This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Interpretable Prompts made Edit-Friendly: Token-to-Token Similarity Reduction in dLLMs for Edit-Friendly Hard Prompt Inversion

Naresh Kumar Devulapally[1†] Shruti Agarwal[2] Vishal Asnani[2] Vishnu Suresh Lokhande[1†]
[1]University at Buffalo, SUNY [2]Adobe Research
{devulapa, vishnulo}@buffalo.edu {shragarw,vasnani}@adobe.com

Abstract

Crafting prompts via Prompt Engineering that steer a model's internal representations toward specific and predefined outcomes can be time-consuming, often requiring multiple iterations. Hard Prompt Inversion offers a complementary workflow: start from a reference image and generate a prompt that conditions a text-to-image (T2I) model to reconstruct the reference image. Existing inversion methods either yield incoherent text, or produce prompts that are overly sensitive to downstream token edits. We propose a dLLM-based prompt inversion framework that yield prompts that are (i) more interpretable to humans, (ii) better aligned with the reference image, and (iii) designed for downstream token swap and token append operations (aka edit-friendly prompts). The method is plug-and-play, requiring no finetuning of either the T2I model or the dLLM. Experiments across three datasets show a $\sim 10\times$ reduction in inversion time relative to existing prompt-inversion baselines, higher interpretability scores, and significantly higher prompt editability, as measured by TIFA, GPT-V preference scoring, and controlled user studies, all while preserving high-fidelity image generation. By coupling diffusion-time sampling with token-similarity control inside a dLLM decoder, our approach extends prompt inversion beyond reconstruction to downstream token-editing tasks, enabling faster, more transferable prompts that generalize across multiple T2I models.

1. Introduction

Text-to-image (T2I) generative models [9] have advanced rapidly in recent years, enabling users to create diverse and high-quality images from natural-language descriptions. Modern diffusion-based systems such as DALL-E 2 [27], Imagen [30], and Stable Diffusion [28], built on denoising diffusion probabilistic models [9], can synthesize photorealistic images with fine-grained control over style and content. Despite this progress, crafting an effective text prompt remains non-trivial: users often rely on ad-hoc "prompt engineering" with extensive trial-and-error, repeatedly querying the model until the output roughly matches their intent. This is costly both for users, who must iteratively probe the model to translate their creative vision into images, and for systems, which must repeatedly perform expensive inference, increasing latency, compute, and energy consumption.

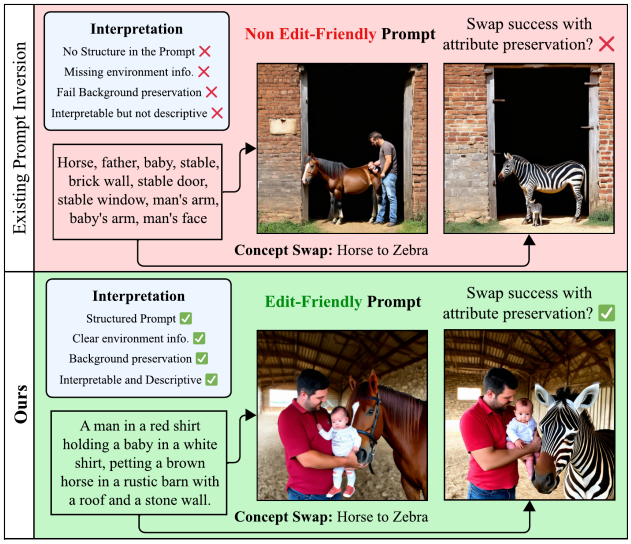

Figure 1. ***What makes a prompt edit-friendly?*** Compared to existing hard prompt inversion methods, our approach produces structured, interpretable prompts that align more accurately with image content. This enables successful prompt-level edits, like swapping "horse" with "zebra" (in the above figure), while maintaining background consistency and fine-grained attributes like clothing, pose, and environment.

Prompt inversion offers a principled way to automate prompt design: given a reference image I, the goal is to find a textual prompt T that makes a text-to-image (T2I) model reproduce I's content and style. Existing methods fall into *soft* and *hard* inversion. Soft approaches such as Textual Inversion [5] and DreamBooth [29] learn continuous embeddings in the diffusion model's text-encoder space; while expressive, these representations are opaque and non-editable, often overfitting fine appearance details and limiting con-

†Corresponding authors: N. K. Devulapally and V. S. Lokhande.

trollability. Hard inversion instead seeks a discrete token sequence $T=[x_1, \ldots, x_N]$ that directly reproduces the reference image. Gradient-based methods optimize tokens or embeddings using CLIP [26] similarity but frequently yield awkward or near-nonsense strings, and more recent gradient-free or autoregressive variants improve fluency yet remain brittle: small token edits can drastically distort the generated image, undermining interactive or compositional use. Evaluation has similarly focused on reconstruction under CLIP-based or captioning metrics, largely ignoring robustness to token-level edits, despite newer tools like TIFA [11] and GPT-4V-based assessments [23] that can directly probe edit behavior.

We revisit hard prompt inversion from the perspective of *edit-friendly prompt design*. Building on discrete diffusion language models [2, 16] and CLIP guidance [26], we construct an inversion procedure that explicitly encourages (i) high alignment between the inverted prompt and the reference image, and (ii) consistent alignment under textual edits such as token swaps and appends. Our method produces human-interpretable prompts that can be meaningfully modified without collapsing the underlying image semantics, narrowing the gap between automatic inversion and how users actually utilize inverted prompts in practice.

Our main contributions are:

- **Edit-friendly prompt inversion.** A hard inversion objective that directly optimizes alignment under token-swap and token-append edits, evaluated on three datasets with TIFA, GPT-4V-based metrics, and human studies.
- **CLIP-guided discrete diffusion.** A CLIP-guided discrete diffusion language model that improves text-image faithfulness while maintaining coherent prompts.
- **Efficient, interpretable prompts.** Our method achieves $\sim 10\times$ faster inversion than strong hard-prompt baselines, without sacrificing interpretability or alignment to the reference image.

2. Related Work

Prompt inversion for text-to-image diffusion has been studied both in continuous and discrete spaces. Soft personalization methods such as Textual Inversion and DreamBooth invert reference images into learned embeddings or fine-tuned weights of a diffusion model, achieving highly faithful reconstructions but producing non-interpretable representations that users cannot easily edit at the token level [6, 29]. More recent hard prompt methods instead optimize directly over discrete tokens, such as gradient-based discrete search for CLIP prompts [34] and prompt inversion for Stable Diffusion via delayed projection onto the vocabulary, which yields readable prompts aligned with image semantics [22]. Follow-up work including EDITOR combines captioning, embedding optimization, and projection back to text to obtain more fluent inversion prompts [15], but all these methods primarily target reconstruction of a fixed image, rather than utilization of the inverted prompt under downstream edits.

