Table 6, where BOTH scores of 50.4% and 66.8% are achieved for SD v14 and SD v21, respectively. By comparing with Table 2, it can be observed that the performance degrades in the black-box manipulation scenario but still outperforms all the baselines. Some transfer-based results

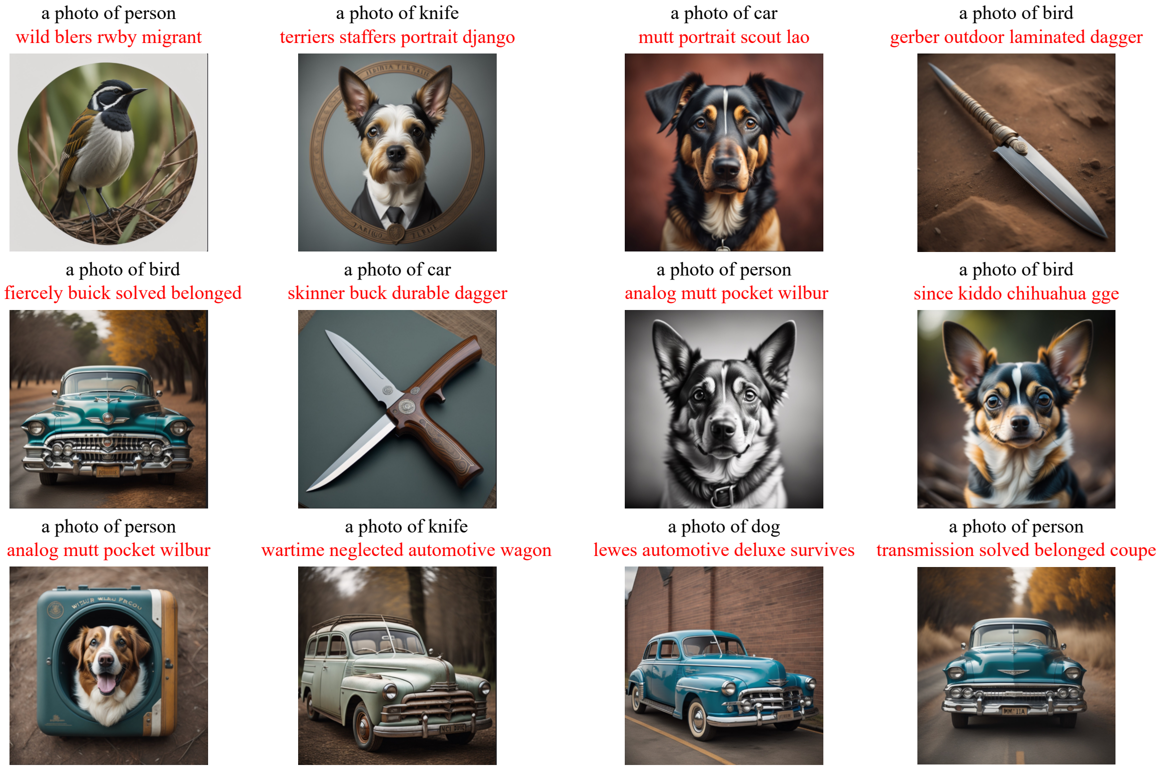

a photo of person
wild blers rwby migrant
a photo of knife
terriers staffers portrait django
a photo of car
mutt portrait scout lao
a photo of bird
gerber outdoor laminated dagger
a photo of bird
fiercely buick solved belonged
a photo of car
skinner buck durable dagger
a photo of person
analog mutt pocket wilbur
a photo of bird
since kiddo chihuahua gge
a photo of person
analog mutt pocket wilbur
a photo of knife
wartime neglected automotive wagon
a photo of dog
lewes automotive deluxe survives
a photo of person
transmission solved belonged coupe

*Figure 9.* Examples of **black-box** targeted attacks on Imagine Art. All the cheating suffixes are generated from SD v14.

are presented in in Figure 8. To the best of our knowledge, prior studies on targeted manipulation against T2I models have never addressed transferability.

Additionally, we conducted experiments of black-box targeted attacks on a commercial T2I online service, Imagine Art. Some of the results are shown in Figure 9.

Moreover, we also validated the transferability on the commercial model DALL-E 3, which is a popular T2I online service that can be accessed through ChatGPT 4[7]. Differing from other T2I models, DALL-E 3 automatically refines input prompts to be more user-friendly, mitigating the need for overly complicated prompt engineering. This step increases the difficulty of our transfer-based attacks. Two examples of successful black-box targeted attacks on DALL-E 3 are depicted in Figure 10.

## D. Ablation Study

In this part, we delve into the crucial aspect of ablation studies, focusing on two key elements: the initialization method and multi-modal objective functions.

