DRAW WHAT LLMS INTEND: PROMPT ALIGNMENT IS NOT ENOUGH FOR TEXT-TO-IMAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

In Text-to-Image (T2I) generation, which image should we generate among multiple candidates that all satisfy the same prompt? T2I generation is no longer a prompt-to-image black box, but a pipeline in which an upstream model resolves semantics before a generator renders the image. This shift matters because prompts are structurally under-specified and permit multiple images that all preserve the same prompt anchors. Yet prior work primarily measures visual quality or prompt-image alignment, leaving unaddressed whether the generator faithfully realizes the semantic decisions formed by its upstream model. We argue that, among prompt-satisfying images, the visual commitment formed by the upstream model should be carried through to the final image; we call this problem *Latent Intent Transfer* (LIT). In this paper, we study LIT as a new axis for evaluating and improving T2I generation systems. We formulate LIT as the second-stage faithfulness problem of a modern T2I pipeline: whether the generator realizes the upstream model's resolved visual commitments. For evaluation, we introduce two metrics, intent win rate and open-slot commitment accuracy, along with DALI-BENCH, a controlled open-slot benchmark where multiple prompt-valid images remain possible. Guided by the four desired properties, we propose DALI, a plug-and-play method for improving existing T2I pipelines. Extensive experiments show that DALI consistently improves LIT while preserving prompt-anchor fidelity and visual quality.

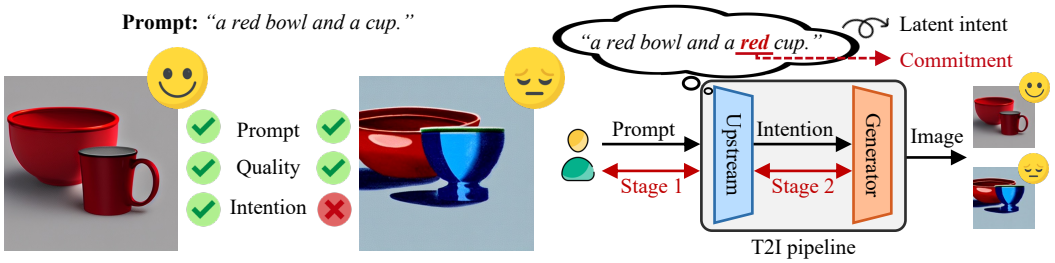
1 INTRODUCTION

Are all prompt-satisfying images equally good? Modern Text-to-Image (T2I) generation can no longer be viewed as a prompt-to-image black box. As T2I pipelines evolve from prompt-conditioned generators (Rombach et al., 2022; Labs et al., 2025) to text-mediated systems (Cho et al., 2023; Yang et al., 2024), latent-bridged hybrids (Hu et al., 2024; Lin et al., 2025), and unified architectures (Chen et al., 2025b; Xie et al., 2025), a functional split between an upstream interpreter and a downstream generator has become increasingly standard. In other words, high-level semantic decisions are made by the upstream LLM, while the diffusion-based generator primarily serves to visualize those decisions (see Section 2). Thus, modern T2I generation is a three-way interaction among the user, the upstream model, and the generator, rather than a direct mapping from prompt to image.

However, existing evaluation and improvement methods still largely treat this entire stack as a prompt-to-image black box. They primarily measure and optimize generic image quality or prompt-image alignment. Specifically, metrics such as IS (Salimans et al., 2016), FID (Heusel et al., 2017), and CLIP-IQA (Wang et al., 2023) assess overall quality, whereas benchmarks such as GenEval (Ghosh et al., 2023), T2I-CompBench (Huang et al., 2023), and DPG-Bench (Hu et al., 2024) evaluate whether explicit prompt anchors are satisfied. Although both are important and necessary, neither can distinguish which image is preferable when multiple images satisfy the same prompt. This limitation matters because prompts cannot practically encode all visual details, thereby allowing multiple valid completions that differ in under-specified properties. A concrete example is shown in Figure 1(a) and discussed further in Section 3.1. This naturally raises a key question beyond prompt alignment: ***Which image should we generate among multiple prompt-satisfying candidates?***

To answer this question within a given pipeline, we evaluate generated images relative to the visual commitment resolved by the pipeline's upstream model. This is because modern T2I pipelines are better understood as two communication stages: from the user to the upstream model, and from the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Prompt alignment is not enough (b) Latent intent transfer

Figure 1: **Latent Intent Transfer:** (a) Among multiple prompt-satisfying possibilities, which image should be drawn by T2I generation pipelines? (b) We claim that the generator should faithfully realize the visual commitments by the upstream model, minimizing the discrepancy in Stage 2.

upstream model to the generator. Accordingly, Figure 1(b) highlights two sources of discrepancy: user-to-upstream miscommunication and upstream-to-generator misrealization. Prior works mostly focus on the first stage, where the goal is to better communicate user intent to the upstream model via prompt engineering (Sahoo et al., 2024; Chen et al., 2025a), rewriting (Chen et al., 2024; Kong et al., 2024), or refinement (Jeon et al., 2025; Khan et al., 2025). By contrast, we focus on the second interface: whether the visual commitment resolved by the upstream model is carried through to the final image. We refer to this problem as ***Latent Intent Transfer*** (LIT), where 'latent intent' refers to the pipeline-internal visual commitment formed within the upstream model.

In this paper, we conduct a systematic study on LIT, spanning its problem formulation, evaluation metrics, benchmark construction, and a plug-and-play algorithm. We begin by formulating the problem as a question of whether the final image reflects what the upstream model has committed to draw (see Section 3.2). To assess this behavior, we introduce two metrics: intent win rate $\mathrm{IWR}(\cdot)$ for comparative evaluation and open-slot commitment accuracy $\mathrm{OpenAcc}(\cdot)$ for diagnostic analysis (see Section 3.3). Moreover, we construct DALI-BENCH, an open-slot benchmark designed to evaluate LIT when multiple prompt-satisfying images remain possible (see Section 4.1).

Then, we identify four properties that any T2I generation method should satisfy to enhance LIT while preserving practical usability (see Section 4.2). These properties are carefully defined to capture intent specificity, prompt-anchor fidelity, visual quality retention, and efficiency. Guided by the properties, we propose **DALI** (**D**raw Wh**a**t **L**LMs **I**ntend), an efficient adaptation method for improving LIT in existing T2I pipelines (see Section 5). DALI combines pairwise preference learning with explicit regularization to preserve prompt anchors and visual quality. To enable efficient adaptation, DALI freezes most of the pipeline and updates only the bridge where LIT failures arise. Extensive experiments demonstrate that DALI consistently and efficiently improves $\mathrm{IWR}(\cdot)$ and $\mathrm{OpenAcc}(\cdot)$ across diverse pipelines while preserving prompt-anchor fidelity and visual quality.

Our contributions are summarized as follows:

- **Problem formulation.** We formulate Latent Intent Transfer (LIT) as a novel problem of assessing whether a generator faithfully realizes the visual commitments resolved by its upstream model.
- **Evaluation framework.** We provide an evaluation framework for LIT consisting of two metrics, intent win rate and open-slot commitment accuracy, and a benchmark DALI-BENCH.
- **Method.** We identify four desired properties for improving LIT and propose DALI to satisfy them. DALI is a plug-and-play adaptation method combining intent-specific supervision with anchor regularization and efficient adaptation.
- **Experiments.** Our empirical validations show that DALI improves intent specificity under lightweight adaptation, while preserving prompt-anchor fidelity and visual quality.

Reproducibility. All of our datasets and implementation are available at `https://anonymous.4open.science/r/DaLI-LIT`.

2 RELATED WORKS

We briefly review the related works. Frequently used notations are formally defined in Appendix A.

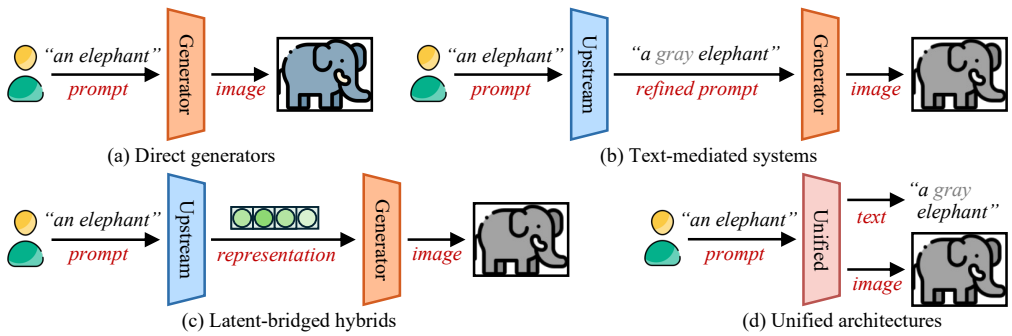


Figure 2: Comparison of T2I generation pipelines: (a) direct generators, (b) text-mediated systems, (c) latent-bridged hybrids, and (d) unified architectures. Refer to Section 2 for details.

Text-to-image generation pipelines. Text-to-Image (T2I) generation has evolved through several pipeline designs, including (a) direct generators, (b) text-mediated systems, (c) latent-bridged hybrids, and (d) unified architectures. Figure 2 summarizes this taxonomy. Direct generators map prompts to images with a single end-to-end generation model, which are typically diffusion-based (Rombach et al., 2022; Saharia et al., 2022; Podell et al., 2024). More recently, text-mediated systems introduce an upstream model that rewrites, plans, or restructures the prompt before image synthesis (Cho et al., 2023; Feng et al., 2024; Lian et al., 2024; Yang et al., 2024). Latent-bridged hybrids further show that this interface need not remain textual: upstream models are connected to generators through learned adapters operating on latent representations (Hu et al., 2024; Wu et al., 2024; Zhao et al., 2024; Lin et al., 2025; Liu et al., 2025). Finally, unified architectures attempt to generate both text and images within a single model (Team, 2024; Wu et al., 2025; Xie et al., 2025). Across these designs, semantic interpretation and visual generation remain conceptually distinguishable. This motivates our formulation of T2I generation as interactions among a user, an upstream model, and a generator.

Preference-based evaluation and alignment. Evaluation of image generation has traditionally centered on generic visual quality, with metrics such as FID and IS measuring sample quality or distributional similarity (Salimans et al., 2016; Heusel et al., 2017). More recent works emphasize prompt-image alignment through reference-free similarity metrics and compositional benchmarks (Hessel et al., 2021; Ke et al., 2021; Wang et al., 2023; Huang et al., 2023; Hu et al., 2023; 2024). In parallel, human-preference datasets and reward models frame evaluation as pairwise or multi-dimensional preference prediction (Kirstain et al., 2023; Xu et al., 2023; Zhang et al., 2024). The preferences also guide post-training alignment of diffusion-based generators, such as RL and DPO methods (Rafailov et al., 2023; Black et al., 2024; Wallace et al., 2024). Following prior works, pairwise preferences from the upstream model guide our evaluation and adaptation process for latent intent transfer.

3 Latent Intent Transfer

We present a counterexample showing that prompt alignment alone is insufficient, formulate the Latent Intent Transfer (LIT) problem, and introduce two metrics for assessing it.

3.1 When Prompt Alignment Is Not Enough

Assume that 1) the user has a target image whose visual properties are fully specified, and 2) there exists a perfect prompt that completely describes and uniquely corresponds to that image. In this ideal case, optimizing and evaluating prompt-image alignment would appear sufficient for moving closer to the desired target image. However, practical T2I generation rarely satisfies either assumption. First, users typically specify only a subset of the visual features of their desired image when prompting a T2I system, often leaving fine-grained details to the pipeline. Second, natural-language prompts inherently remain under-specified and therefore do not uniquely determine a fully specified image. This is because text and images express visual scenes through different modalities; a single image may be described by many prompts, and a single prompt may allow many visual realizations.

