Rhetorical Text-to-Image Generation via Two-layer Diffusion Policy Optimization

Anonymous ACL submission

Abstract

Generating images from rhetorical languages remains a critical challenge for text-to-image models. Even state-of-the-art (SOTA) multimodal large language models (MLLM) fail to generate images based on the hidden meaning inherent in rhetorical language—despite such content being readily mappable to visual representations by humans. A key limitation is that current models emphasize object-level word embedding alignment, causing rhetorical expressions to steer image generation towards their literal visuals and overlook the intended semantic meaning. To address this, we propose Rhet2Pix, a framework that formulates rhetorical text-to-image generation as a multi-step policy optimization problem, incorporating a two-layer MDP diffusion module. In the outer layer, Rhet2Pix converts the input prompt into incrementally elaborated sub-sentences and executes corresponding image-generation actions, constructing semantically richer visuals. In the inner layer, Rhet2Pix mitigates reward sparsity during image generation by discounting the final reward and optimizing every adjacent action pair along the diffusion denoising trajectory. Extensive experiments demonstrate the effectiveness of Rhet2Pix in rhetorical text-to-image generation. Our model outperforms SOTA MLLMs such as GPT-5.1, Grok-3, and leading academic baselines, across both qualitative and quantitative evaluations. The code and dataset used in this work are publicly available at: https://anonymous.4open.science/r/Rhet2Pix-2D52/.

1 Introduction

Rhetorical language infuses text with vivid sensory and imaginative depth, and visualizing linguistic metaphors amplifies multimodal understanding while deepening engagement with rhetorical expression (Chakrabarty et al., 2023). Advanced text-to-image models offer a promising approach for this visualization task. Models such as GPT-5.1 (OpenAI et al., 2024), DALL-E 3 (Betker et al., 2023), Stable Diffusion (Rombach et al., 2022b), and Imagen (Saharia et al., 2022) are capable of generating highly realistic images from textual input. However, these models often prioritize general-purpose generation and may struggle with capturing the abstract and figurative nature of rhetorical expressions. Rhetorical language visualization, framed as a task-specific text-to-image generation problem, can benefit from reinforcement learning (RL), which is increasingly used to fine-tune diffusion models for specialized generative objectives (Black et al., 2024; Fan et al., 2023; Hu et al., 2025; Lee et al., 2023a; Prabhudesai et al., 2024; Wallace et al., 2023; Xu et al., 2023). Recasting the denoising process in diffusion models as a Markov Decision Process (MDP) enables the application of RL and policy optimization to achieve task-specific generative objectives. (Kaelbling et al., 1996; Rombach et al., 2022b; Wang et al., 2023).

Despite recent progress, generating faithful visualizations of rhetorical expressions remains a significant challenge. Existing text-to-image models typically encode input text as a guiding condition for image generation, treating individual words as discrete visual components. While this approach enables the image generation of recognizable objects and attributes, it results in fragmented visuals that fails to capture the compositional semantics and figurative structures essential for rhetorical meaning. Even models with strong language capabilities, such as GPT-5.1, are limited by this overly strict alignment between text and image, often producing unnatural fusions of subject and metaphorical vehicle that distort the intended interpretation.

To build a high-quality text-to-image model for rhetorical language, we propose Rhet2Pix. Inspired by the incremental process by which a painter develops a scene, Rhet2Pix models rhetorical image generation as a multi-step, two-layer Markov Decision Process (MDP), as illustrated in Figure 1. In