Text-image alignment has traditionally been evaluated with CLIP-based similarity measures, but these metrics struggle with fine-grained semantics. TIFA [11] addresses this by generating question-answer pairs from the text and checking, via VQA models, whether they hold in the generated image, providing an automatic, interpretable faithfulness score that correlates well with human judgments [11]. ImageReward instead learns a reward model from large-scale human preference comparisons to rank text-image pairs and guide model optimization toward human-desired outputs [39]. In parallel, multimodal LMs such as GPT-4V(ision) have emerged as powerful "LLM-as-a-judge" evaluators: early studies show GPT-4V can perform nuanced visual reasoning and caption assessment [37], and the GPT-4 system report characterizes it as a general-purpose multimodal model [1]. Building on these advances, we use TIFA and GPT-4V-based scoring to assess not only reconstruction fidelity but also how well alignment is preserved under systematic token-level edits to the inverted prompt.

3. Preliminaries

Given a reference image I, *hard prompt inversion* seeks a text prompt $T = \left[x_1^{\text{txt}}, x_2^{\text{txt}}, \ldots, x_N^{\text{txt}}\right]$ such that conditioning a text-to-image Latent Diffusion Model (LDM) on T makes it reproduce I (or images that are visually similar to I). Let $p(I \mid T)$ denote the LDM likelihood of I given T. The ideal inversion prompt T_* maximizes this likelihood:

$$T_* = \arg\max_T p(I \mid T) = \arg\max_T \frac{p(I)\, p(T \mid I)}{p(T)} \propto \arg\max_T \frac{p(T \mid I)}{p(T)} \tag{1}$$

that is, hard prompt inversion implicitly trades off an image-conditioned prompt posterior $p(T \mid I)$ against a prompt prior $p(T)$. Hard prompt inversion methods can be broadly categorized into *gradient-based* and *gradient-free* approaches.

Hard prompt inversion is complementary to modern image-editing techniques. Image editing maps a source image to an edited image ($I_i \rightarrow I_e$), whereas hard prompt inversion maps the source image to an interpretable, reusable prompt ($I_i \rightarrow T_*$) that supports text-only downstream generation and localized token operations such as swap and append. This distinction matters because prompt-centric workflows must begin from text rather than carry the source image at inference time, and inverted prompts can also serve as reusable handles for downstream personalization (e.g., by appending learned concept tokens [6]).

3.1. Gradient-Based Inversion

Gradient-based methods directly optimize a continuous representation of the prompt with respect to an image-similarity objective, and then map it back to discrete tokens. For example, PEZ [35] performs gradient descent in embedding

space with projection to the vocabulary, while PH2P [21] incorporates diffusion-aware schedules into the optimization. Although effective at matching the reference image, these approaches often produce brittle or non-fluent token sequences that are difficult to interpret or edit.

3.2. Gradient-Free Inversion

Gradient-free hard prompt inversion instead searches directly over discrete token sequences using external scores, without backpropagating through the T2I model. Recent work such as VGD [13] proposes a *gradient-free* decoding scheme that balances *visual alignment* and *linguistic fluency*. A typical objective is

$$\hat{T} \;=\; \arg\max_{T}\; p_{\text{LLM}}(T)\; p_{\text{CLIP}}(I \,|\, T) \tag{2}$$

where $p_{\text{CLIP}}(I \,|\, T)$ is a CLIP-based surrogate for image–text alignment [26] and $p_{\text{LLM}}(T)$ is a language-model score. Decoding proceeds token-by-token (beam search) over the vocabulary.

While such methods usually produce *interpretable* prompts, they provide little control over *downstream* text-to-image generation: small edits to $\hat{T}$ (such as swapping or appending a style or attribute token) can induce large, unpredictable changes in the generated image. In other words, current gradient-free hard inversion yields prompts that are readable but not reliably *edit-friendly*, limiting their usefulness as building blocks for user-driven prompt design.

4. Method

4.1. Overview

Given a reference image I, we seek an inverted prompt T_* that (i) is well aligned with I, (ii) is fluent and human-readable, and (iii) yields predictable image changes under simple token edits (swap/append of style or attributes). As illustrated in Fig. 1, an edit-friendly prompt acts as a stable handle for downstream text-to-image editing rather than a brittle one-shot description.

To encode this goal, we augment the gradient-free objective with an explicit editability term:

$$T_* \;=\; \arg\max_{T}\; p_{\text{LLM}}(T)\; p_{\text{CLIP}}(I \,|\, T)\; p_{\text{edit}}(T) \tag{3}$$

where $p_{\text{CLIP}}(I \,|\, T)$ measures image-text alignment, $p_{\text{edit}}(T)$ measures how well T behaves under token-level edits. Fluency is enforced implicitly by the language prior used during decoding by $p_{\text{LLM}}(T)$ term. Section 4.4 describes p_{edit} in detail; below we outline how we instantiate the prior and the CLIP-alignment term.

4.2. Discrete Diffusion Language Model Prior

To avoid slow token-by-token beam search, we replace the Autoregressive (AR) LLM used in prior work with a *discrete diffusion language model* (dLLM) that generates text via iterative parallel refinement [2, 3, 17, 31]. Starting from a fully noised sequence x_T, the dLLM defines a reverse Markov chain

$$p_{\text{dLLM}}(x_T, \ldots, x_0) \;=\; \pi_{\text{prior}}(x_T) \prod_{t=T-1}^{0} p(x_t \mid x_{t+1}) \tag{4}$$

where π_{prior} is a dLLM prior and $p(x_t \mid x_{t+1})$ are learned denoising transitions in token space. Each step updates all positions in parallel, and we decode x_t to text as $\tilde{T}_t = \text{decode}(x_t)$.

Compared to AR decoding, which grows one token at a time and repeatedly scores partial prefixes, dLLM decoding refines full-length sequences in $T \ll N$ steps and naturally enforces global coherence. In our setting, the dLLM serves as the fluency prior that proposes globally consistent candidate prompts; the CLIP and editability terms in Eq. (3) then bias these candidates toward better alignment and edit behavior.