We focus on the failure of the second assumption: a single prompt generally admits multiple prompt-satisfying candidates. Figure 1(a) illustrates this issue by comparing two images generated from

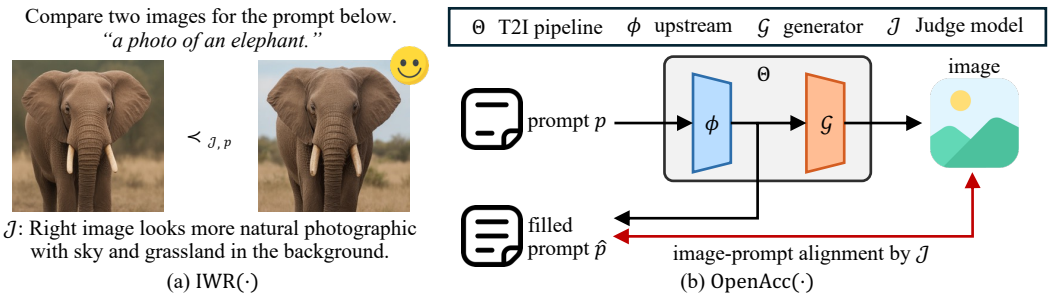


Figure 3: We introduce two metrics to assess LIT: (a) intent win rate $\mathrm{IWR}(\cdot)$ and (b) open-slot commitment accuracy $\mathrm{OpenAcc}(\cdot)$. See Section 3.3 for details.

the prompt *"a red bowl and a cup"*. Despite its simplicity, the prompt under-specifies many visual properties, such as color, size, texture, pose, composition, and additional objects. Although the user does not explicitly specify these properties, an upstream model in an LLM-mediated T2I pipeline should implicitly determine at least some of them to generate an image. The example shows that the two candidates make different choices for ambiguous properties, including the cup color, background, lighting, shadow patterns, and rendering style (realistic vs. 2D-like). Such variation is acceptable in general, but once the upstream model has committed to a specific visual choice, an image inconsistent with that choice is transfer-mismatched despite being prompt-aligned. Therefore, we argue that the generator should regard the upstream model's visual commitment as an indirect prompt, in addition to the user's explicit prompt anchors, and aim to align the generated image with both.

3.2 Problem Definition

Given a T2I generation pipeline Θ, a judge model $\mathcal{J}$, and a calibration dataset $\mathbb{D}_{calib}$, Latent Intent Transfer (LIT) aims to optimize the generation pipeline toward producing images that realize the visual commitments resolved by $\mathcal{J}$. Specifically, the upstream model ϕ serves as the judge model $\mathcal{J}$ whose latent intent is to be transferred. We provide the formal definition as Problem 1. Note that the upstream model ϕ and the generator $\mathcal{G}$ are conceptually distinguished by their functional roles, but they are not required to be separate modules and may instead be unified in a single architecture.

Problem 1 (Latent Intent Transfer)**.** *We have a T2I generation pipeline Θ consisting of an upstream model ϕ, a generator $\mathcal{G}$, and bridge modules β connecting them, with a sample dataset $\mathbb{D}_{calib} = \{(p_i)\}_{i=1}^{S}$ of size S. The goal is to obtain a T2I generation pipeline Θ^* that maximizes the expected latent utility $U_\phi(x \mid p)$ assigned by ϕ to pipeline-generated images x for prompts $p \sim \mathbb{D}_{test}$. Let $\pi_\Theta(\cdot \mid p)$ denote the image distribution from a pipeline Θ given prompt p. Then,*

$$\Theta^* = \arg\max_{\Theta} \mathbb{E}_{p\sim\mathbb{D}_{test}, x\sim\pi_\Theta(\cdot|p)}[U_\phi(x \mid p)].$$

3.3 Performance Metrics

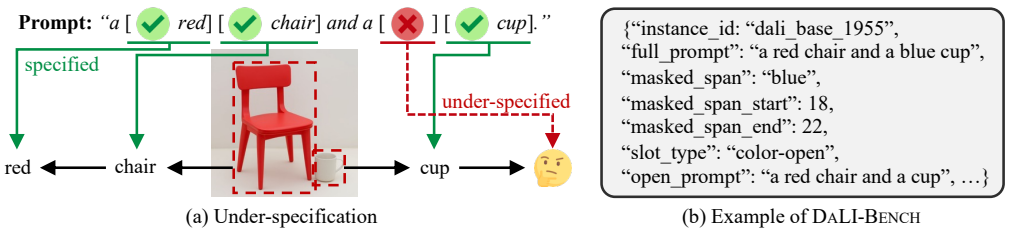
How can we assess the latent utility $U_\phi(\cdot)$ of upstream model ϕ? We introduce two metrics for LIT evaluation: Intent Win Rate (IWR) $\mathrm{IWR}(\cdot)$ and open-slot commitment accuracy $\mathrm{OpenAcc}(\cdot)$. $\mathrm{IWR}(\cdot)$ captures relative preference by comparing images, and $\mathrm{OpenAcc}(\cdot)$ externalizes the upstream model's commitment as a filled prompt for absolute evaluation. Figure 3 illustrates both metrics.

Comparative metric. Latent utility is inherently implicit and is not directly observable, making it challenging to assign absolute utility scores. We therefore introduce Intent Win Rate (IWR) as a comparative metric based on pairwise preference. IWR computes the win rate of a target pipeline against its baseline, where a win means that the judge $\mathcal{J}$ prefers the target image for prompt p.

Definition 1 (Intent Win Rate)**.** *We have a judge model $\mathcal{J}$ and n triples $\{(p_i, x_i, x_i')\}_{i=1}^{n}$, where images x_i and x_i' are generated from prompt p_i by pipelines Θ and Θ', respectively. Intent win rate*

$$\mathrm{IWR}(\Theta'; \Theta) := \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{\{x_i' \succ_{\mathcal{J},p_i} x_i\}},$$

where $x_i' \succ_{\mathcal{J},p_i} x_i$ denotes that the judge $\mathcal{J}$ prefers x_i' over x_i under prompt p_i, and $\mathbf{1}_{\{\cdot\}}$ is the indicator function that returns 1 if the inner statement is true and 0 otherwise.



(a) Under-specification (b) Example of DALI-BENCH

Figure 4: We construct DALI-BENCH to evaluate LIT by controlling under-specification. (a) A visual property is under-specified when it is not stated in the prompt, such as the color of the cup in the open prompt. (b) We obtain a DALI-BENCH instance by masking such a property from a full prompt.

For fair comparison, we allow ties to avoid forcing preferences, but exclude tied comparisons from the IWR computation (see Appendix C.2). IWR corresponds to LIT evaluation when the judge $\mathcal{J}$ is instantiated as the upstream model ϕ, which we call self-IWR $\mathrm{IWR}_\phi(\cdot)$. $\mathcal{J}$ may also be either an external model $\psi \neq \phi$ or a human evaluator. We refer to these variants as cross-IWR $\mathrm{IWR}_\psi(\cdot)$ and human-IWR $\mathrm{IWR}_H(\cdot)$, respectively, and report them for judge robustness analysis (see Section 6.3).

Absolute metric. While pairwise preference provides a practical comparative metric, it is applicable only when the upstream model ϕ can judge pairs of images. To address this limitation, we introduce open-slot commitment accuracy $\mathrm{OpenAcc}(\cdot)$ as an absolute metric applicable even to text-only LLMs. $\mathrm{OpenAcc}(\cdot)$ evaluates whether the generated image aligns with the filled prompt $\hat{p}$ produced by ϕ from the original prompt p. Here, the judge $\mathcal{J}$ should be an external judge distinct from ϕ, to avoid using ϕ as both the source of prompt commitment and the alignment evaluator.

Definition 2 (Open-slot Commitment Accuracy)**.** *We have a T2I generation pipeline Θ, an external judge $\mathcal{J} \neq \phi$, and a set $P = \{p_i\}_{i=1}^n$ of n prompts p_i. Open-slot commitment accuracy*

$$\mathrm{OpenAcc}(\Theta; P, \mathcal{J}) := \frac{1}{n} \sum_{i=1}^{n} A_{\mathcal{J}}(x_i, \hat{p}_i),$$

where $x_i \sim \pi_\Theta(\cdot \mid p_i)$ is the image generated by Θ, $\hat{p}_i$ is the filled prompt produced by the upstream model ϕ for p_i, and $A_{\mathcal{J}}(x_i, \hat{p}_i) \in \{0, 1\}$ indicates whether $\mathcal{J}$ judges x_i to align with $\hat{p}_i$.

4 BENCHMARK AND DESIGN PRINCIPLES

We construct an open-slot evaluation dataset DALI-BENCH and propose four desired properties.

4.1 DALI-BENCH: OPEN-SLOT EVALUATION DATASET

Under-specification. What kind of dataset is needed to evaluate LIT? As LIT concerns visual properties left open by the prompt and resolved by the upstream model, evaluation requires a systematic categorization of those missing properties. We call these prompt-omitted visual properties as *under-specification*. Figure 4(a) illustrates the concept of under-specification. We select seven categories of under-specification for constructing DALI-BENCH: color, material, size, texture, shape, lighting, and background. Further details on each category are given in Appendix B.1.

Proposed dataset. How can we control under-specification for evaluation? We propose DALI-BENCH, an open-slot evaluation benchmark for LIT. Starting from prompts in existing prompt-alignment datasets (Ghosh et al., 2023; Huang et al., 2023; Hu et al., 2024), we create open-slot prompts by masking words that instantiate the chosen under-specification categories. This inherits the reliability of existing prompt-alignment datasets while enabling controlled analysis of under-specification. Figure 4(b) describes an instance from DALI-BENCH. To support different evaluation settings, we provide three releases of DALI-BENCH: small, base, and multi. We adopt the base release as default unless stated otherwise. We provide details on the three dataset releases and construction process in Appendices B.2 and B.3, respectively.

4.2 DESIRED PROPERTIES

Given this evaluation setup, we now consider how to improve LIT in T2I generation pipelines. We outline four desired properties that LIT-improving methods in T2I generation pipelines should satisfy.

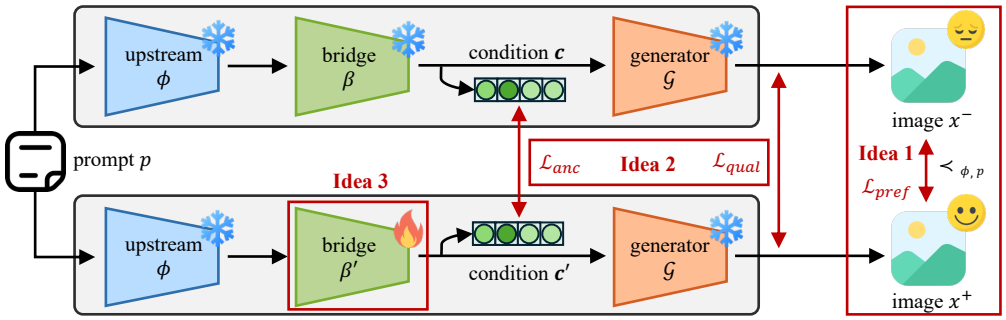


Figure 5: Overall architecture of DALI. Our main ideas are 1) pairwise preference optimization, 2) prompt fidelity and quality regularization, and 3) selective bridge adaptation.

- **Intent specificity.** The primary goal of LIT is to better achieve its core objective: the faithful transfer of latent intent. This is reflected in higher values of the proposed LIT metrics, which indicate stronger alignment between the upstream model and the generator.
- **Prompt-anchor fidelity.** Although latent intent may function as an indirect prompt, it is meaningful only when the explicit prompt anchors are preserved. Accordingly, improving LIT should not reduce the prompt-alignment performance measured by existing evaluation protocols.
- **Visual quality retention.** Visual quality has long been a central objective in image generation, where outputs should remain perceptually appealing to humans. Improving LIT should therefore retain image quality, rather than trading visual fidelity for intent transfer.
- **Efficiency.** Updating the entire T2I pipeline may incur large computational and memory overhead. For practical deployment, the method should avoid costly global updates and remain efficient.