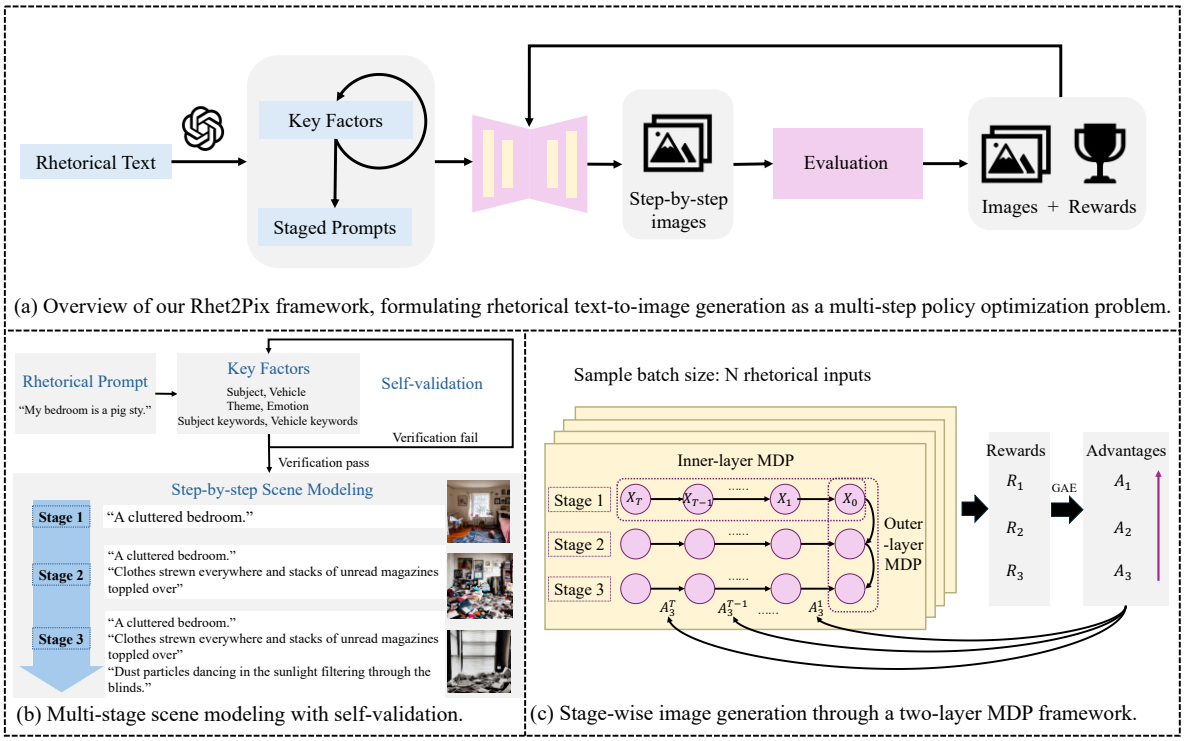


Figure 1: **Rhet2Pix framework.** Our approach formulates rhetorical image generation as a two-layer Markov Decision Process.

the outer layer, the input rhetorical prompt is progressively decomposed into a sequence of semantically enriched sub-prompts, guiding step-wise image generation that incrementally refines the visual scene. In the inner layer, Rhet2Pix models the denoising process of the diffusion model as a trajectory of actions and addresses the sparse reward problem by discounting the final reward along this trajectory, enabling fine-grained policy optimization over consecutive action pairs. To facilitate learning across both layers, we introduce a rhetorical-specific reward function that jointly evaluates semantic alignment and visual element consistency.

To summarize, we make the following key contributions:

- To the best of our knowledge, our work is the first to identify the misalignment issue in rhetorical text-to-image generation and trace its root cause: the word embedding of the vehicle object introduces a bias that leads the model to generate its literal visual counterpart. This creates a counter-effect — stronger object-level correspondence results in images that deviate further from the intended semantic meaning. This issue highlights a fundamental limitation in current MLLMs, which predominantly depend on surface-level text–image similarity.
- Our Rhet2Pix framework formulates rhetorical image generation as a multi-step policy optimization problem based on a two-layer Markov Decision Process (MDP), progressively generating images that reflect the intended rhetorical meaning. In the outer layer, it incrementally converts the input into semantically enriched prompts to guide progressive visual refinement, while in the inner layer, it optimizes adjacent denoising steps in the diffusion process using discounted rewards to address the challenge of reward sparsity.
- In the rhetorical image generation task, Rhet2Pix outperforms state-of-the-art academic baselines and leading MLLMs such as GPT-5.1 and Grok-3, demonstrating superior performance in both qualitative and quantitative evaluations.