4.3. CLIP-Guided Steering

We incorporate CLIP guidance directly into the dLLM sampler to realize the alignment term $p_{\text{CLIP}}(I \,|\, T)$. Let $r_{\text{CLIP}}(T; I) \;=\; \cos\big(f_I(I),\, f_T(T)\big)$ denote the CLIP similarity between image I and text T, with fixed encoders f_I and f_T [26]. We adopt Feynman-Kac (FK) steering [32] to reweight each reverse step of the diffusion chain.

Given decoded text $\tilde{T}_t = \text{decode}(x_t)$ at step t, we define a difference potential [32]:

$$G_t(x_{T:t}) \;=\; \exp\Big(\lambda_{\text{CLIP}}\, \big[r_{\text{CLIP}}(\tilde{T}_t; I) - r_{\text{CLIP}}(\tilde{T}_{t+1}; I)\big]\Big) \tag{5}$$

with tilt parameter $\lambda_{\text{CLIP}} \geq 0$. The telescoping form ensures $\prod_{t=0}^{T-1} G_t \;=\; \exp\big(\lambda_{\text{CLIP}}\, r_{\text{CLIP}}(\tilde{T}_0; I)\big)$, so trajectories that end with higher CLIP similarity receive higher overall weight. In practice, each denoising step draws candidate sequences from $p(x_t \mid x_{t+1})$, decodes them to text, evaluates r_{CLIP}, and resamples candidates according to G_t. This preserves the efficiency of dLLM decoding while steering the sampler toward prompts that are strongly aligned with the reference image.

4.4. Edit-Friendly Inversion

The term $p_{\text{edit}}(T)$ in Eq. 3 is a surrogate *editability score* for a prompt T, rewarding prompts whose token-level edits induce controlled, localized image changes (e.g., style or attribute changes) without collateral drift in content.

A natural but costly strategy is to keep a full text-to-image (T2I) diffusion model in the inversion loop and optimize cross-attention maps, as done in prior work on word-region alignment and style editing [4, 8, 18, 40]. However, such methods, including PH2P [21] and PRISM [7], require repeated model queries, rely on high-dimensional attention tensors, and are tightly coupled to specific architectures and tokenizers, resulting in high memory and latency costs.

Furthermore, attention is an imperfect proxy for token importance [12, 36], and token interactions in T2I models are complex and architecture-dependent [20, 33]. To address these limitations, we adopt a lightweight, model-agnostic editability signal derived from the dLLM's predictive distributions, which integrates directly into our steering objective.

Token-Token similarity edit reward. To quantify editability at the level of individual tokens, we encourage prompts whose tokens are *disentangled* and *modular*, so that small token edits correspond to localized semantic changes. At refinement step t, let $\mathbf{z}_i^{(t)} \in \mathbb{R}^V$ be the logits over a vocabulary of size V at position $i \in \{1, \dots, N\}$, and

$$\tilde{\mathbf{z}}_i^{(t)} = \mathbf{z}_i^{(t)} - \tfrac{1}{V}\sum_{v=1}^{V} z_i^{(t)}(v), \quad \hat{\mathbf{p}}_i^{(t)} = \mathrm{softmax}\big(\tilde{\mathbf{z}}_i^{(t)}\big) \tag{6}$$

be mean-centered logits and the corresponding token distribution. Stacking $\hat{\mathbf{p}}_i^{(t)}$ as rows yields $\hat{\mathbf{P}}^{(t)} \in \mathbb{R}^{N\times V}$ and the row Gram matrix

$$\mathbf{S}^{(t)} \;=\; \hat{\mathbf{P}}^{(t)}\big(\hat{\mathbf{P}}^{(t)}\big)^{\top}, \qquad S_{ij}^{(t)} = \big\langle \hat{\mathbf{p}}_i^{(t)}, \hat{\mathbf{p}}_j^{(t)} \big\rangle \tag{7}$$

Large off-diagonal entries $S_{ij}^{(t)}$ indicate that positions i and j are predicted with similar token distributions, suggesting coupled roles and poor editability.

Let $\mathrm{Off}\big(\mathbf{S}^{(t)}\big) = \mathbf{S}^{(t)} - \mathbf{I}_N$ denote the matrix of off-diagonal token–token similarities. We summarize off-diagonal coupling using the mean and variance of these entries:

$$\mu_{\text{off}}^{(t)} = \mathbb{E}_{i\neq j}\big[\mathrm{Off}\big(\mathbf{S}^{(t)}\big)_{ij}\big], \;\; \sigma_{\text{off}}^{2(t)} = \mathrm{Var}_{i\neq j}\big[\mathrm{Off}\big(\mathbf{S}^{(t)}\big)_{ij}\big] \tag{8}$$

High $\mu_{\text{off}}^{(t)}$ indicates globally entangled tokens; high $\sigma_{\text{off}}^{2(t)}$ indicates a few strongly coupled outliers. We define a bounded stepwise edit reward

$$r_{\text{edit}}^{(t)} \;=\; 1 \;-\; \big(\mu_{\text{off}}^{(t)} + \sigma_{\text{off}}^{(t)}\big) \tag{9}$$

which increases only when both average coupling and its variability decrease. At termination we use $r_{\text{edit}}(x_0) \triangleq r_{\text{edit}}^{(0)}$ as the editability score of the final prompt.

Edit reward as a difference potential. We incorporate r_{edit} into decoding via a telescoping potential [32] with strength λ_{edit}:

$$G_t^{\text{edit}} \;=\; \exp\Big(\lambda_{\text{edit}}\,\big[r_{\text{edit}}^{(t)} - r_{\text{edit}}^{(t+1)}\big]\Big), \quad G_T^{\text{edit}} = 1 \tag{10}$$

so that $\prod_{t=0}^{T} G_t^{\text{edit}} = \exp\big(\lambda_{\text{edit}}\, r_{\text{edit}}^{(0)}\big)$ and the terminal distribution is tilted toward low-coupling, edit-friendly prompts.