5 Proposed Method

We propose **DALI** (**D**raw Wh**a**t **LLM**s **I**ntend), a lightweight plug-and-play adaptation method for improving LIT in existing T2I pipelines. These are the three main challenges that should be tackled:

C1. Hidden intent supervision. Latent intent is not directly observable and is revealed through comparisons among images. How can we learn from a target that is indirectly revealed?

C2. Preference confounding. Attempts to improve LIT may also increase preference through prompt-anchor violations or changes in overall visual quality, rather than through genuine LIT. How can we improve LIT without relying on such confounds?

C3. Costly global adaptation. Updating the full T2I pipeline is expensive and may perturb components unrelated to LIT. How can we improve LIT efficiently with minimal overhead?

We propose three main ideas to address the challenges:

I1. Pairwise preference optimization (Section 5.1). We perform transfer optimization through pairwise preference learning and instantiate it with a DPO-style objective. This converts latent intent directly into revealed preference supervision.

I2. Prompt fidelity and quality regularization (Section 5.2). We add explicit regularization to preserve prompt-anchor fidelity and overall visual quality during optimization. This reduces spurious gains by discouraging improvements from anchor violations or quality degradations.

I3. Selective bridge adaptation (Section 5.3). We freeze most of the existing T2I pipeline, and update only a small set of modules that mediate intent transfer into generation. This makes adaptation efficient while directly targeting the source of LIT failures.

Figure 5 depicts the overall procedure of DALI. Our intuition is that only a small set of components bridging the two models should be updated, guided by pairwise preferences and regularization losses. The update is guided by three objectives: preference loss $\mathcal{L}_{pref}$, prompt-anchor loss $\mathcal{L}_{anc}$, and quality loss $\mathcal{L}_{qual}$. We formulate the final loss function as follows:

$$\mathcal{L}_{\text{DALI}} = \mathcal{L}_{pref} + \lambda_{anc}\mathcal{L}_{anc} + \lambda_{qual}\mathcal{L}_{qual}, \tag{1}$$

where λ_{anc} and λ_{qual} are loss hyperparameters. Note that DALI is compatible with any T2I pipelines with upstream ϕ and generator $\mathcal{G}$. We formulate the algorithm of DALI in Appendix D.

Table 1: Evaluation of LIT, prompt-anchor fidelity, and visual quality across diverse T2I pipelines.

Pipeline	Model		LIT		Prompt	Quality		
	Upstream ϕ	Generator $\mathcal{G}$	$\text{IWR}_\phi(\cdot)^*\uparrow$	$\text{OpenAcc}(\cdot)\uparrow$	Acc. $\uparrow$	FID $\downarrow$	MUSIQ $\uparrow$	C-I+■ $\uparrow$
Direct Gen.	✗	GLIGEN	✗	20.1†	50.4	19.4	67.5	**68.5**
VPGen	Vicuna 13B		(text-only ϕ)	**21.9**	**60.5**	**15.9**	**68.6**	68.3
Direct Gen.	✗	SDXL	44.0	36.2†	82.6	**15.3**	**72.9**	72.0
RPG	Qwen2.5-VL 7B		**56.0**	**41.5**	**86.2**	15.5	**72.9**	**72.5**
ELLA	Flan-T5-XL	SD1.5	✗	40.6	**86.4**	**19.4**	72.2	72.4
+ DALI			(text-only ϕ)	**43.7**	86.0	20.1	**72.7**	**73.0**
Bifrost-1	Qwen2.5-VL 7B	FLUX.1	44.2	45.5	89.3	**23.0**	74.0	73.1
+ DALI			**55.8**	**48.6**	**91.3**	24.7	**74.6**	**73.6**
Janus	Janus-1.3B		45.6	43.7	**90.5**	**8.5**	65.0	67.8
+ DALI			**54.4**	**48.3**	90.1	9.1	**65.8**	**68.0**
Janus-Pro	Janus-Pro-1B		42.8	44.8	**89.2**	**16.1**	68.6	71.1
+ DALI			**57.2**	**48.6**	88.6	17.2	**69.0**	**72.4**

* Comparison between paired methods, † Upstream ϕ from corresponding text-mediated systems, ■ CLIP-IQA+ score $\times 100$.

5.1 Pairwise Preference Optimization

Latent intent is not directly observable as an explicit target image. However, it may be revealed through pairwise comparisons among candidate generations. For a prompt p_i, we generate candidate images and query the upstream model ϕ to select the image that better matches its visual commitment. This yields preference triples $\{(p_i, x_i^+, x_i^-)\}_{i=1}^B$, where $x_i^+ \succ_{\phi,p_i} x_i^-$ denotes that ϕ prefers x_i^+ over x_i^- under prompt p_i. We then optimize the trainable parameters so that the adapted pipeline Θ' assigns higher score to x_i^+ compared to x_i^-. Accordingly, we employ a DPO-style preference loss

$$\mathcal{L}_{pref} = -\frac{1}{B}\sum_{i=1}^{B} \log \sigma\left(\tau\left[\Delta_{\Theta'}(p_i, x_i^+, x_i^-) - \Delta_{\Theta}(p_i, x_i^+, x_i^-)\right]\right), \tag{2}$$

where $\sigma(\cdot)$ is the sigmoid function, τ is a temperature hyperparameter, Θ is the reference pipeline before adaptation, and $\Delta_\Theta(p_i, x_i^+, x_i^-) = s_\Theta(x_i^+ \mid p_i) - s_\Theta(x_i^- \mid p_i)$ denotes the preference margin computed from conditional score $s_\Theta(\cdot)$. We detail our implementation choices in Appendix C.1.

5.2 Prompt Fidelity and Quality Regularization

Pairwise preference optimization encourages the pipeline to follow the upstream model's revealed intent, but it may drift from explicit prompt anchors or alter the generator's visual behavior. We therefore add two reference-based regularizers: prompt-anchor loss $\mathcal{L}_{anc}$ and quality loss $\mathcal{L}_{qual}$.

The prompt-anchor loss $\mathcal{L}_{anc}$ preserves prompt semantics by minimizing the gap between adapted $\mathbf{c}'$ and reference $\mathbf{c}$ conditions from adapted β' and original β bridges, respectively. This cosine regularizer keeps the adapted condition semantically close to the reference condition, while allowing intent-specific changes needed for LIT. For each prompt p_i, let representation $\mathbf{z}_i = \phi(p_i)$, adapted condition $\mathbf{c}'_i = \beta'(\mathbf{z}_i)$, and reference condition $\mathbf{c}_i = \beta(\mathbf{z}_i)$. We define the prompt-anchor loss

$$\mathcal{L}_{anc} = 1 - \frac{1}{B}\sum_{i=1}^{B} \cos\left(\mathbf{c}'_i, \mathbf{c}_i\right). \tag{3}$$

The quality loss $\mathcal{L}_{qual}$ preserves the generator's visual behavior by discouraging from changing the generator's prediction. This regularizer keeps the adapted condition from altering the generator's reference behavior, thereby preserving visual quality. For prompt p_i, let g_i be the generator input state shared across $\mathbf{c}'_i$ and $\mathbf{c}_i$ conditions. We compare generator predictions under $\mathbf{c}'_i$ and $\mathbf{c}_i$:

$$\mathcal{L}_{qual} = \frac{1}{B}\sum_{i=1}^{B} \left\|\mathcal{G}\left(g_i, \mathbf{c}'_i\right) - \mathcal{G}\left(g_i, \mathbf{c}_i\right)\right\|_2^2. \tag{4}$$

5.3 Selective Bridge Adaptation

Full-pipeline adaptation is costly and may perturb components unrelated to LIT. Therefore, our idea is to update only a small subset of parameters that mediate LIT, rather than adapting the entire pipeline. Specifically, we freeze most parameters θ_{fr} and optimize only trainable parameters θ_{train} (i.e., $\min_{\theta_{train}} \mathcal{L}_{\text{DALI}}(\theta_{train}; \theta_{fr})$.) This raises a key question: which parameters should be selected

Table 2: Parameter and runtime efficiency of DALI's local adaptation. The update is measured over a 100-epoch run on a single A100 GPU.

Pipeline	# of trainable parameters	Maximum memory	Per-epoch train time
ELLA	2.2M (3.3%)	8.5 GB	3.3 min.
Bifrost-1	12.8M (0.1%)	68.3 GB	75.9 min.
Janus	42.1M (2.2%)	14.1 GB	14.5 min.
Janus-Pro	42.1M (2.2%)	29.2 GB	14.9 min.

Table 3: Preference results and $\mathrm{IWR}_{\mathcal{J}}(\cdot)$ [%] under six model and human judges.

Judge $\mathcal{J}$	Preference [%]			$\mathrm{IWR}_{\mathcal{J}}(\cdot)$
	Base	Tie	DALI	
Qwen2.5-VL 7B	34.2	22.7	43.1	55.8
InternVL-3.5 4B	35.0	23.8	41.2	54.1
Qwen2.5-VL 32B	32.4	24.9	42.7	56.9
Qwen3-VL 32B	36.7	5.6	57.7	61.1
LLaVA-OV2 8B	35.5	13.8	50.7	58.8
GPT-5.6 Terra	41.3	6.8	51.9	55.7
Human	32.7	24.9	42.4	56.5

as θ_{train}? In LLM-mediated T2I pipelines, an upstream model ϕ is typically connected to a generator $\mathcal{G}$ through bridge modules β that translate upstream representations into generator conditioning. As LIT concerns whether the upstream visual commitment is faithfully rendered by the generator $\mathcal{G}$, this bridge β is the most direct adaptation target. Accordingly, we instantiate θ_{train} as the parameters of β, while frozen parameters θ_{fr} includes upstream model ϕ, generator $\mathcal{G}$, and all non-bridge modules.

6 EXPERIMENTS

We conduct experiments to empirically analyze LIT and demonstrate that DALI improves it.

6.1 EXPERIMENTAL SETUP

We briefly introduce the experiment setup. Further details are within Appendix E.

Baselines and datasets. We adopt two text-mediated systems (VPGen (Cho et al., 2023) and RPG (Yang et al., 2024)), two latent-bridged hybrids (ELLA (Hu et al., 2024) and Bifrost-1 (Lin et al., 2025)), and two unified architectures (Janus (Wu et al., 2025) and Janus-Pro (Chen et al., 2025b)) as baselines. We conduct property validation primarily on DALI-BENCH, and evaluate visual quality with FID (Heusel et al., 2017), MUSIQ (Ke et al., 2021), and CLIP-IQA+ (Wang et al., 2023) metrics.

Implementation details. We follow the settings from prior works (Hu et al., 2024; Lin et al., 2025) for fair comparison. All experiments are conducted on a workstation with Intel Xeon Gold 6338 and NVIDIA A100 80GB. We design each experiment to run within a single-GPU budget.

6.2 MAIN RESULTS AND PROPERTY VALIDATION

We evaluate whether DALI satisfies the desired properties introduced in Section 4.2. Tables 1 and 2 report the main results on property validation across diverse T2I pipelines.

Latent intent transfer. 'LIT' columns of Table 1 shows that DALI improves LIT over the corresponding baseline pipelines. The improvement in $\mathrm{OpenAcc}(\cdot)$ indicates that generated images better follow the upstream model's open-slot commitments. When the upstream model ϕ can judge images, $\mathrm{IWR}_{\phi}(\cdot)$ further measures whether the adapted outputs are preferred under its own latent intent.