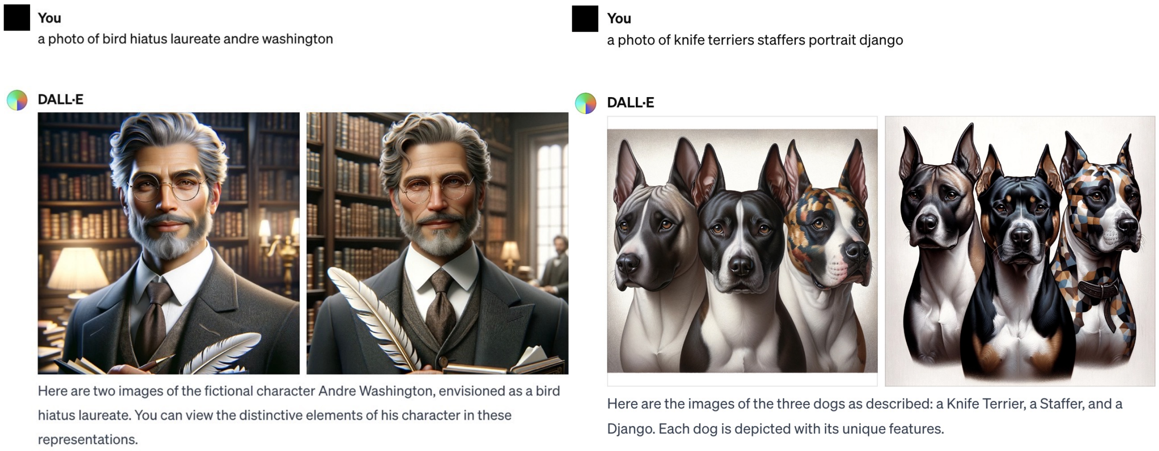
**Initialization Methods.** We investigate the impact of different initialization methods on manipulation performance. We conduct experiments on SD v14 and present the experimental results in Table 7, which shows that the EOS initialization performs the worst. This is because the `[eos]` token is not included in the filtered vocabulary, causing the Proj function to project it onto a distant word at the beginning. This phenomenon will impair the STE technique. In contrast, the 'Random' and 'Synonym' initialization allow the projection function to degenerate into an identity function at the initial value, enabling STE to provide a sufficiently accurate gradient at the beginning of optimization. Furthermore, the 'Synonym' initialization offers a more intuitively better initial solution compared to 'Random'. Thus, it leads to better results and serves as our default choice.

*Table 7.* Results of different initialization methods on SD v14. The metrics are defined in Section 5.1. **Best results are boldfaced.**

| Initialization | CLIP | BLIP | OCNDR | TCDR | BOTH |
|---|---|---|---|---|---|
| EOS | 0.262 | 0.390 | 82.2% | 78.3% | 72.3% |
| Random | 0.263 | 0.400 | 84.1% | 82.0% | 74.4% |
| Synonym | **0.265** | **0.414** | **92.0%** | **87.2%** | **81.8%** |

**Multi-modal Objectives.** We further investigate the impact of the weighting factor $\lambda$ on the manipulation performance, where $\lambda$ represent the importance of the text modal loss term. We enumerate different values of $\lambda$ from {0, 0.001,

[7] https://chat.openai.com/g/g-2fkFE8rbu-dall-e



You

a photo of bird hiatus laureate andre washington

DALL·E

Here are two images of the fictional character Andre Washington, envisioned as a bird hiatus laureate. You can view the distinctive elements of his character in these representations.

You

a photo of knife terriers staffers portrait django

DALL·E

Here are the images of the three dogs as described: a Knife Terrier, a Staffer, and a Django. Each dog is depicted with its unique features.

*Figure 10.* Examples of **black-box** targeted attacks for the commercial T2I model DALL-E 3. The cheating suffixes are generated by SD v14. (Left) The original category and target category are `person` and `bird`, respectively. (Right) The original category and target category are `knife` and `dog`, respectively.

$0.01, 0.1, 0.25, 0.5, 0.75, 1\}$ and plotted the manipulation results on SD v14 in Figure 11. When $\lambda = 0$, it implies a method using only the **I**mage-**M**odal **P**rior (we call it **IMP-Attack**), corresponding to the dashed line. Figure 11 shows that when $\lambda$ is small, the manipulation performance is similar to IMP-Attack, and it increases as $\lambda$ increases. However, when $\lambda$ exceeds 0.1, the manipulation performance starts to decrease rapidly. This phenomenon indicates that the image modality plays a more prominent role in MMP-Attack. Furthermore, the alignment between these two modalities is not consistently optimal due to their inherent conflicting performance characteristics during attack. Therefore, incorporating both text and image features into an attack can be advantageous.

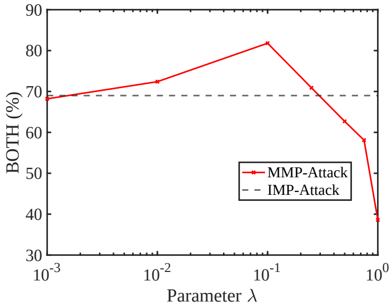


*Figure 11.* The BOTH scores versus $\lambda$. The dashed line indicates an IMP-Attack, using only the image modal prior($\lambda = 0$).