During decoding we apply both CLIP and edit potentials. With CLIP reward $r_{\text{CLIP}}(\tilde{T}_t; I)$ as in Sec. 4, the combined potential at step t is

$$\begin{aligned} G_t = \exp\Big(&\lambda_{\text{CLIP}}\,[\,r_{\text{CLIP}}(\tilde{T}_t; I) - r_{\text{CLIP}}(\tilde{T}_{t+1}; I)\,] \\ &+ \lambda_{\text{edit}}\,[\,r_{\text{edit}}^{(t)} - r_{\text{edit}}^{(t+1)}\,]\Big) \end{aligned} \tag{11}$$

which induces the steered final distribution

$$\begin{aligned} p_{\text{final}}(x_0 \mid I) \propto p_{\text{dLLM}}(x_0 \mid I)\, \exp\Big(&\lambda_{\text{CLIP}}\, r_{\text{CLIP}}(x_0; I) \\ &+ \lambda_{\text{edit}}\, r_{\text{edit}}(x_0)\Big) \end{aligned} \tag{12}$$

matching the objective in Eq. 3 with the dLLM prior providing linguistic fluency and the CLIP and edit rewards enforcing alignment and edit-friendliness.

Algorithm 1 Steering with CLIP & Edit Potentials

Require: Diffusion model $p(x_{0:T} \mid c)$; proposal $\tau(x_t \mid x_{t+1}, c)$; steps T; particles K; strengths $(\lambda_{\text{CLIP}}, \lambda_{\text{edit}})$; CLIP encoders f_I, f_T

Returns: set of decoded prompts $\{\mathrm{decode}(x_0^i)\}_{i=1}^K$ and the selected prompt T_*

Sample: $x_T^i \sim \tau(x_T \mid c)$ for $i \in [K]$

Score (init): $G_T^i \leftarrow 1$ for $i \in [K]$

for $t \in \{T, \dots, 1\}$ **do**

  Resample: draw ancestors $a_t^i \sim \mathrm{Multinomial}\big(\{G_t^j\}_{j=1}^K\big)$ and set $x_t^i \leftarrow x_t^{a_t^i}$

  Propose: $x_{t-1}^i \sim \tau(x_{t-1} \mid x_t^i, c)$ (one reverse step)

  Compute rewards:

    CLIP: $\tilde{T}_t^i \leftarrow \mathrm{decode}(x_t^i)$, $\tilde{T}_{t-1}^i \leftarrow \mathrm{decode}(x_{t-1}^i)$

      $r_{\text{CLIP}}^{(t)}(x_t^i) \leftarrow \cos\big(f_I(I), f_T(\tilde{T}_t^i)\big)$

    Edit: form $\hat{\mathbf{P}}_i^{(t)}$ and $\hat{\mathbf{P}}_i^{(t-1)}$; Eq. (6)

      $\mathbf{S}_i^{(t)} \leftarrow \hat{\mathbf{P}}_i^{(t)}(\hat{\mathbf{P}}_i^{(t)})^{\top}$, $\mathbf{S}_i^{(t-1)} \leftarrow \hat{\mathbf{P}}_i^{(t-1)}(\hat{\mathbf{P}}_i^{(t-1)})^{\top}$

      Compute $r_{\text{edit}}^{(t)}(x_t^i)$ and $r_{\text{edit}}^{(t-1)}(x_{t-1}^i)$ Eq. (9)

      Compute G_{t-1} via Eq. (11)

  Re-weight: for each i,

$$G_{t-1}^i \;=\; \frac{p\big(x_{t-1}^i \mid x_t^i, c\big)}{\tau\big(x_{t-1}^i \mid x_t^i, c\big)}\, G_{t-1}\big(x_T^i, \dots, x_{t-1}^i, c\big)$$

end for

Output: return samples $\{x_0^i\}$

5. Experiments

5.1. Datasets

We conduct our experiments on three public datasets. **MS COCO** [19]: natural photographs with multi-object scenes and human-written captions; **Flickr8K** [10]: everyday scenes with single-sentence captions; **JourneyDB** [24]: a collection of text-image pairs curated from community T2I generations (diverse, stylized, long prompts).[†] For each dataset, following [13], we uniformly sample 200 images and report means across 5 runs.

5.2. Evaluation Metrics

We evaluate our gradient-free, interpretable, and edit-friendly prompt inversion along three axes: (i) **prompt text quality**,

[†]We use a filtered subset of JourneyDB containing unique images and de-duplicated prompts; details in the Supplement.

Table 1. Text and Source-Image Reconstruction evaluation compared to baselines. Text Metrics are grouped into *Prompt Accuracy* (Precision/Recall/F1; ↑), *Interpretability* (Perplexity, PPL; ↓), and *Alignment* (CLIP Text Sim., CL-T; ↑ and CLIP Image Sim., CL-I; ↑). ***We observe that our method generates Text Prompts with improved Prompt Accuracy, Interpretability, and Alignment scores.***