Prompt-anchor fidelity. Improving LIT should not come at the cost of violating the original prompt. 'Prompt' column of Table 1 shows that DALI maintains prompt alignment with only a small decrease while improving LIT. This indicates that the adapted pipeline does not simply increase latent-intent agreement by ignoring prompt anchors. Moreover, while prompt-anchor fidelity ranges from 50% to 90%, $\mathrm{OpenAcc}(\cdot)$ remains much lower at only 20% to 50%, leaving a gap of roughly 40%p in many cases. This suggests that LIT captures failures beyond prompt alignment.

Visual quality retention. We also evaluate whether LIT improvement affects visual quality. As shown in the 'Quality' column of Table 1, DALI maintains competitive FID, MUSIQ, and CLIP-IQA+ metrics compared with the corresponding baseline pipelines. This suggests that the adaptation does not substantially disrupt the generator's visual prior. Together with prompt-anchor fidelity, this confirms that LIT gains are not obtained by sacrificing general generation quality.

Efficiency of local adaptation. Table 2 evaluates the efficiency of local adaptation. DALI updates only the bridge modules, substantially reducing the number of trainable parameters and making adaptation feasible under a single-GPU budget. Moreover, per-epoch training takes only three to seventy-five minutes across pipelines, showing the practicality of DALI on existing T2I pipelines.

Table 4: Commitment value consistency and pairwise preference-label agreement. 'Commitment' reports modal consensus, whereas 'Preference' reports label agreement on fixed image pairs.

Perturbation	Commitment [%]		Preference [%]
	Strict	Semantic	
Repeated run	81.8	93.4	98.5
Prompt paraphrase	75.1	86.7	96.1
Cross-model	48.0	60.7	74.8

Table 5: Ablation study of DALI. 'Prompt' denotes prompt alignment accuracy [%].

I1	I2	I3	$\mathrm{OpenAcc}(\cdot)\uparrow$	Prompt ↑	FID ↓
Baseline			40.6	86.4	19.4
✗			Same as baseline		
		✗	OOM		
✓	✗	$✗^L$	42.8	82.0	30.5
✓	✓	$✗^L$	42.0	84.2	27.2
✓	✗	✓	44.2	81.7	24.8
✓	✓	✓	43.7	86.0	20.1

L: LoRA adapter with same number of parameters.

6.3 Robustness Across Judges

We evaluate whether LIT improvements are robust to judge choices in Table 3. As preference-based evaluation may depend on its judge, we compare self-IWR with cross-model and human evaluations.

Cross-model evaluation. In cross-model evaluation, we replace the upstream-model judge ϕ with an external model ψ. This setting tests whether the measured LIT improvement is robust to the choice of judge model. Bifrost-1 (Lin et al., 2025) serves as the baseline. The results show that the preference trend remains consistent across different model judges. This suggests that the improvement is not specific to an upstream judge, but reflects visual-semantic preferences shared across models.

Human evaluation. We further conduct human evaluation to validate whether the preference trend agrees with human judgment. Fifty human annotators each compare forty randomly-sampled image pairs generated from the same prompt and select the preferred image for each pair. The results are consistent with the model-based evaluations, suggesting that model preferences are broadly aligned with humans in our setting. We present the example image pairs and prompts in Appendix C.2.

6.4 Diagnostic Analyses

We conduct diagnostic analyses to examine the output consistency and the contribution of each idea. Further results such as open-slot type or hyperparameter analyses are provided in Appendix F.

Output consistency. Table 4 evaluates the consistency of commitment values and pairwise preference labels across repeated runs, prompt paraphrases (by GPT-5.6-Terra), and different models. We compute semantic modal consensus by requiring meaning-equivalent (judged by GPT-5.6-Terra) commitments, with the cross-model setting comparing Flan-T5-XL, Qwen2.5-VL-7B, and InternVL-3.5-4B. We have two observations from the results. First, both commitments and preference labels are highly reproducible for a fixed model, supporting the reliability of the conditional target and preference labels. Second, the cross-model row shows lower agreement for both commitment values and pairwise preference labels. This indicates that visual commitments and preferences are reproducible within a fixed upstream model but partly specific to that model.

Ablation study. Table 5 evaluates the effect of three main components of DALI described in Section 5. ELLA (Hu et al., 2024) serves as the baseline, and OOM denotes out-of-memory on a single A100 80GB GPU. We include a parameter-matched LoRA (Hu et al., 2022) applied to the generator layers to compare bridge-side and generator-side adaptation. We have two observations from the results. First, I3 achieves higher $\mathrm{OpenAcc}(\cdot)$ and substantially lower FID than parameter-matched LoRA while maintaining comparable or better prompt alignment. Second, removing any component weakens at least one desired property, confirming that all the components are complementary and essential.

7 Conclusion

We investigate the problem of Latent Intent Transfer (LIT) in T2I generation pipelines: whether the downstream generator faithfully renders the visual commitment by the upstream model. We provide a systematic formulation of LIT, introduce an evaluation protocol with two metrics and a benchmark DALI-Bench, derive four desired properties, and propose DALI, an efficient plug-and-play adaptation method. Future directions include extending our framework to full-pipeline training, longer prompt inputs, and other output modalities such as video and audio (see Appendix G.2).

AI USE STATEMENT

We employ ChatGPT[1] (GPT-5.5 Pro and GPT-5.6 Sol) for language polishing purposes, specifically for improving grammar and clarity at the sentence level. We do not use them for research-related tasks, including theoretical derivation and result analysis. The authors take all the responsibility for the final content of this work.

REFERENCES

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, et al. Qwen2.5-vl technical report, 2025.

Eslam Mohamed Bakr, Pengzhan Sun, Xiaoqian Shen, Faizan Farooq Khan, Li Erran Li, and Mohamed Elhoseiny. Hrs-bench: Holistic, reliable and scalable benchmark for text-to-image models. In *ICCV*, 2023.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *ICLR*, 2024.

Tingfeng Cao, Chengyu Wang, Bingyan Liu, Ziheng Wu, Jinhui Zhu, and Jun Huang. Beautifulprompt: Towards automatic prompt engineering for text-to-image synthesis. In *EMNLP*, 2023.

Agneet Chatterjee, Gabriela Ben Melech Stan, Estelle Aflalo, Sayak Paul, Dhruba Ghosh, Tejas Gokhale, Ludwig Schmidt, Hannaneh Hajishirzi, Vasudev Lal, Chitta Baral, et al. Getting it right: Improving spatial consistency in text-to-image models. In *ECCV*, 2024.

Chieh-Yun Chen, Min Shi, Gong Zhang, and Humphrey Shi. T2I-Copilot: A training-free multi-agent text-to-image system for enhanced prompt interpretation and interactive generation. In *CVPR*, 2025a.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025b.

Zijie Chen, Lichao Zhang, Fangsheng Weng, Lili Pan, and Zhenzhong Lan. Tailored visions: Enhancing text-to-image generation with personalized prompt rewriting. In *CVPR*, 2024.

Jaemin Cho, Abhay Zala, and Mohit Bansal. Visual programming for step-by-step text-to-image generation and evaluation. In *NeurIPS*, 2023.

Siddhartha Datta, Alexander Ku, Deepak Ramachandran, and Peter Anderson. Prompt expansion for adaptive text-to-image generation. In *ACL*, 2024.

Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Reddy Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-free structured diffusion guidance for compositional text-to-image synthesis. In *ICLR*, 2023a.

Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Reddy Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-free structured diffusion guidance for compositional text-to-image synthesis. In *ICLR*, 2023b.

Weixi Feng, Wanrong Zhu, Tsu-jui Fu, Varun Jampani, Arjun Akula, Xuehai He, Sugato Basu, Xin Eric Wang, and William Yang Wang. Layoutgpt: Compositional visual planning and generation with large language models. In *NeurIPS*, 2023c.

Yutong Feng, Biao Gong, Di Chen, Yujun Shen, Yu Liu, and Jingren Zhou. Ranni: Taming text-to-image diffusion for accurate instruction following. In *CVPR*, 2024.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. In *NeurIPS*, 2023.

[1] https://chatgpt.com/

Yaru Hao, Zewen Chi, Li Dong, and Furu Wei. Optimizing prompts for text-to-image generation. In *NeurIPS*, 2023.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: A reference-free evaluation metric for image captioning. In *EMNLP*, 2021.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*, 2017.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022.

Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. Ella: Equip diffusion models with llm for enhanced semantic alignment. *arXiv preprint arXiv:2403.05135*, 2024.

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. In *ICCV*, 2023.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. In *ICLR*, 2024.

Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. In *NeurIPS*, 2023.

Nam Hyeon-Woo, Moon Ye-Bin, Wonseok Choi, Lee Hyun, and Tae-Hyun Oh. VLM's eye examination: Instruct and inspect visual competency of vision language models. *TMLR*, 2025.

Jinwoo Jeon, JunHyeok Oh, Hayeong Lee, and Byung-Jun Lee. Iterative prompt refinement for safer text-to-image generation. In *EMNLP*, 2025.

Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. Musiq: Multi-scale image quality transformer. In *ICCV*, 2021.

Mohammed Abdul Hafeez Khan, Yash Jain, Siddhartha Bhattacharyya, and Vibhav Vineet. Test-time prompt refinement for text-to-image models. In *ICCV Workshops*, 2025.

Minjun Kim, Jongjin Kim, and U Kang. Synq: Accurate zero-shot quantization by synthesis-aware fine-tuning. In *ICLR*, 2025.

Yunji Kim, Jiyoung Lee, Jin-Hwa Kim, Jung-Woo Ha, and Jun-Yan Zhu. Dense text-to-image generation with attention modulation. In *ICCV*, 2023.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. In *NeurIPS*, 2023.

Weize Kong, Spurthi Hombaiah, Mingyang Zhang, Qiaozhu Mei, and Michael Bendersky. Prewrite: Prompt rewriting with reinforcement learning. In *ACL Short Paper*, 2024.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, and more. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025.

Zhengfeng Lai, Vasileios Saveris, Chen Chen, Hong-You Chen, Haotian Zhang, Bowen Zhang, Wenze Hu, Juan Tebar, Zhe Gan, Peter Grasch, et al. Revisit large-scale image-caption data in pre-training multimodal foundation models. In *ICLR*, 2025.

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. In *CVPR*, 2023.

Long Lian, Boyi Li, Adam Yala, and Trevor Darrell. LLM-grounded diffusion: Enhancing prompt understanding of text-to-image diffusion models with large language models. *Transactions on Machine Learning Research*, 2024.

Yijun Liang, Ming Li, Chenrui Fan, Ziyue Li, Dang Nguyen, Kwesi Adu Cobbina, Shweta Bhardwaj, Jiuhai Chen, Fuxiao Liu, and Tianyi Zhou. Colorbench: Can VLMs see and understand the colorful world? a comprehensive benchmark for color perception, reasoning, and robustness. In *NeurIPS*, 2025.

Han Lin, Jaemin Cho, Amir Zadeh, Chuan Li, and Mohit Bansal. Bifrost-1: Bridging multimodal LLMs and diffusion models with patch-level CLIP latents. In *NeurIPS*, 2025.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

Mushui Liu, Yuhang Ma, Zhen Yang, Jun Dan, Yunlong Yu, Zeng Zhao, Zhipeng Hu, Bai Liu, and Changjie Fan. Llm4gen: Leveraging semantic representation of llms for text-to-image generation. In *AAAI*, 2025.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-eval: Nlg evaluation using gpt-4 with better human alignment. In *EMNLP*, 2023.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

Arjun Panickssery, Samuel R. Bowman, and Shi Feng. LLM evaluators recognize and favor their own generations. In *NeurIPS*, 2024.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *ICLR*, 2024.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*, 2023.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *NeurIPS*, 2022.

Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. A systematic survey of prompt engineering in large language models: Techniques and applications. *arXiv preprint arXiv:2402.07927*, 2024.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *NeurIPS*, 2016.