2 Related work

Rhetoric research. Current research on rhetorical language mainly focuses on natural language processing tasks (Chakrabarty et al., 2021; Veale, 2016; Chakrabarty et al., 2022a), particularly the detection and interpretation of metaphors (Gong et al., 2020; Leong et al., 2018; Tsvetkov et al., 2014). Studies that bridge rhetorical text and image generation are limited. Akula et al. (2023) introduced the MetaCLUE framework for identifying metaphorical concepts in images, while

Chakrabarty et al. (2023) explored the integration of large language models with diffusion models for metaphor understanding. Despite these advances, a clear gap remains in generating images that accurately reflect the underlying semantics of rhetorical input. Our method addresses this challenge by generating images that align with the intended rhetorical meaning and accurately represent key semantic elements.

Automated prompt construction. Automatically transforming user input into effective prompts is essential for building high-performing and user-friendly text-to-image generation systems (Xie et al., 2023). Traditional approaches—ranging from hand-crafted templates (Dai et al., 2021; Petroni et al., 2019), to interactive refinement tools (Brade et al., 2023; Feng et al., 2023; Liu and Chilton, 2022), and prompt optimization via reinforcement learning or genetic algorithms (Gao et al., 2023; Hao et al., 2023; Mo et al., 2024)—rely heavily on expert intervention and iterative tuning, thereby limiting scalability and hindering full automation. Recent advancements in large language models (LLMs) have enabled autonomous semantic decomposition (Yang et al., 2024), task planning (Dai et al., 2024; Rana et al., 2023), and structured scene layout (Prasad et al., 2024) in multimodal tasks, opening new possibilities for more efficient text-to-image prompt construction. Building on these developments, our method leverages LLMs to automatically convert rhetorical inputs into a sequence of staged prompts with progressively enriched semantic content, providing structured guidance for stepwise image generation.

Text-to-image generation. In recent years, diffusion models (Ho et al., 2020; Sohl-Dickstein et al., 2015) and autoregressive models (Tian et al., 2024; Jiang et al., 2025; Sun et al., 2024) have gained significant attention for their outstanding performance in text-to-image generation. Compared to autoregressive models, diffusion models excel in generating high-quality and diverse images (Batzolis et al., 2021; Ho et al., 2022; Ramesh et al., 2021; Rombach et al., 2022a). By refining random noise through multiple denoising steps, diffusion models produce more coherent images that better align with the input text (Black et al., 2024; Hu et al., 2025). This method has quickly become the dominant framework for text-to-image generation (Betker et al., 2023; Saharia et al., 2022; Zhang et al., 2023; Yang et al., 2025).

Reinforcement Learning for Text-to-Image Generation. Despite advancements in diffusion models, aligning generated images with the complex semantics of input text remains challenging. Several studies have framed the denoising process of diffusion models as a reinforcement learning (RL) problem (Black et al., 2024; Fan et al., 2023; Hu et al., 2025; Lee et al., 2023a; Prabhudesai et al., 2024; Wallace et al., 2023; Xu et al., 2023), enabling fine-tuning to better align outputs with human preferences (Lee et al., 2023b; Zhang et al., 2024) and semantic intent (Dong et al., 2023; Huang et al., 2023). A key issue with RL-based diffusion models is reward sparsity, where feedback is only given after the full image is generated, leading to high gradient variance and slow convergence (Arriola et al., 2025). Recent solutions, including progressive backward training and branch-based sampling, have aimed to address this (Hu et al., 2025). Our approach further reduces reward sparsity by discounting the final reward across the diffusion trajectory and optimizing each adjacent action pair.