Method	MS COCO						Flickr8k						JourneyDB					
	Prompt Acc.			Interpret.	Alignment		Prompt Acc.			Interpret.	Alignment		Prompt Acc.			Interpret.	Alignment	
	Pre.↑	Rec.↑	F1↑	PPL↓	CL-T↑	CL-I↑	Pre.↑	Rec.↑	F1↑	PPL↓	CL-T↑	CL-I↑	Pre.↑	Rec.↑	F1↑	PPL↓	CL-T↑	CL-I↑
16 Tokens																		
Captioning [38]	0.82	0.86	0.84	59.65	0.47	0.46	0.83	0.84	0.83	**60.80**	0.40	0.48	0.82	0.84	0.83	**59.92**	0.47	0.46
BLIP-2 [14]	0.89	0.92	0.91	32.49	0.55	0.48	0.89	0.90	0.90	105.39	0.53	0.49	0.87	0.85	0.86	101.39	0.45	0.48
CLIP Int. 2.1 [25]	0.88	0.90	0.89	164.93	0.60	0.51	0.88	0.88	0.88	142.45	0.57	0.47	0.86	0.84	0.85	162.57	0.40	0.51
PEZ [35]	0.77	0.83	0.80	7411.62	0.25	0.68	0.76	0.80	0.78	6770.91	0.15	0.62	0.77	0.81	0.79	6275.60	0.29	0.65
VGD [13]	0.88	0.90	0.89	100.17	0.60	0.67	0.89	0.89	0.89	93.11	0.62	0.49	0.86	0.85	0.86	109.45	0.45	0.57
Ours	**0.90**	**0.92**	**0.91**	**54.68**	**0.64**	**0.71**	**0.90**	**0.90**	**0.90**	67.53	**0.64**	**0.65**	**0.88**	**0.86**	**0.87**	99.52	**0.49**	**0.66**
32 Tokens																		
Captioning [38]	0.81	0.86	0.84	36.89	0.49	0.48	0.82	0.84	0.83	37.51	0.44	0.49	0.81	0.85	0.83	**34.88**	0.41	0.45
BLIP-2 [14]	0.87	0.91	0.89	40.72	0.52	0.48	0.85	0.90	0.87	47.27	0.58	0.50	0.83	0.85	0.84	45.86	0.43	0.49
CLIP Int. 2.1 [25]	0.83	0.89	0.86	142.02	0.55	0.52	0.83	0.88	0.86	123.94	0.54	0.49	0.83	0.84	0.83	163.07	0.45	0.52
PEZ [35]	0.74	0.83	0.78	4946.56	0.21	0.69	0.75	0.80	0.77	4686.81	0.13	0.61	0.75	0.81	0.78	4250.46	0.27	0.67
VGD [13]	0.87	0.91	0.89	37.78	0.56	0.68	0.87	0.90	0.88	37.54	0.59	0.51	0.87	0.87	0.87	47.25	0.48	0.58
Ours	**0.87**	**0.91**	**0.89**	**33.16**	**0.59**	**0.71**	**0.87**	**0.90**	**0.88**	**32.44**	**0.61**	**0.64**	**0.86**	**0.85**	**0.86**	37.94	**0.48**	**0.66**
64 Tokens																		
Captioning [38]	0.80	0.87	0.83	24.22	0.49	0.47	0.82	0.85	0.83	25.16	0.45	0.52	0.80	0.86	0.83	23.86	0.42	0.47
CLIP Int. 2.1 [25]	0.79	0.88	0.83	112.62	0.51	0.51	0.80	0.87	0.83	94.10	0.51	0.52	0.80	0.84	0.82	113.10	0.47	0.50
PEZ [35]	0.72	0.82	0.76	2501.24	0.17	0.67	0.72	0.80	0.76	2366.88	0.10	0.58	0.56	0.72	0.80	2403.29	0.24	0.67
VGD [13]	0.84	0.90	0.87	25.10	0.52	0.67	0.84	0.89	0.86	23.64	0.56	0.53	0.85	0.87	0.86	28.70	0.49	0.58
Ours	**0.84**	**0.90**	**0.87**	**19.08**	**0.57**	**0.73**	**0.84**	**0.89**	**0.86**	**19.05**	**0.59**	**0.66**	**0.85**	**0.86**	**0.85**	**20.37**	**0.49**	**0.67**
∼ 77 Tokens																		
Captioning [38]	0.80	0.87	0.83	22.08	0.49	0.49	0.81	0.85	0.83	22.96	0.45	0.51	0.80	0.86	0.82	22.06	0.42	0.50
CLIP Int. 2.1 [25]	0.78	0.88	0.83	108.15	0.50	0.53	0.79	0.87	0.83	92.14	0.50	0.50	0.79	0.84	0.81	104.93	0.47	0.51
PEZ [35]	0.70	0.82	0.76	1879.27	0.15	0.66	0.52	0.71	0.79	1811.17	0.06	0.58	0.71	0.80	0.75	1905.68	0.25	0.68
VGD [13]	0.83	0.90	0.86	21.79	0.51	0.68	0.83	0.89	0.86	21.39	0.55	0.54	0.85	0.87	0.86	24.23	0.49	0.59
Ours	**0.84**	**0.90**	**0.87**	**17.10**	**0.56**	**0.74**	**0.84**	**0.89**	**0.86**	**17.42**	**0.59**	**0.67**	**0.84**	**0.86**	**0.85**	**18.47**	**0.49**	**0.68**

(ii) **reference image reconstruction**, and (iii) **prompt editability**. Our goal is to produce prompts whose token-level edits, including concept swap and concept append, induce localized and consistent image changes while preserving unrelated content. We assess how well an inverted prompt aligns with the source caption using BERTScore (precision, recall, and F1) and CLIP-Text similarity. For fair comparison with prior work [13], we use separate CLIP models for guidance and evaluation: CLIP-ViT-H-14 for steering and CLIP-ViT-G-14 for similarity measurement. We measure prompt interpretability using GPT-2 perplexity (PPL). We also report complexity and wall-clock inversion time in Sec. 6.3, where our method achieves $\sim 10\times$ efficiency gains over hard-prompt baselines. We evaluate reconstruction fidelity using CLIP Image similarity between each reference image and its reconstruction, and additionally report controlled user studies (see Supplement), following the protocol of [13]. Our central objective is to generate prompts that support reliable token-level edits with corresponding image-level effects. We evaluate text-to-image generation quality after concept swap and concept append operations using TIFA [11] and GPT-4V-based scores [41], and further validate these results through controlled user studies (Fig. 4). In **token swap**, a subject or concept token x_i is replaced with $\tilde{x}$, for example, "A **cat** in a park" $\rightarrow$ "A **dog** in a park." In **token append**, attribute or style tokens are added to the prompt.

5.3. Baselines

We compare our method against hard prompt inversion methods, as well as captioning-based baselines: VGD [13], PEZ [35], BLIP-2 [14], CLIP-Interrogator [25], and Florence-2 [38]. We report results across prompt inversion token budgets of $16, 32, 64$, and ~ 77 tokens.

6. Results

6.1. Prompt Inversion and Image Reconstruction

Tab. 1 depicts the performance of our method for Prompt Inversion compared to baselines. We observe that our method that incorporates CLIP-guidance token-similarity regularization outperforms existing soft and hard prompt inversion

Figure 2. **Reference-Image reconstructions qualitative results compared to baselines:** Top row: Reference-image reconstruction generated from inverted prompts. Bottom row: Inverted Text Prompt. We observe that our method generates coherent, aligned, structured prompts that preserve all attributes. We list these attributes in the left-most image (Reference Image). We also provide the reference caption (not used for inversion) for comparison with the inverted prompt. Our quantitative results are provided in Tab. 1.

baselines on all Text and Reference Image Reconstruction evaluation metrics. Notably, we observe significant reduction in PPL Scores (improved interpretability) when compared to interpretable baselines including BLIP-2 [14] and VGD [13]. We observe that as the token length increases, we observe consistent reduction in Perplexity scores while the CLIP-Image similarity (CL-I) between generated image and reference image increases indicating that descriptive prompts generate images that are aligned with reference images. We also observe that the gains of our method at text-level and image-level evaluation metrics are consistent across datasets.

Summary: Our method generates Text Prompts with improved Prompt Accuracy, Interpretability, and Alignment.