Preethi Seshadri, Sameer Singh, and Yanai Elazar. The bias amplification paradox in text-to-image generation. In *NAACL*, 2024.

Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024.

Xinyu Tian, Shu Zou, Zhaoyuan Yang, and Jing Zhang. Identifying and mitigating position bias of multi-image vision-language models. In *CVPR*, 2025.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *CVPR*, 2024.

Jianyi Wang, Kelvin CK Chan, and Chen Change Loy. Exploring clip for assessing the look and feel of images. In *AAAI*, 2023.

Ziqi Wang, Hanlin Zhang, Xiner Li, Kuan-Hao Huang, Chi Han, Shuiwang Ji, Sham M. Kakade, Hao Peng, and Heng Ji. Eliminating position bias of language models: A mechanistic approach. In *ICLR*, 2025.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, et al. Transformers: State-of-the-art natural language processing. In *EMNLP*, 2020.

Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. In *CVPR*, 2025.

Mingrui Wu, Lu Wang, Pu Zhao, Fangkai Yang, Jianjin Zhang, Jianfeng Liu, Yuefeng Zhan, Weihao Han, Hao Sun, Jiayi Ji, Xiaoshuai Sun, Qingwei Lin, Weiwei Deng, Dongmei Zhang, Feng Sun, and Rongrong Ji. Reprompt: Reasoning-augmented reprompting for text-to-image generation via reinforcement learning. In *ICLR*, 2026.

Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. Next-gpt: Any-to-any multimodal llm. In *ICML*, 2024.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. In *ICLR*, 2025.

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 *NeurIPS*, 2023.

Hongji Yang, Yucheng Zhou, Wencheng Han, and Jianbing Shen. Self-rewarding large vision-language models for optimizing prompts in text-to-image generation. 2025.

Ling Yang, Zhaochen Yu, Chenlin Meng, Minkai Xu, Stefano Ermon, and Bin Cui. Mastering text-to-image diffusion: Recaptioning, planning, and generating with multimodal llms. In *ICML*, 2024.

Sixian Zhang, Bohan Wang, Junqiang Wu, Yan Li, Tingting Gao, Di Zhang, and Zhongyuan Wang. Learning multi-dimensional human preference for text-to-image generation. In *CVPR*, 2024.

Shihao Zhao, Shaozhe Hao, Bojia Zi, Huaizhe Xu, and Kwan-Yee K Wong. Bridging different language models and generative vision models for text-to-image generation. In *ECCV*, 2024.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *NeurIPS*, 2023.

A NOTATION

We summarize the frequently used notations in the paper as Table 6.

Table 6: Frequently used notations.

Symbol	Description
Θ	Text-to-Image (T2I) generation pipeline
ϕ	Upstream model
$\mathcal{G}$	Generator
β	Bridge modules
$\mathcal{J}$	Judge model
ψ	An external model
x_i, p_i	i-th image and its corresponding prompt
$\mathbb{D}_{calib}$, $\mathbb{D}_{test}$	Calibration and test datasets, respectively
$U_\phi(x \mid p)$	Latent utility assigned by ϕ to image x generated from prompt p
$\pi_\Theta(\cdot\|p)$	Image distribution induced by pipeline Θ given prompt p
$\hat{p}$	Filled prompt by upstream model ϕ for prompt p
$\mathrm{IWR}(\cdot)$	Intent Win Rate (IWR)
$\mathrm{IWR}_\phi(\cdot)$, $\mathrm{IWR}_\psi(\cdot)$, $\mathrm{IWR}_H(\cdot)$	Self-, cross-, and human-IWR, respectively
$\mathrm{OpenAcc}(\cdot)$	Open-slot commitment accuracy
$A_\mathcal{J}(x_i, \hat{p}_i)$	Indicator function for whether $\mathcal{J}$ judges x_i to align with $\hat{p}_i$
$\mathcal{L}_{\text{DALI}}$	Final loss function by DALI
$\mathcal{L}_{pref}$, $\mathcal{L}_{anc}$, $\mathcal{L}_{qual}$	Preference, prompt-anchor, and quality losses, respectively
λ_{anc}, λ_{qual}	Balancing hyperparameters
τ	Temperature hyperparameter
$\sigma(\cdot)$	Sigmoid function
$\Delta(\cdot)$	Preference margin
$s(\cdot)$	Conditional score

B CONSTRUCTION OF DALI-BENCH

We provide details on the construction of DALI-BENCH, including seven categories of under-specification, three releases, and additional implementation details.

B.1 CATEGORIES OF UNDER-SPECIFICATION

As discussed in Section 4.1, we consider seven categories of under-specification when constructing DALI-BENCH. In other words, we identify words corresponding to these seven visual properties in the full prompts of existing datasets, mask them to create open prompts, and treat them as prompts with the corresponding visual property under-specified. Five categories are object-level, while the remaining two categories mask scene-level features. Following the visual properties commonly considered in existing compositional T2I datasets (Bakr et al., 2023; Feng et al., 2023a; Ghosh et al., 2023; Huang et al., 2023), we select the following categories:

- **Color.** Color is one of the most widely adopted visual properties in prior work, as the color of each object is highly salient and can be easily quantified using color spaces such as RGB, CMYK, and CIE XYZ (Hyeon-Woo et al., 2025; Liang et al., 2025). Accordingly, we select the following fifteen representative colors for masking: `{red, orange, yellow, green, blue, purple, pink, brown, black, white, gold, golden, silver, gray (grey), beige}`.

- **Material.** Material captures the physical substance or composition of an object, and strongly influences the object's visual appearance, including reflectance, transparency, and surface quality. Accordingly, we select the following seventeen material descriptors for masking: {wooden, metallic, plastic, glass, rubber, fabric, leather, marble, ceramic, silk, wool (woolen), stone, brick, denim, velvet, cotton, steel}.
- **Size.** Size describes the relative scale or extent of an object, and is a common attribute for evaluating whether T2I models correctly bind comparative visual properties to the intended object among multiple objects. We select sixteen size descriptors as follows: {small, big, large, tiny, huge, tall, short, wide, narrow, thin, thick, long, giant, miniature, oversized, little}.
- **Texture.** Texture captures fine-grained surface appearance, and patterns, such as smoothness, roughness, glossiness, and repeated motifs, which are important for evaluating detailed attribute understanding. We select twenty-two texture descriptors for masking: {textured, striped, fluffy, smooth, rough, glossy, matte, furry, silky, plush, shiny, polished, rugged, velvety, crisp, bumpy, checkered, patterned, spotted, knitted, woven, sleek}.
- **Shape.** Shape characterizes the geometric form or outline of an object and is widely used to evaluate whether models preserve object-level structural attributes. These are the selected nineteen shape descriptors: {round, oval, circular, square, rectangular, triangular, diamond, spherical, spiral, cylindrical, curved, conical, hexagonal, octagonal, cone-shaped, star-shaped, heart-shaped, crescent-shaped, cube-shaped}.
- **Lighting.** Lighting describes scene-level illumination conditions that affect the appearance and atmosphere of the generated image. We mask the following illumination descriptors: {warm, soft, lit, overcast, overhead, sunlit, gentle, soft overhead, faint, ambient, atmospheric, bright artificial, dim, ambient, dynamic, cinematic, flickering, golden}.
- **Background.** Background describes the scene or environmental context surrounding the foreground entities. Each instance is obtained by masking a background description such as: {plain, white, stark white, bustling, clean, serves as the backdrop, softly blurred, against a backdrop of a clear sky}

B.2 Releases of DaLI-Bench

We construct three releases of DaLI-Bench (small, base, and multi) from a total of 9,577 unique source prompts, consisting of 553 prompts from GenEval (Ghosh et al., 2023), 7,959 prompts from T2I-CompBench (Huang et al., 2023), and 1,065 prompts from DPG-Bench (Hu et al., 2024). From these prompts, we obtain 8,154 raw candidates, 7,797 valid candidates, and 4,587 unique source-prompt pairs. For all releases, we split the selected instances into train, validation, and test sets with a ratio of 0.6 / 0.2 / 0.2.

- **Small.** The small release contains 350 selected instances. It is balanced across slot types, with 50 instances per each category. By construction, 50 instances come from GenEval, 200 come from T2I-CompBench, and 100 come from DPG-Bench: GenEval supplies the color-open portion, T2I-CompBench supplies the four object-level slot types, and DPG-Bench supplies the two scene-level slot types.
- **Base.** The base release contains 2,295 selected instances, each from a distinct source prompt. The slot distribution is 502 color-open, 672 material-open, 490 shape-open, 208 size-open, 208 texture-open, 133 lighting-open, and 82 background-open instances. The selected set contains 191 GenEval instances, 1,889 T2I-CompBench, and 215 DPG-Bench instances.
- **Multi.** The multi release contains 2,295 selected instances and allows multiple selected instances to originate from the same source prompt, drawing them from 2,116 distinct source prompts with at most two instances per prompt. The selected instances come from T2I-CompBench and DPG-Bench, contributing 2,080 and 215 instances respectively. The slot distribution is 398 color-open, 767 material-open, 499 shape-open, 208 size-open, 208 texture-open, 133 lighting-open, and 82 background-open instances.

B.3 CONSTRUCTION DETAILS

We detail the construction procedure of DALI-BENCH, including source prompts, open-prompt construction, candidate extraction, validation, scoring, and instance selection. In overall, DALI-BENCH converts fully specified prompts into open-ended prompts by masking one visual attribute while preserving the masked value as a evaluation target and controlling for slot balance and prompt provenance.

Source prompts. We construct DALI-BENCH from prompts collected from GenEval (Ghosh et al., 2023), T2I-CompBench (Huang et al., 2023), and DPG-Bench Hu et al. (2024). GenEval and T2I-CompBench provide the five object-level categories, whereas DPG-Bench supplies prompts for the two scene-level categories (lighting-open and background-open). We first normalize whitespace in each prompt and then de-duplicate prompts based on their lower-cased forms. For T2I-CompBench, we exclude `new_objects` from the source pool.

Open-prompt construction. Each instance is created by removing a visual attribute span from a fully specified prompt, or namely `full_prompt`. The resulting prompt is named as the `open_prompt`, while the removed attribute value is stored as the `"target_value"`. We also store the object or entity modified by the removed attribute as the `"target_entity"`. For example, given the full prompt `"a photo of a pink parking meter"`, removing `"pink"` yields the open prompt `"a photo of a parking meter"`, with `"target_value"` = `"pink"` and `"target_entity"` = `"parking meter"`. The resulting open-slot types are `color-open`, `material-open`, `size-open`, `texture-open`, `shape-open`, `lighting-open`, and `background-open` as described in Appendix B.1.

Candidate extraction. Candidates are generated from lexical heuristics rather than LLM-based rewriting. For each visual property, we define a list of attribute terms and search for matching spans in the full prompt. When a matched span is removed, we extract the target entity from the right context of the span. The entity phrase is truncated when a stop word or a new article is encountered, and we keep at most four words. In addition, each candidate is annotated with an `anchor_set`, which consists of remaining entities, relation words, and keywords in the open prompt. This anchor set may later be used to evaluate whether generated images preserve the non-masked content of the prompt.

Validation and scoring. We apply conservative validation to filter out malformed candidates. Specifically, we check whether the masked span is correctly aligned with the full prompt, whether the open prompt can be reconstructed from the masking operation, whether the open prompt is sufficiently informative, whether the masked value does not remain in the open prompt, and whether the anchor set is non-empty. We also filter out basic grammatical errors introduced by masking, such as `a/an` mismatches and cases where an article is followed by a plural or uncountable noun. Valid candidates are then scored based on slot-source subset alignment, preference for short and simple prompts, and penalties for punctuation-heavy or elliptical scene descriptions. For example, shape candidates extracted from shape-oriented source subsets receive a higher score.