3 Preliminaries

Text-to-image diffusion models. The diffusion process consists of two stages: the forward process and the reverse process. In the forward process, the image x_0 is gradually corrupted into pure noise x_T over T steps, by adding Gaussian noise at each step. The reverse process generates an image from pure noise by iteratively denoising, conditioned on a textual description $\mathbf{c}$. It is parameterized by a neural network $\boldsymbol{\mu}_\theta(\mathbf{x}_t, \mathbf{c}, t)$, predicting the noise ϵ that transforms x_0 to x_T. Sampling starts from $x_T \sim N(0, I)$ and generates denoised samples iteratively (Ho et al., 2020; Song et al., 2020).

$$p_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{c}) = \mathcal{N}\left(\mathbf{x}_{t-1} \mid \boldsymbol{\mu}_\theta(\mathbf{x}_t, \mathbf{c}, t), \sigma_t^2 \mathbf{I}\right),$$

where μ_θ is predicted by a diffusion model parameterized by θ, and σ_t is the fixed timestep-dependent variance.

Markov decision process and reinforcement learning. The denoising process in diffusion models can be viewed as a Markov Decision Process (MDP), formulated as a sequential decision-making problem. We define an MDP $(\mathcal{S}, \mathcal{A}, P_0, P, R)$ with states $s \in \mathcal{S}$, actions $a \in \mathcal{A}$, initial state distribution P_0, transition probabilities P, and reward R. At each timestep t, the agent (e.g., diffusion model) observes state $s_t \in \mathcal{S}$, selects action $a_t \sim \pi_\theta(a_t \mid s_t) \in \mathcal{A}$, transitions to the next state $s_{t+1} \sim P(s_{t+1} \mid s_t, a_t)$, and receives reward

$R(s_t, a_t)$. As the agent interacts with the MDP, it generates a trajectory: a sequence of states and actions $\tau = (s_0, a_0, s_1, a_1, \ldots, s_T, a_T)$. The goal of Reinforcement Learning (RL) is to maximize expected cumulative reward from trajectories sampled according to its policy.

$$\mathcal{J}_{\mathrm{RL}}(\pi_\theta) = \mathbb{E}_{\tau \sim p(\tau|\pi_\theta)} \left[\sum_{t=0}^{T} R(s_t, a_t) \right]$$

where the policy π_θ parameterized by θ defines the action selection strategy and θ is updated by gradient descent.

4 Rhet2Pix framework

In this section, we introduce the Rhet2Pix framework, which automatically and efficiently generates high-quality images from rhetorical text. It captures the underlying semantics and expressive intent of the input without human intervention.

4.1 Problem statement

Given a dataset $\{T_1, T_2, \ldots, T_n\}$ of rhetorical texts, the objective is to generate a corresponding set of images $\{I_1, I_2, \ldots, I_n\}$ that accurately reflect the semantic meanings of each T_i without manual intervention. This task presents two key challenges:

- **Semantic-Level Alignment.** Models often collapse distinct ideas or oversimplify complex rhetorical expressions into literal interpretations, resulting in visuals that overlook deeper contextual meanings.
- **Object and Image-Level Alignment.** The alignment of semantic meaning sometimes causes the destruction of the original images. Models must balance meaning preservation with aesthetic integrity.

4.2 Multi-stage Scene Modeling

Given each rhetorical text T_i, we decompose it into a compact set of key factors F_i and derive a sequence of detailed scene instruction $\{P_i^j\}_{j=1}^{C}$, where C denotes the total number of stages (i.e., the number of prompts).