6.2. Token Editability

We evaluate the effectiveness of inverted hard prompts on a downstream text to image generation task. Recall from Sec. 4.4 that our method aims to perform *Edit-Friendly Prompt Inversion*. We present our results in Fig. 4. We choose Captioning [38], VGD [13], and CLIP Interrogator 2.1 [25] for comparison as they generate interpretable prompts that allow token-level concept swap. For consistency, we choose prompts that have the same concept in the inverted text. From Fig. 4, we observe that our method results in modular, disentangled prompts that allow seamless token-level Concept Swap and Concept Append image generation. We also provide qualitative results compared to VGD (interpretable baseline with best clip-image similarity with generated and reference image) [13] for Concept Swap and Concept Append in Fig. 3. We clearly observe that our method introduces consistent and localized concept swaps and successful concept append results while preserving other image attributes such as background without introducing undesirable artifacts into the image.

Summary: Our method generates edit-friendly prompt with improved TIFA and GPT-V scores.

6.3. Prompt Inversion Efficiency

Let N denote prompt length, V vocabulary size, and B beam size. In autoregressive (AR) decoding, tokens are generated sequentially, so CLIP-guided search over full prompts scales as $\mathcal{O}\big(BN\,E_{\text{LM}} + BN\,C_{\text{align}}\big)$, where E_{LM} is the cost of one

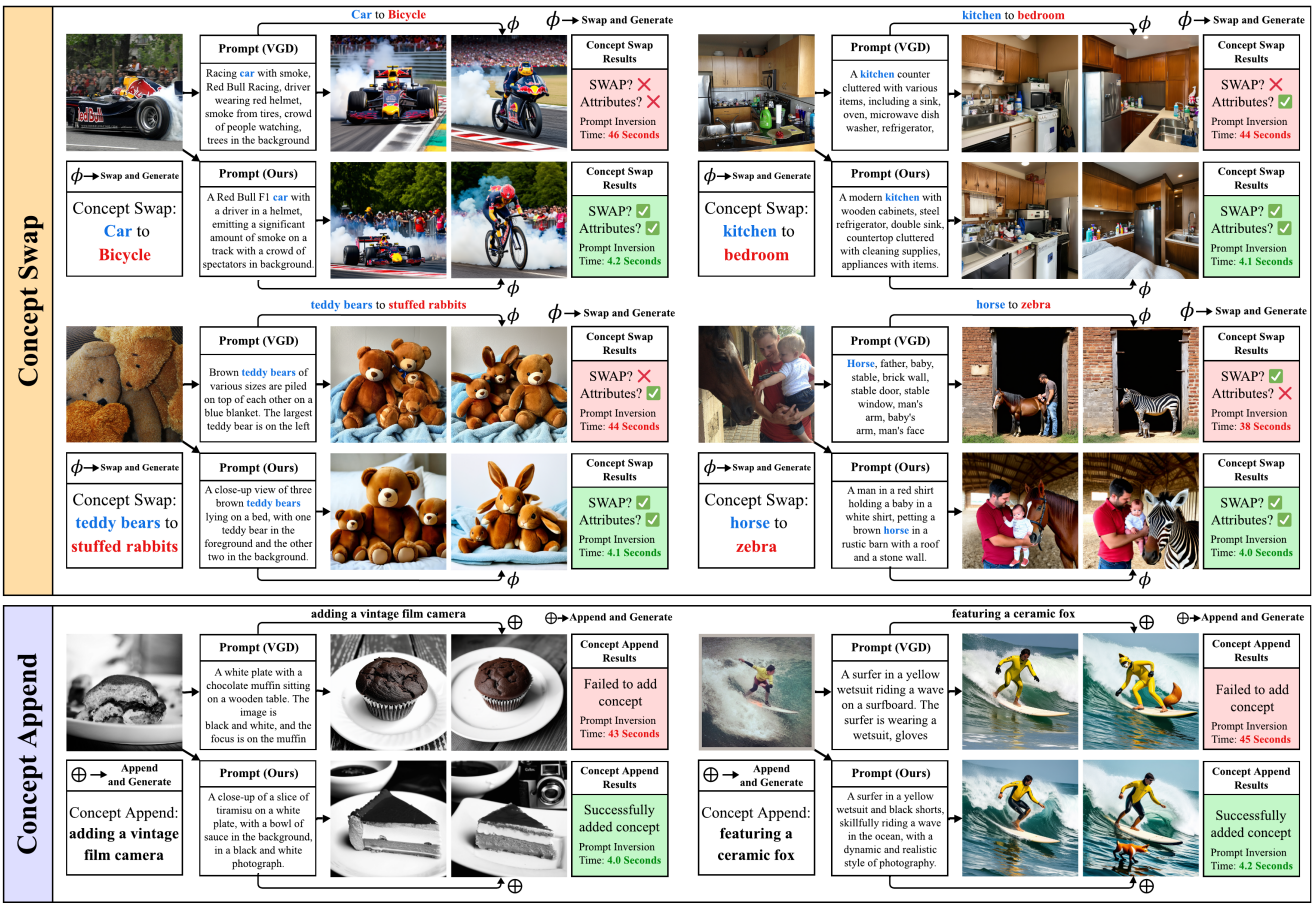

Figure 3. **Edit-Friendly evaluation of our method compared to interpretable baseline:** *Each example: Top-left image is inverted to prompt (VGD and Ours). This prompt then undergoes token-level Concept-Swap or Concept Append and the final image is generated.* Our prompts introduce localized, consistent image level changes through token-level concept swap and concept append operation. In the qualitative samples above, we see that our method consistently preserves background and attributes while ensuring swap and append success.

Table 2. Prompt Inversion time and Image Reconstruction Similarity across token budgets. Metrics per token: Time (s; ↓) and CLIP-Image Similarity (CL-I (↑)). We observe that when compared to best performing prompt inversion techniques such as PEZ [35] and VGD [13], our method demonstrates significant efficiency improvements. All experiments are run on a single A6000 GPU.