Instance selection. We keep the best candidate for each source-prompt and slot-type pair. For the `mini` and `base` releases, we select at most one instance per source prompt, whereas the `multi` release allows multiple instances to be selected from the same source prompt. To avoid severe slot imbalance, we enforce a minimum slot ratio of $\lceil \frac{N}{5C} \rceil$, where N and C are the numbers of selected instances and categories, respectively. After satisfying these slot constraints, the remaining budget is filled with the highest-scoring candidates.

C TRAINING AND EVALUATION

We provide additional details on the training and evaluation procedures, along with the prompt templates used throughout the experiments.

C.1 TRAINING DETAILS OF DALI

We train DALI on the training split of DALI-BENCH by first constructing a reusable preference dataset and then optimizing a lightweight model-specific adapter while keeping the others frozen.

Candidate images. Candidate images are generated using only the *open prompt*. For each training prompt, we sample multiple candidate images: four for ELLA (Hu et al., 2024) and Bifrost-1 (Lin et al., 2025), and two for Janus (Wu et al., 2025) and Janus-Pro (Chen et al., 2025b). We then construct a pairwise preference dataset by comparing all candidate pairs generated from the same open prompt, randomizing the display order of Image A and Image B before judging. For pipelines whose upstream model ϕ can inspect images, preferences are labeled by ϕ itself. Only pairs with a strict preference label are used for optimization, while ties are excluded from the trainable subset.

When the upstream model ϕ is a text-only LLM and thus cannot directly compare images, we instead follow a rewrite-based construction similar to $\mathrm{OpenAcc}(\cdot)$. Specifically, we ask ϕ to commit to one slot value and rewrite the prompt into a completed prompt. We then generate an image from the completed prompt and treat it as the preferred sample over an image generated from the original prompt. This yields a deterministic positive/negative pair rather than an ϕ-judged pairwise ranking.

Conditional score. The conditional score $s_\Theta(\cdot)$ in $\mathcal{L}_{pref}(\cdot)$ measures how well an image agrees with the conditioning prompt under a given pipeline (see Equation (2)). During optimization, we evaluate the same scoring function for both the updated pipeline Θ' and the frozen reference pipeline Θ. For ELLA, the score is defined as the negative per-example denoising MSE, computed after sampling a timestep from the middle portion of the diffusion horizon. For Bifrost-1, the score is the negative per-example reconstruction MSE between predicted and target visual tokens under teacher-forced prompt conditioning. For Janus and Janus-Pro, the score is the negative log-likelihood of the generated image tokens under teacher forcing.

Trainable parameters. The trainable components differ by baseline: for ELLA (Hu et al., 2024), we update the connector module; for Bifrost-1 (Lin et al., 2025), we update only the bridging vision head; and for Janus models (Wu et al., 2025; Chen et al., 2025b), we update the generation-side modules.

C.2 Evaluation details

We provide the details on our automated and human evaluations.

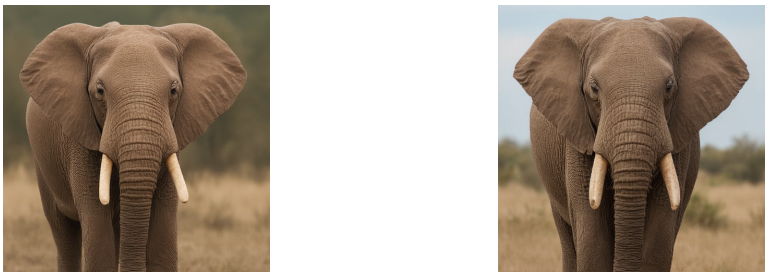
Intent win rate $\mathrm{IWR}(\cdot)$**.** Given a set of prompts and pairs of images generated by baseline Θ and adapted Θ' pipelines, we assess the win rate of Θ' over Θ following Definition 1. For fair evaluation, we introduce two design choices: order-symmetric judging and tie exclusion.

First, we perform order-symmetric judging to mitigate position bias in VLM-based evaluation. As noted in prior work (Tian et al., 2025; Wang et al., 2025), LLMs and VLMs struggle from position bias, where their judgments may change depending only on the order in which options are presented. To address this issue, we evaluate each image pair in both orders and aggregate the two judgments to determine the final outcome. If the two judgments agree after mapping Image A/B back to the corresponding pipelines, we adopt the agreed outcome. If they conflict, we select the judgment with the higher reported confidence; when the confidence scores are also tied, we mark the instance as a tie. For this reason, we ask the judge model to return not only its preference but also a confidence score and a short explanation, which improves interpretability of the automated evaluation.

Second, we exclude ties from the denominator when computing IWR. Although ties can be handled in several ways, we avoid forcing a preference when neither image is clearly better. Thus, if the judge or human annotator naturally identifies an image pair as a tie, we treat the instance as non-decisive and remove it from win-rate computation. In practice, both automated and human evaluations produce tie responses for about 20%–30% of the pairs (see Table 3). Reducing such ties by generating more preferable images remains an important objective for future work.

Open-slot commitment accuracy $\mathrm{OpenAcc}(\cdot)$**.** We evaluate open-slot commitment accuracy $\mathrm{OpenAcc}(\cdot)$ (see Definition 2) in the same manner as prompt-alignment accuracy, with one key difference: the prompt is the filled prompt by upstream model ϕ, rather than the original prompt.

Human evaluation. We conduct a small-scale human evaluation with fifty participants. We randomly select forty prompts from DALI-BENCH and compare images generated by a baseline Bifrost-1 (Lin et al., 2025) model, and its DALI-adapted variant. Figure 6 visualizes five example pairs for each prompt, where the baseline images are shown on the left and the adapted images on the right. For a fair comparison, we randomly shuffle the baseline and adapted images in the annotation interface and present them as Images A and B.



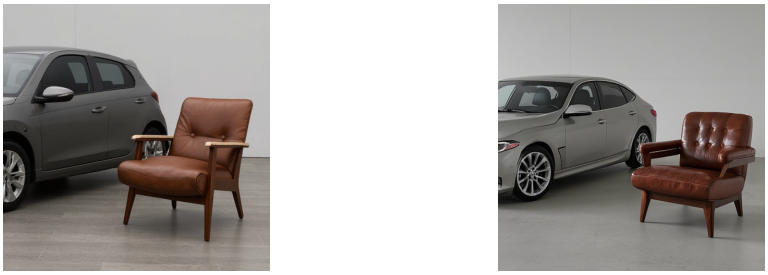
(a) Prompt 1: "a photo of an elephant"



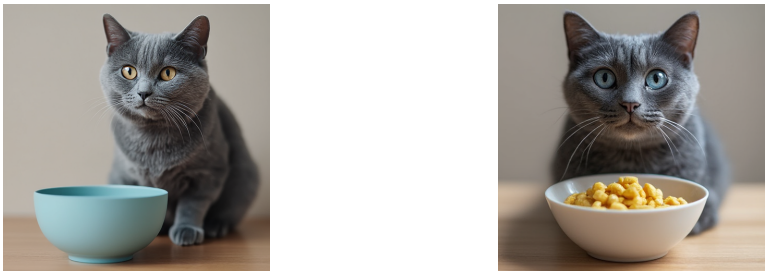
(b) Prompt 2: "a fabric dress and a belt"



(c) Prompt 3: "a fluffy sweater and a jar"



(d) Prompt 4: "a car and a leather chair"



(e) Prompt 5: "a bowl and a blue cat"

Figure 6: Five examples of image pairs adapted for human evaluation. For each prompt, human annotators compare the baseline (left) and adapted (right) images and select the one that better matches their visual intent. All pairs are randomly mixed in order to disregard to A/B preference.

C.3 PROMPTS

We adopt fixed prompt templates for automated and human evaluation. In all templates, the highlighted fields denote instance-specific variables that are replaced at evaluation time, such as the open prompt, slot type, and generated images. For automated evaluation, we require JSON-only responses with predefined keys to ensure consistent parsing. The prompt templates are shown below.

Intent win rate. For $\mathrm{IWR}(\cdot)$, a judge model $\mathcal{J}$ directly compares two images generated from the same open prompt. The judge is instructed to infer the latent visual intent only from the open prompt itself and the slot type. This protocol measures which image better matches the plausible intention induced by the under-specified prompt.

System:
You are the self judge for an open-prompt interpreter. Return JSON only.

User:
Compare image A and image B for the open prompt below.

Open prompt: {open_prompt}

Slot type: {slot_type}

Judge which image better matches the latent visual intent you would infer from the open prompt itself. Do not assume access to any revealed full prompt or post-hoc completed prompt. Return a JSON object with exactly these keys:

{ "preference": "a|b|tie", "confidence": 0.0, "notes": "<short note>" }

Image A is attached first. Image B is attached second.
{image_a} {image_b}

Open-slot commitment accuracy. Open-slot commitment accuracy $\mathrm{OpenAcc}(\cdot)$ measures whether a generated image is consistent with an explicit commitment made for an under-specified slot. We follow a two-step protocol that first elicits a concrete slot-level commitment and then verifies whether the generated image realizes that commitment. First, we perform self-intention elicitation. Given the open prompt, target entity, and slot type, the upstream model ϕ commits to one slot value and produces a completed prompt that reflects this commitment. This step converts the open-ended latent intent into an explicit textual commitment for subsequent evaluation.

System:
You are an interpreter for under-specified image prompts. Commit to one concrete visual intention and return JSON only.

User:
Read the open prompt and resolve the under-specified slot with one concrete choice.

Open prompt: {open_prompt}

Target entity: {target_entity}

Slot type: {slot_type} ({slot_label})

Choose one concise slot value, even if multiple values would be plausible. Also produce a natural completed prompt that reflects your chosen value.

Return a JSON object with exactly these keys:

{ "committed_value": "<short value>", "completed_prompt": "<natural completed prompt>", "confidence": 0.0, "notes": "<short note>" }

Second, we perform commitment-aware alignment judging. Given the committed value and completed prompt from the previous step, an external judge evaluates whether the generated image realizes the

commitment. The judge is asked to focus on the target entity and determine whether it exhibits the committed slot value. We use a three-way label, `yes`, `partial`, or `no`, to capture full, partial, and failed alignment, respectively.

System:
You are an external image-text alignment judge for text-to-image research. Return JSON only.

User:

{image}

The prompt below was completed by a separate interpreter from an under-specified open prompt.

Open prompt: {open_prompt}

Completed prompt: {completed_prompt}

Target entity: {target_entity}

Slot type: {slot_type} ({slot_label})

Committed value: {committed_value}

Judge whether the image aligns with the completed prompt, with special attention to whether the target entity realizes the committed slot value. Return a JSON object with exactly these keys:

{ "alignment_label": "yes | partial | no", "confidence": 0.0, "notes": "<short note>" }

Human evaluation instruction. For human evaluation, we follow a pairwise comparison protocol aligned with IWR. Annotators are shown only the open prompt and two generated images, and are asked to choose the image that better matches the latent visual intent inferred from the prompt itself. Each annotator answers 50 pairwise questions divided into two parts: 40 intent-preference questions followed by 10 prompt-independent visual-quality questions.

Instructions
You will compare 50 pairs of AI-generated images (5 pages, 10 pairs per page). For each pair, select **Image A**, **Image B**, or **Tie**.

Part 1 - Intent (Questions 1-40)
Compare image A and image B for the prompt below. Judge which image better matches the latent visual intent you would infer from the prompt itself. Image A is attached first. Image B is attached second.

Part 2 - Visual quality (Questions 41-50)
Compare image A and image B. Judge which image shows better overall visual quality, regardless of the prompt. Image A is attached first. Image B is attached second.

Notes
Use **Tie** sparingly - only when neither image is clearly preferable. **If the two images look identical, select Tie.** Please answer every question; you cannot move to the next page until all 10 questions on the current page are answered.