In the first stage, we leverage a powerful large language model (GPT-5.1) to extract seven semantic dimensions, including the rhetorical device, literal subject, metaphorical vehicle, overarching theme, emotional tone, subject keywords, and vehicle keywords. The first five are used for the subsequent scene description generation, while the last two, containing elements related to the subject and vehicle, are employed for object detection in the reward evaluation.

$$F_i = \Big[d_i^{\text{device}},\ d_i^{\text{sub}},\ d_i^{\text{veh}},\ d_i^{\text{theme}},\ d_i^{\text{emotion}},\ d_i^{\text{sub\_list}},\ d_i^{\text{veh\_list}} \Big] \tag{1}$$

To guarantee the accuracy of factor extraction, we employ a generate-verify-retry loop: for each candidate set of key factors F, the LLM is asked to compute a semantic coherence score and a rhetorical consistency score, and we accept the first candidate F satisfying the following criteria as the validated factor set F_i:

$$\mathrm{verify}(F, T) := \mathbb{I}\Big[s_{\mathrm{coh}}(F, T) \geq \tau_c \wedge s_{\mathrm{rhet}}(F, T) \geq \tau_r \Big], \tag{2}$$

where τ_c and τ_r denote predefined threshold values for coherence and rhetorical alignment.

In the second stage, inspired by the incremental process by which a painter develops a scene, we utilize reinforcement learning and construct an agentic framework to model the text-to-image generation process that simulates the multi-steps task in control(Ren et al., 2024). Specifically, we define a staged prompting sequence $\{P_i^1, P_i^2, \ldots, P_i^C\}$, where each prompt P_i^j introduces additional semantic elements to the scene. The sequence begins with a core subject description and is followed by progressively enriched content such as environmental context, lighting conditions, spatial layout, emotional tone, and other such attributes.

During training, the prompts $\{P_i^j\}_{j=1}^{C}$ are fed into the diffusion-based generator to yield images $\{I_i^j\}_{j=1}^{C}$, while F_i supplies the RL reward guiding prompt optimization and boosting semantic fidelity. At inference, only the final-stage prompts P_i^C will be used for image generation.

4.3 Stage-wise Image Generation

We formulate rhetorical image generation as a two-layer Markov Decision Process (MDP), where the inner layer models the denoising trajectory of the stable diffusion model and the outer layer captures the semantic progression guided by staged prompt refinement.

In the inner-layer MDP, the denoising process begins from pure Gaussian noise $\mathbf{x}_T \sim \mathcal{N}(0, I)$ and proceeds through T iterative steps to produce

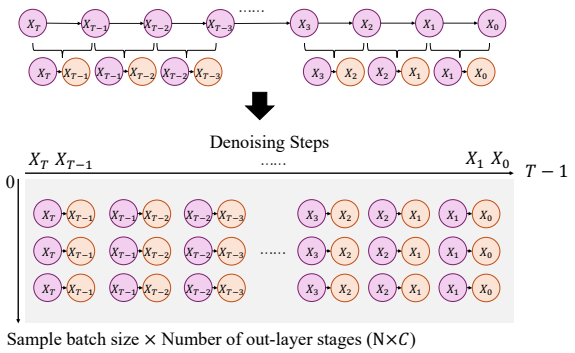


Figure 2: **Sampling and optimization over adjacent action pairs.** We collect denoising trajectories across N rhetorical inputs, each with C staged prompts and T diffusion steps, yielding $N \times C \times T$ adjacent action pairs. These pairs are shuffled across input, stage, and timestep dimensions to remove correlation. Sampled pairs are used to compute action likelihoods under current and past policies, and the policy is updated via PPO using reweighted semantic advantages.

a final image $\mathbf{x}_0$. At each denoising step t, the components of the MDP are defined as follows:

$$\begin{aligned} \text{State: } \mathbf{s}_t &:= (\mathbf{c}, t, \mathbf{x}_t) \\ \text{Action: } \mathbf{a}_t &:= \mathbf{x}_{t-1} \\ \text{Policy: } \pi_\theta\left(\mathbf{a}_t \mid \mathbf{s}_t\right) &:= p_\theta\left(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{c}\right) \end{aligned}$$

where $\mathbf{c}$ is the conditioning prompt, t is the diffusion timestep, and $\mathbf{x}_t$ is the intermediate latent. This process yields a trajectory $\tau = \{(\mathbf{s}_t, \mathbf{a}_t)\}_{t=T}^{0}$, where we assign credit and optimize the policy.