Method	**16 tokens**		**32 tokens**		**64 tokens**		**77 tokens**	
	Time↓	CL-I↑	Time↓	CL-I↑	Time↓	CL-I↑	Time↓	CL-I↑
Captioning [38]	2.80	0.46	4.20	0.48	8.20	0.47	9.30	0.49
BLIP-2 [14]	0.80	0.48	1.40	0.48	3.60	0.51	4.80	0.53
CLIP Int. [25]	**0.30**	0.51	**0.60**	0.52	**0.70**	0.67	**0.80**	0.66
PEZ [35]	191.03	0.68	193.70	0.69	194.47	0.67	194.40	0.68
VGD [13]	18.70	0.67	40.20	0.68	81.72	0.67	104.20	0.68
Ours	2.43	**0.71**	5.60	**0.71**	6.90	**0.73**	10.50	**0.74**

language-model forward pass and C_{align} is the cost of one CLIP evaluation. Since CLIP must be applied repeatedly to partial prefixes, both costs grow linearly with N. In contrast, a discrete diffusion language model (dLLM) refines the full sequence in T parallel denoising steps, giving complexity $\mathcal{O}(T\,E_{\text{dLLM}} + T\,C_{\text{align}})$, where E_{dLLM} is the cost of one refinement step. Because typically $T \ll N$, dLLM decoding replaces many sequential expansions with a small number of parallel refinements while still enabling CLIP guidance at each step. Our steering module adds only a lightweight potential update with cost $C_{\text{Steer}} \ll E_{\text{dLLM}}$, so the overall asymptotic scaling remains unchanged. Empirically (Tab. 2), captioning methods such as BLIP-2 are fast but yield much lower CLIP-Image (CL-I) scores, while our dLLM-based approach achieves substantial speedups over hard prompt inversion baselines PEZ [35] and VGD [13] ($\sim 10\times$ vs. VGD and $95\times$ vs. PEZ) while also improving CL-I alignment. These results support discrete diffusion LMs as an efficient and accurate backbone for hard prompt inversion.

6.4. Token-level Overlap Analysis

We visualize token-level interactions (as mentioned in Sec. 4.4) during inversion by plotting *token-token similarity heatmaps* accumulated across denoising time steps.

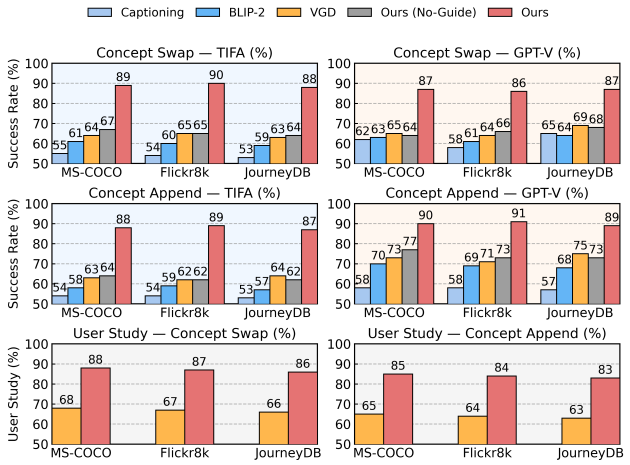

Figure 4. **Edit-friendly prompt inversion enables seamless downstream T2I editing.** We compare concept swap (top) and concept append (middle) operations from inverted prompts across datasets. Interpretable hard inversion (VGD [13]) outperforms captioning and BLIP-2, while our method achieves further gains on both TIFA and GPT-4V scores for swap and append. Controlled user studies (bottom) against VGD show consistent preferences.

Off-diagonal intensity serves as a proxy for *entanglement*: when off-diagonals are high, changing one token (a subject) undesirably perturbs other attributes (background). We observe that although baseline methods such as VGD [13] generate interpretable prompts, they exhibit persistent, off–diagonals, indicating coupled tokens. In contrast, our similarity-regularized decoding rapidly suppresses these off-diagonals and "freezes" confident tokens, yielding *disentangled words* that enable targeted edits without reducing alignment with reference image. We present a heatmap and concept swap comparison with VGD [13] in (Fig. 5) to show that leakage effect in concept swap while our method induces localized, consistent concept-swap result.

7. Ablation Studies

Tab. 3 provides an ablation to demonstrate the role of CLIP guidance vs our method. "No Guide" is the plain dLLM decoder without external signals. "CLIP Guide" adds steering with a CLIP-based reward. "Sim. Guide" adds Token-Token Similarity reduction without CLIP guidance. "Ours" incorporates our Token-Token Similarity reduction and CLIP guidance into prompt inversion. We observe that across settings, adding guidance improves CLIP-I over *No Guide* but increases runtime. **Ours** achieves the strongest text-semantic scores (Prec/Rec/F1 = 0.87/0.91/0.89) and the highest CLIP-I (0.71), indicating the best overall alignment to both text and image with time overhead. However, it is important to note that our method is $\sim 10\times$ faster than the best performing interpretable baselines [13]. **CLIP Guide** attains the lowest perplexity (PPL = 28.84) with moderate time (3.42s). **Sim. Guide** peaks on TIFA (0.92), suggesting better faithfulness to fine-grained attributes.

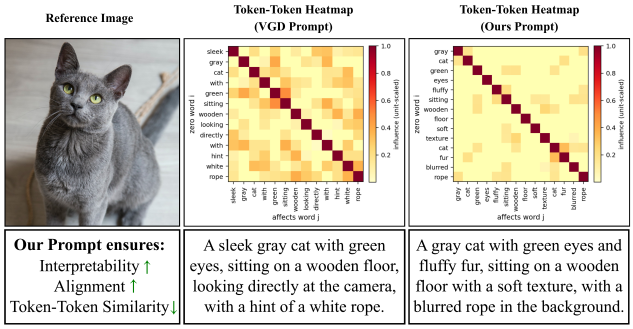

Figure 5. Token–token similarity heatmaps for the VGD prompt (middle) and our prompt (right). Each entry (i, j) measures similarity between the predicted token distributions at positions i and j during decoding (Sec. 4.4). VGD shows strong off-diagonal structure, indicating tightly coupled tokens, whereas our prompt yields a near-diagonal matrix with lower off-diagonal similarity, reflecting more modular tokens and enabling localized, *edit-friendly* word edits.