Each annotation item presents one open prompt and the corresponding pair of generated images. Annotators select one of three options: 'Image A', 'Image B', or 'Tie'. They may select a tie when neither image is clearly preferred.

Prompt {i} : " {prompt} "

{image_a} {image_b}

☐ Image A ☐ Image B ☐ Tie

D ALGORITHM

We formulate the overall algorithm of DALI in Algorithm 1.

Algorithm 1 DALI

Input: Pre-trained T2I generation pipeline $\Theta = \{\phi, \beta, \mathcal{G}\}$ consisting of upstream model ϕ, bridge modules β, and generator $\mathcal{G}$, the number of $epochs$, calibration dataset $\mathbb{D}_{calib}$, hyperparameters λ_{anc}, λ_{qual}, and τ.
Output: Adapted T2I generation pipeline $\Theta' = \{\phi', \beta', \mathcal{G}'\}$.

Freeze the upstream model ϕ and generator $\mathcal{G}$.
Initialize the adapted pipeline $\Theta' \leftarrow \Theta$.

// T2I pipeline adaptation
for $t \leftarrow 1$ to $epochs$ **do**
  for each mini-batch $\mathbb{B} \subset \mathbb{D}_{calib}$ **do**
    Extract reference $\{\mathbf{c}_i\}_{i\in\mathbb{B}}$ and adapted conditions $\{\mathbf{c}'_i\}_{i\in\mathbb{B}}$ from β and β', respectively.
    Compute the preference loss $\mathcal{L}_{pref}$ (Equation (2)).
      ▷ Idea 1: pairwise preference optimization
    Compute the prompt-anchor loss $\mathcal{L}_{anc}$ and the quality loss $\mathcal{L}_{qual}$ (Equations (3) and (4)).
      ▷ Idea 2: prompt fidelity and quality regularization
    Form the final loss function $\mathcal{L}_{\text{DALI}} \leftarrow \mathcal{L}_{pref} + \lambda_{anc}\mathcal{L}_{anc} + \lambda_{qual}\mathcal{L}_{qual}$ (Equation (1)).
    Update only the adapted bridge modules β' by minimizing $\mathcal{L}_{\text{DALI}}$
      ▷ Idea 3: selective bridge adaptation
  end for
end for
return Θ'

E EXPERIMENTAL SETUP

We provide details for the general experimental setup, including datasets, baselines, hyperparameters, and implementation details. Refer to Appendix C for DALI-specific settings.

E.1 DATASET AND BASELINE MODELS

We evaluate DALI and competing methods primarily on DALI-BENCH. For the image quality experiment, we compute FID (Heusel et al., 2017) on the COCO (Lin et al., 2014) dataset, following the COCO-30K FID protocol. We use Qwen2.5-VL-7B-Instruct (Bai et al., 2025) as the judge for prompt alignment.

We briefly summarize the baseline models below. To cover the four pipeline categories described in Section 2 and Figure 2, we evaluate two representative pipelines for each category. Table 1 summarizes the upstream model ϕ and image generator $\mathcal{G}$ used in each pipeline.

- **GLIGEN** (Li et al., 2023) is a grounded T2I diffusion model that conditions generation on both a text prompt and grounding inputs such as bounding boxes, serving as a layout-guided generation baseline.
- **Stable Diffusion XL (SDXL)** (Podell et al., 2024) is a strong text-only latent diffusion model and serves as a standard end-to-end T2I baseline without explicit upstream planning or grounding.
- **VPGen** (Cho et al., 2023) is an interpretable planning-based baseline based on visual programming that decomposes generation into object, count prediction, layout generation, and image synthesis.
- **RPG** (Yang et al., 2024) is a training-free recaptioning-and-planning framework that adopts a multi-modal LLM (MLLM) to decompose complex prompts into regional generation plans and synthesizes images through regional diffusion.

Table 7: Hyperparameter ranges for DALI.

Hyperparameter	Range
λ_{anc}	[0, 0.05, 0.1, 0.5, 1, 5, 10, 50]
λ_{qual}	[0, 0.5, 1, 5, 10, 50, 100, 500]
τ	[500, 1000, 2000]
lr	[1e-4, 1e-5]
$epochs$	[10, 20, 30, 40, 50, 100]

Table 8: 95% confidence intervals for the property-validation results in Table 1. Accuracy changes are reported in percentage points (%p), and CLIP-IQA+ is multiplied by 100 for readability.

Pipeline	$\mathrm{IWR}_\phi(\cdot)$	Δ OpenAcc$(\cdot)$	Δ Prompt	-Δ FID	Δ MUSIQ	Δ CLIP-IQA+
ELLA	-	$+3.1$ [$+1.2$, $+5.0$]	-0.4 [-1.4, $+0.6$]	-0.7 [-2.4, $+1.0$]	$+0.5$ [$+0.2$, $+0.8$]	$+0.6$ [$+0.3$, $+0.9$]
Bifrost-1	55.8 [53.5, 58.2]	$+3.1$ [$+0.8$, $+5.4$]	$+1.0$ [-0.2, $+2.2$]	-1.7 [-3.5, $+0.1$]	$+0.6$ [$+0.3$, $+0.9$]	$+0.5$ [$+0.1$, $+0.9$]
Janus	54.4 [51.5, 57.3]	$+4.6$ [$+2.1$, $+7.1$]	-0.4 [-1.1, $+0.3$]	-0.6 [-1.5, $+0.3$]	$+0.8$ [$+0.4$, $+1.2$]	$+0.2$ [-0.3, $+0.7$]
Janus-Pro	57.2 [55.4, 58.0]	$+3.8$ [$+1.4$, $+5.2$]	-0.6 [-1.4, $+0.2$]	-1.1 [-2.5, $+0.3$]	$+0.4$ [-0.2, $+1.0$]	$+1.3$ [$+0.7$, $+1.9$]

- **ELLA** (Hu et al., 2024) equips diffusion models with an LLM through an efficient semantic adapter, improving dense prompt following while keeping the diffusion backbone and LLM frozen.
- **Bifrost-1** (Lin et al., 2025) bridges a pre-trained MLLM and a diffusion model using patch-level CLIP latents, allowing the MLLM to provide visual generation priors for diffusion-based synthesis.
- **Janus** (Wu et al., 2025) is an autoregressive generator that decouples visual encoding for understanding and generation while sharing a unified transformer backbone.
- **Janus-Pro** (Chen et al., 2025b) is an improved version of Janus with an optimized training strategy, expanded training data, and larger model scale, serving as a stronger unified generation baseline.

E.2 Hyperparameters

We conduct a grid search to validate hyperparameters and select the set with the best performance. Table 7 reports the searched hyperparameter ranges of DALI. Note that lr and $epochs$ denote the learning rate and number of calibration iterations, respectively. For baselines, we search within the range described in each paper.

E.3 Implementation Details and Machine

We follow the settings from ELLA (Hu et al., 2024) and Bifrost-1 (Lin et al., 2025) for fair comparison. The size S of the calibration dataset $\mathbb{D}_{calib}$ is 1,024, with a batch size of 32. We adopt the AdamW (Loshchilov & Hutter, 2019) optimizer for the update of DALI and decay the learning rate by 10% every eight steps. We implement DALI with the PyTorch and Transformers (Wolf et al., 2020) libraries in Python. For the other methods, we reproduce them based on their open-source code. For the efficiency evaluation, we conduct a 100-epoch run for each method and report the average per-epoch runtime and peak GPU memory usage. All experiments are done at a workstation with Intel Xeon Gold 6338 and NVIDIA A100 80GB.

F Extended Experiments

We provide extended analyses of DALI along the dimensions raised by the main evaluation: confidence intervals, joint compression with prompt completion, slot-type analysis, hyperparameter sensitivity, and human-marked quality.

Table 9: Cross-IWR $\mathrm{IWR}_{\mathcal{J}}(\cdot)$ with 95% CI under six model judges.

Judge $\mathcal{J}$	$\mathrm{IWR}_{\mathcal{J}}(\cdot)$	95% CI
Qwen2.5-VL-7B	55.8	[54.0, 57.5]
InternVL-3.5-4B	54.1	[51.3, 56.9]
Qwen2.5-VL-32B	56.9	[54.8, 59.0]
Qwen3-VL-32B	61.1	[58.0, 64.4]
LLaVA-OV2-8B	58.8	[56.1, 61.5]
GPT-5.6 Terra	55.7	[53.2, 58.2]

Table 10: Commitment-value consistency and pairwise preference-label agreement with 95% confidence intervals.

Perturbation	Commitment [%] [95% CI]		Preference [%] [95% CI]
	Strict	Semantic	
Repeated run	81.8 [80.8, 82.8]	93.4 [92.8, 94.0]	98.5 [98.0, 99.0]
Prompt paraphrase	75.1 [73.8, 76.4]	86.7 [85.8, 87.6]	96.1 [95.4, 96.8]
Cross-model	48.0 [46.8, 49.2]	60.7 [58.8, 62.6]	74.8 [72.5, 77.1]

Table 11: Effects of prompt completion on open-slot commitment accuracy and prompt alignment [%]. Differences compare completed prompts with the corresponding original prompts, with paired 95% confidence intervals.

Method	Prompt	$\mathrm{OpenAcc}(\cdot)$	$\Delta\,\mathrm{OpenAcc}(\cdot)$ [95% CI]	Prompt Acc.	Δ Prompt Acc. [95% CI]
FLUX.1	Original	42.7	+3.3 [2.1, 4.5]	86.6	+2.0 [0.4, 3.6]
	Completed	46.0		88.6	
Bifrost-1	Original	45.5	+2.3 [1.0, 3.6]	89.3	+1.0 [-0.3, 2.3]
	Completed	47.8		90.3	
Bifrost-1 + DALI	Original	48.6	+1.3 [0.3, 2.3]	91.3	+0.4 [-0.7, 1.5]
	Completed	49.9		91.7	

F.1 Confidence Intervals of Reported Results

We compute 95% confidence intervals to assess statistical uncertainty. Tables 8, 9, and 10 report the uncertainty associated with the property-validation, multi-judge, and output-consistency results. Overall, the confidence intervals support our principal findings, demonstrating consistent LIT improvements, robustness across judges, and reproducible output consistency.

F.2 Complementary Axis of Prompt Completion

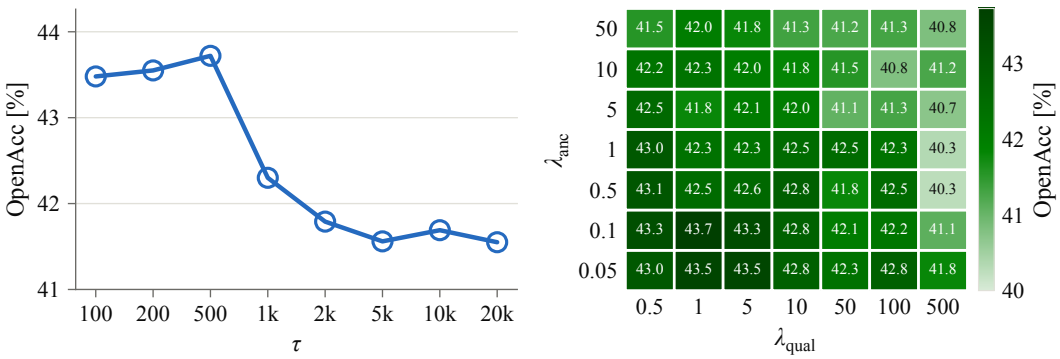
We evaluate prompt completion as a complementary axis that intervenes before generation rather than adapting the upstream-to-generator interface.