In the outer-layer MDP, a rhetorical input T_i is decomposed into a sequence of staged prompts $\{P_i^j\}_{j=1}^{C}$, progressively enriching the semantic content. At outer timestep j, the state is defined as $\mathbf{s}_j^{\text{outer}} := I_i^j$, the image generated from P_i^j, and the action $\mathbf{a}_j^{\text{outer}}$ represents the semantic substances added to P_i^j to obtain P_i^{j+1}. Each generated image I_i^j, i.e., outer-layer state $\mathbf{s}_j^{\text{outer}}$, will be evaluated by the reward function.

4.4 Optimization over Adjacent Image Pairs

To optimize the denoising policy π_θ, we leverage the two-layer MDP structure by first computing advantages of generated images at the outer layer, and then propagating these advantages to the inner-layer action space via trajectory-level discounting.

Given N rhetorical inputs, decomposed into C staged prompts, we obtain $N \times C$ images and their corresponding outer-layer states $\{\mathbf{s}_j^{\text{outer}}\}_{j=1}^{C}$. For each outer timestep j, a reward $r(I_i^j)$ is computed, and advantages $\hat{A}_i$ are estimated using Generalized Advantage Estimation (GAE):

$$\begin{aligned} \delta_j &:= r_j + \gamma V^{\pi_{\theta_{\text{old}}}}\left(I_{j+1}\right) - V^{\pi_{\theta_{\text{old}}}}\left(I_j\right), \\ \hat{A}_j &:= \sum_{l=0}^{C-j} (\gamma\lambda)^l \delta_{j+l}. \end{aligned} \tag{3}$$

These outer-layer advantages are then discounted backward across their corresponding denoising trajectories. For each inner timestep t , the step-wise advantage is:

$$\hat{A}_t^{(j)} := \gamma_{\text{denoise}}^t \cdot \hat{A}_j, \quad t = T, T-1, \ldots, 1.$$

As shown in Figure 2, we collect all denoising trajectories across the dataset, yielding $N \times C \times T$ adjacent action pairs $(\mathbf{s}_t, \mathbf{s}_{t+1})$. Each pair corresponds to a denoising action, where the latent is updated from $\mathbf{x}_t$ to $\mathbf{x}_{t-1}$. All action pairs are shuffled across rhetorical input, prompt stage, and timestep dimensions to remove temporal correlation, and mini-batches are sampled for policy updates.

For each sampled action pair $(\mathbf{s}_t, \mathbf{s}_{t+1})$, a single DDIM denoising step is applied to $\mathbf{s}_t$, producing the predicted mean μ_t and variance σ_t^2. A Gaussian $\mathcal{N}\left(\mu_t, \sigma_t^2\right)$ defines the distribution over possible actions. The likelihood of generating $\mathbf{a}_t = \mathbf{x}_{t-1}$ is computed under both current and old policies:

$$\begin{aligned} \pi_\theta\left(\mathbf{a}_t \mid \mathbf{s}_t\right) &= \mathcal{N}\left(\mathbf{x}_{t-1} \mid \mu_t, \sigma_t^2\right), \\ \pi_{\theta_{\text{old}}}\left(\mathbf{a}_t \mid \mathbf{s}_t\right) &= \mathcal{N}\left(\mathbf{x}_{t-1} \mid \mu_t^{\text{old}}, (\sigma_t^2)^{\text{old}}\right). \end{aligned} \tag{4}$$