Table 3. Role of CLIP guidance and Token-Similarity Steering. Metrics: BERTScore (Prec/Rec/F1; ↑), Perplexity (PPL; ↓), CLIP image similarity (CLIP-I; ↑), TIFA (↑), and Time (s; ↓). 32 Tokens

Method	CLIP Model	Prec↑	Rec↑	F1↑	PPL↓	CLIP-I↑	TIFA↑	Time↓
No Guide	—	0.80	0.81	0.80	36.91	0.62	0.74	**2.12**
CLIP Guide	ViT-H-14	0.81	0.83	0.82	**28.84**	0.68	0.81	3.42
Sim. Guide	ViT-H-14	0.78	0.80	0.79	45.93	0.64	**0.92**	4.98
Ours	ViT-H-14	**0.87**	**0.91**	**0.89**	33.16	**0.71**	0.89	5.60

8. Conclusion

We introduce a novel hard prompt inversion framework that combines discrete diffusion language models (dLLMs) with CLIP-guided steering to generate edit-friendly, interpretable prompts. Unlike prior methods, our approach supports downstream concept edits such as token swaps and appends while preserving image attributes unrelated to the modified concept. Across three datasets, our method achieves state-of-the-art performance in interpretability, alignment, and editability, with strong gains in prompt perplexity, CLIP similarity, TIFA, and GPT-V scores. User studies further confirm better human-perceived interpretability and editability. The method also delivers over $10\times$ faster inversion for prompts up to 77 tokens, making it practical for interactive and scalable use.

Acknowledgments

Prof. Lokhande acknowledges support from University at Buffalo startup funds, an Adobe Research Gift, an NVIDIA Academic Grant, and the National Center for Advancing Translational Sciences of the NIH (award UM1TR005296 to the University at Buffalo).

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Sanjay Ahmed, et al. GPT-4 technical report, 2023. 2

[2] Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. 2, 3

[3] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 3

[4] Hila Chefer, Sagie Benaim, Roni Paiss, and Lior Wolf. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics (TOG)*, 42(4):1–11, 2023. 3

[5] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion, 2022. 1

[6] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation with textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 2

[7] Yutong He, Alexander Robey, Naoki Murata, Yiding Jiang, Joshua N. Williams, George J. Pappas, Hamed Hassani, Yuki Mitsufuji, Ruslan Salakhutdinov, and J. Zico Kolter. Automated black-box prompt engineering for personalized text-to-image generation. *arXiv preprint arXiv:2403.19103*, 2024. 3

[8] Amir Hertz, Ron Mokady, Neta Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross-attention control. *arXiv preprint arXiv:2208.01626*, 2022. 3

[9] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 1

[10] Micah Hodosh, Peter Young, and Julia Hockenmaier. Framing image description as a ranking task. *Journal of Artificial Intelligence Research*, 47:853–899, 2013. 4

[11] Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A Smith. Tifa: Accurate and interpretable text-to-image faithfulness evaluation with question answering. *arXiv preprint arXiv:2303.11897*, 2023. 2, 5

[12] Sarthak Jain and Byron C. Wallace. Attention is not explanation. In *Proceedings of NAACL-HLT*, pages 3543–3556, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. 4

[13] Donghoon Kim, Minji Bae, Kyuhong Shim, and Byonghyo Shim. Visually guided decoding: Gradient-free hard prompt inversion with language models. In *International Conference on Learning Representations (ICLR)*, 2025. 3, 4, 5, 6, 7, 8

[14] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models, 2023. 5, 6, 7

[15] 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, 2025. 2

[16] Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B. Hashimoto. Diffusion-LM improves controllable text generation. *arXiv preprint arXiv:2205.14217*, 2022. 2

[17] Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B. Hashimoto. Diffusion-LM improves controllable text generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 3

[18] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. *arXiv preprint arXiv:2301.07093*, 2023. 3

[19] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015. 4

[20] Wenbo Lu, Shaoyi Zheng, Yuxuan Xia, and Shengjie Wang. Toma: Token merge with attention for diffusion models, 2025. 4

[21] Shweta Mahajan, Tanzila Rahman, Kwang Moo Yi, and Leonid Sigal. Prompting hard or hardly prompting: Prompt inversion for text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3

[22] Shweta Mahajan, Tanzila Rahman, Kwang Moo Yi, and Leonid Sigal. Prompting hard or hardly prompting: Prompt inversion for text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

[23] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2024. 2

[24] Junting Pan, Keqiang Sun, Yuying Ge, Hao Li, Haodong Duan, Xiaoshi Wu, Renrui Zhang, Aojun Zhou, Zipeng Qin, Yi Wang, Jifeng Dai, Yu Qiao, and Hongsheng Li. Journeydb: A benchmark for generative image understanding, 2023. 4

[25] pharmapsychotic. Clip interrogator 2.1, 2022. Image-to-prompt tool combining BLIP and CLIP; includes v2.1 release/Colab links. 5, 6, 7

[26] Alec Radford, Jong Wook Kim, Chris Hallacy, et al. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, 2021. 2, 3

[27] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with CLIP latents. *arXiv preprint arXiv:2204.06125*, 2022. 1

[28] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1

[29] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, et al. Dreambooth: Fine tuning text-to-image diffusion models for subject-

driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2

[30] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo-Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models. *arXiv preprint arXiv:2205.11487*, 2022. 1

[31] Yeongbin Seo, Dongha Lee, Jaehyung Kim, and Jinyoung Yeo. Fast and fluent diffusion language models via convolutional decoding and rejective fine-tuning. *arXiv preprint arXiv:2509.15188*, 2025. OpenReview preprint version available. 3

[32] Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen McKeown, and Rajesh Ranganath. A general framework for inference-time scaling and steering of diffusion models. *arXiv:2501.06848*, 2025. Feynman–Kac (FK) Steering. 3, 4

[33] Zirui Wang, Zhizhou Sha, Zheng Ding, Yilin Wang, and Zhuowen Tu. Tokencompose: Text-to-image diffusion with token-level supervision, 2024. 4

[34] 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 *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 2

[35] Yuxin Wen, Yiran Li, Sifei Liu, Xiaolong Wang, et al. Hard prompts made easy: Gradient-based discrete optimization for prompt tuning and inversion. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 2, 5, 7

[36] Sarah Wiegreffe and Yuval Pinter. Attention is not not explanation. In *Proceedings of EMNLP-IJCNLP*, pages 11–20, Hong Kong, China, 2019. Association for Computational Linguistics. 4

[37] Yang Wu, Shilong Wang, Hao Yang, Tian Zheng, Hongbo Zhang, Yanyan Zhao, and Bing Qin. An early evaluation of GPT-4V(ision), 2023. 2

[38] Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a variety of vision tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 5, 6, 7

[39] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 2

[40] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023. 3

[41] Xinlu Zhang, Yujie Lu, Weizhi Wang, An Yan, Jun Yan, Lianke Qin, Heng Wang, Xifeng Yan, William Yang Wang, and Linda Ruth Petzold. Gpt-4v(ision) as a generalist evaluator for vision-language tasks. *arXiv preprint arXiv:2311.01361*, 2023. 5