Prompt completion. Prompt completion converts an under-specified prompt into a more explicit textual specification before image generation (Hao et al., 2023; Datta et al., 2024; Wu et al., 2026). Related approaches enrich the generator input through prompt rewriting (Cao et al., 2023; Yang et al., 2025), recaptioning (Yang et al., 2024; Chatterjee et al., 2024), dense captions (Kim et al., 2023; Lai et al., 2025), or structured representations (Feng et al., 2023b;c). In our setting, prompt completion mitigates user-to-upstream miscommunication at the first stage, whereas DALI addresses upstream-to-generator misrealization at the second stage. We evaluate the independent effect of this explicit intervention and its joint effect with the bridge-level adaptation of DALI.

Comparison and adaptation with DALI. Table 11 compares original and completed prompts for FLUX.1, Bifrost-1, and Bifrost-1 adapted with DALI. We generate one completed prompt from each open prompt using Qwen2.5-VL-7B-Instruct in a text-only setting and evaluate all conditions under the same image-generation and evaluation protocols. We have three observations from the results. First, prompt completion consistently improves open-slot commitment accuracy, indicating that explicitly verbalizing a missing visual detail resolves part of the upstream-to-generator transfer gap without modifying the generator. Second, DALI outperforms prompt completion alone, indicating that second-stage bridge adaptation addresses transfer failures that first-stage completion does not fully resolve. Lastly, the strongest result arises when completion is combined with DALI, suggesting that explicit textual specification and bridge-level adaptation address complementary sources of transfer failure.

F.3 Hyperparameter Analysis

Figure 7 reports open-slot commitment accuracy $\mathrm{OpenAcc}(\cdot)$ from representative operating points of the ELLA hyperparameter sweep. We have two observations from the results. First, DALI maintains stable performance across a broad range of hyperparameter settings. Second, excessively large values degrade commitment accuracy, highlighting the importance of selecting moderate operating points.



(a) Temperature

(b) Balancing hyperparameters λ_{anc} and λ_{qual}

Figure 7: Hyperparameter analysis on (a) temperature τ and (b) balancing hyperparameters λ_{anc} and λ_{qual}. See Appendix F.3 for details.

Table 12: Open-slot commitment accuracy $\mathrm{OpenAcc}(\cdot)$ of Bifrost-1 across five object-level and two scene-level categories.

Method	Color	Material	Shape	Size	Texture	Lighting	Background
Bifrost-1	50.5	52.2	36.7	39.0	39.0	46.2	37.5
+ DALI	52.2	55.2	40.8	40.7	41.9	57.7	43.8

F.4 Category-wise Improvement

Table 12 analyzes open-slot commitment accuracy across different open-slot types in DALI-BENCH. The results on ELLA (Hu et al., 2024) reveal two observations. First, DALI improves LIT across multiple under-specification categories, showing that the adaptation is not limited to a single visual property. Second, the improvement varies across properties, with lighting showing the largest gain of 11.5%p. This suggests that the difficulty of LIT depends on the type of visual property.

F.5 Human-marked Image Quality

We complement the automated quality metrics with a blind human comparison of the overall visual quality of Bifrost-1 and its DALI-adapted counterpart. Fifty participants each evaluate 10 matched image pairs with randomized image order and an explicit tie option, yielding 500 pairwise judgments. The resulting preference rates for Bifrost-1, tie, and DALI are 36.8%, 25.6%, and 37.6%, respectively. The nearly balanced non-tie preferences and substantial tie rate suggest that adaptation with DALI does not lead to a noticeable degradation in perceived visual quality.

G Further Discussion

We conduct a case study on generated images and offer further discussion and remarks on our work.

G.1 Visualization and Case Study: Gray Area of Image Preference

Latent intent raises a subtle question: when an open prompt leaves a visual property under-specified, how freely should a generative model complete the missing visual details? We examine this question through two gray-area cases: contextual background construction and the addition of plausible but unmentioned objects. These cases illustrate that intent following is not always reducible to strict lexical prompt matching, and that acceptable visual completion may depend on user expectation, prompt context, and the salience of the added visual content.

Background effect. A first gray area concerns the background or scene context. Consider an open prompt such as `a photo of a soccer player`. Figure 8 illustrates some possible outputs generated by Bifrost-1 (Lin et al., 2025). The prompt specifies the subject but leaves the surrounding



Figure 8: Background ambiguity under the prompt "`a photo of a soccer player`". Multiple backgrounds are plausible, such as a plain scene (left), a training facility (middle), and a crowded stadium (right). While none of them contradicts the prompt, which background should be preferred?



Figure 9: Object-addition ambiguity under the prompt "`a dog and a chair`". Adding a small water bowl may be perceived as plausible contextual enrichment, but additional salient objects can also be viewed as undesired deviations from the specified prompt.

scene open, leading to outputs with a plain background, a training facility, or a stadium. None of these choices directly contradicts the prompt, but they convey different latent intentions. In practice, such background choices are often accepted by users as part of the model's visual discretion, especially when the background is contextually compatible with the main subject and does not alter the requested object. This suggests that background completion should not always be treated as hallucination or prompt violation; rather, it occupies a soft region between faithful realization and creative elaboration.

Addition of new objects. A second gray area concerns the addition of new objects. For example, Figure 9 shows possible outputs generated by Bifrost-1 (Lin et al., 2025) for the prompt "`a dog and a chair`". While some outputs contain only the specified dog and chair, others introduce additional objects such as a water bowl or multiple surrounding props. Such an image can be preferred by a judge model because the water bowl is a plausible contextual object for a dog scene. However, this behavior is more controversial than background selection: the prompt explicitly specifies two objects, and adding objects may be perceived as either helpful contextual enrichment or an unwanted deviation from the prompt. The acceptability of such additions therefore depends on the role and salience of the new object. A small supporting prop may be acceptable in some contexts, while a newly introduced foreground object can change the semantic content of the image. This boundary is continuous rather than discrete; for instance, a crowd in a stadium may consist of many individual people, but it is often perceived as background once it becomes sufficiently collective and non-salient.

Scope of acceptable visual discretion. We do not attempt to define a universal rule for acceptable visual discretion. Instead, these examples highlight the need to distinguish several levels of completion. At one end, choosing an unspecified attribute of an explicitly mentioned entity, such as the color of a dog or the material of a chair, is a natural form of latent-intent resolution. Contextual background choices are also often acceptable when they support the requested scene. In contrast, adding new salient objects, changing object counts, or introducing objects that compete with the prompt should be treated more cautiously. A useful evaluation protocol should therefore account not only for whether an

image contains the words in the prompt, but also for whether additional visual content is compatible with the inferred intent and remains within the user's likely tolerance. This motivates future work on more fine-grained taxonomies of visual discretion, separating attribute completion, background completion, supporting props, salient object additions, and semantic contradictions.

G.2 Additional Remarks

We discuss additional considerations behind our design, summarize the current limitations of LIT and DALI, and present possible future extensions.

Drawbacks of self-evaluation. Self-evaluation with LLMs or VLMs provides a scalable way to approximate human preferences, but it may also introduce evaluation artifacts. Prior studies have shown that LLM-as-a-judge protocols may suffer from limitations such as position bias, verbosity bias, self-enhancement bias, self-preference bias, and limited reasoning ability (Liu et al., 2023; Panickssery et al., 2024; Tian et al., 2025; Zheng et al., 2023). In particular, when the same model family is involved in both generation and evaluation, the evaluator may prefer outputs that resemble its own generations or internal style, even when human annotators do not show the same preference (Panickssery et al., 2024). Moreover, intrinsic self-correction without external feedback is not always reliable and may even lead to performance degradation in heavily reasoning settings (Huang et al., 2024).

Our evaluation is designed to reduce these concerns. First, for $\mathrm{IWR}(\cdot)$, we conduct pairwise comparison under the same open prompt and apply order-symmetric judging by evaluating both A/B orders. This mitigates position bias and makes the comparison depend less on presentation order. Second, open-slot commitment accuracy $\mathrm{OpenAcc}(\cdot)$ separates commitment elicitation from image-text alignment judging. The upstream model ϕ commits to a concrete slot value before seeing the generated image, while a separate judge evaluates whether the image realizes the resulting completed prompt. Thus, the model is not directly asked to grade its own generated image. Third, we allow ties and exclude them from the win-rate denominator instead of forcing an artificial preference. Finally, we complement automated evaluation with human evaluation, where the image order is randomly shuffled. Therefore, while automated self-evaluation remains imperfect, we employ it as a controlled comparative diagnostic rather than as an absolute measure of visual quality.

Choice of under-specified features. Throughout the paper, we argue that T2I pipelines should better reflect the latent intent of the upstream model ϕ. This raises a natural question: where does such latent intent come from? In our setting, latent intent is not treated as a hidden ground-truth attribute. Rather, it is an operational commitment induced by the prompt context, the pre-trained priors of ϕ, and common visual conventions learned from data. For example, when the prompt leaves the color of an object under-specified, ϕ may prefer a color that is typical for the object or contextually coherent with the rest of the scene.

This perspective also reveals an important risk. If latent intent is dominated by biased or overly concentrated model priors, adapting a generator to follow it may amplify those priors and reduce output diversity. This concern is related to prior observations that T2I models may reproduce or amplify social and visual biases under neutral or under-specified prompts (Seshadri et al., 2024). Therefore, LIT should not be interpreted as enforcing a single universally correct completion. Instead, it provides an optimization framework for studying whether a generator follows the upstream model's explicit or implicit commitment when such a commitment is made.

User intent and upstream commitment. The main goal of T2I generation is to draw an image which perfectly follows the user's intent. However, the user's intent is generally not identifiable from the prompt alone as discussed in Section 3.1. We consequently formulate LIT conditionally on a fixed upstream interpreter rather than treating its commitment (named as *latent intent* in this paper) as a universally correct completion or as a proxy of user's intent. Improving user-to-upstream alignment and improving upstream-to-generator faithfulness remain complementary objectives.

In other words, LIT is not intended to infer or approximate the user's unobserved intent from an under-specified prompt. Instead, it evaluates a conditional internal-consistency property: given that a designated upstream model has formed a visual commitment, does the downstream generator faithfully realize that commitment? Whether the commitment itself matches the user's preference is a separate user-to-upstream alignment problem.

Diversity and bias amplification. A concentrated or biased upstream prior may lead DALI to favor a narrow set of visual completions. Better inter-stage faithfulness may therefore amplify upstream biases and reduce output diversity. Future evaluation should measure commitment fidelity and diversity jointly, and extend LIT to stochastic commitments so that faithfulness can be improved without collapsing prompt-valid diversity.

Limitations and future work. The current study has several limitations. First, DALI updates only lightweight model-specific components while keeping the main generator frozen. This design is efficient and enables fair adaptation across heterogeneous pipelines, but fine-tuning only a small subset of parameters may introduce a trade-off between improving LIT and preserving prompt alignment or image quality. For example, stronger alignment to inferred latent intent may sometimes come at the cost of reduced diversity or degraded visual fidelity. A natural extension is to investigate full-model fine-tuning and even from-scratch training strategies that jointly improve the desired properties defined in Section 4.2.

Second, the current scope of DALI-BENCH is limited to seven visual properties: color, material, size, texture, shape, lighting, and background. These categories cover common object-level under-specification, but real image prompts can also be under-specified in many other properties, such as style, viewpoint, pose, count, spatial relation, and scene-level composition. Extending the benchmark to these additional slots would allow a more comprehensive analysis of LIT in T2I generation.

Third, our prompts are relatively concise because they are derived from existing compositional T2I benchmarks. Longer prompts may contain multiple objects, multiple under-specified slots, or competing visual constraints, making the latent intent harder to resolve. Future work will cover long-form prompts with multiple commitments, hierarchical slot resolution, or multi-turn protocols.

Beyond T2I generation, the same principle can be extended to other generative outputs such as video and audio, where an under-specified prompt may imply latent choices about motion, temporal events, sound texture, or acoustic environment. Another promising direction is to adopt synthetic preference data such as DALI-BENCH for model compression. Open-slot preference data may provide calibration signals for compressing generative pipelines while preserving intent-following behavior.