The PPO update is applied over the sampled action pairs, using the clipped surrogate loss

$$\begin{aligned} \nabla_\theta \mathcal{L}_{\text{PPO}}(\theta) = \mathbb{E}_{(\mathbf{s}_t, \mathbf{s}_{t+1}) \sim \mathcal{D}} \Big[\nabla_\theta \min \big(& w_t \cdot \hat{A}_t^{(j)}, \\ & \text{clip}(w_t, 1-\epsilon, 1+\epsilon) \cdot \hat{A}_t^{(j)} \big) \Big], \end{aligned} \tag{5}$$

where the importance weight $w_t = \frac{\pi_\theta(\mathbf{a}_t|\mathbf{s}_t)}{\pi_{\theta_{\text{old}}}(\mathbf{a}_t|\mathbf{s}_t)}$, ϵ is the PPO clipping parameter, and $\mathcal{D}$ is the distribution of sampled action pairs. This formulation enables efficient reinforcement learning over diffusion-based generation, guided by rhetorical structure and semantic alignment.

4.5 Tailored Reward Mechanism

We design a tailored reward mechanism for rhetorical text-to-image generation, consisting of three modules: staged semantic alignment, elemental alignment, and aesthetic quality. Together, these modules guide the model to generate high-quality

images that capture both the coarse-to-fine structure of the rhetorical prompt and its true metaphorical intent.

Staged semantic alignment. To overcome the limitations of using a single global similarity metric-which treats all parts of the prompt equally and tends to overweight incidental visual attributes such as background or lighting while neglecting the hierarchical importance of semantic components-we propose a staged alignment strategy. Specifically, each scene prompt P_i^j is decomposed into an ordered sequence of j subsentences $\{S_i^k\}_{k=1}^j$, designed to reflect the prompt's progressive semantic elaboration from core concepts to contextual details.

We compute the unit-normalized image embedding $\mathbf{v}_i^j$, $j = 1, 2, \cdots, C$, for each training image I_i^j, and the unit-normalized text embeddings $\mathbf{u}_i^k$ for each sub-sentence. We then introduce a monotonic weight vector $\mathbf{w}^{(j)} \in \mathbb{R}^j$ satisfying $w_1 > w_2 > \cdots > w_j$ and $\sum_{k=1}^j w_k = 1$, to reflect the decreasing semantic centrality of later sub-sentences. The staged alignment reward is computed as

$$r^{\text{stage}}\left(I_i^j, P_j^j\right) = \sum_{k=1}^{j} w_k \left\langle \mathbf{v}_i^j, \mathbf{u}_i^k \right\rangle,$$

where $\langle \cdot, \cdot \rangle$ denotes the cosine similarity between the image and text embeddings. By assigning greater weight to earlier (core) components of the prompt, this reward formulation encourages the model to prioritize essential semantic elements in the generation process, while still attending to finer contextual details in later stages.

Elemental alignment. To correct the overemphasis on metaphorical vehicles that leads to semantic misalignment in generated images, we propose an elemental alignment mechanism that separately evaluates the vehicle and the true subject. Specifically, for each text input T_i, we construct two keyword sets: $d_i^{\text{sub list}}$, which includes a collection of subject-related elements such as core entities, associated states, and descriptive attributes; and $d_i^{\text{veh list}}$ which contains the metaphorical vehicle. Given a generated image I_i^j with its unit-normalized embedding $\mathbf{v}_i^j$, we compute the subject reward as

$$r^{\text{subject}}\left(I_i^j\right) = \sum_{k \in d_i^{\text{sub list}}} \left\langle \mathbf{v}_i^j, \frac{f_{\text{text}}(k)}{\|f_{\text{text}}(k)\|} \right\rangle,$$

which reflects the cumulative semantic alignment between the image and all subject-relevant components. Each term in the summation corresponds to the CLIP-based similarity between the image and a subject-related element, directly reflecting its semantic presence in the visual output.

To penalize inappropriate emphasis on the metaphorical vehicle, we define the vehicle penalty

$$\begin{aligned} \hat{f}_{\text{text}}(k) &:= \frac{f_{\text{text}}(k)}{\|f_{\text{text}}(k)\|}, \\ s_{\text{veh}}(I_i) &:= \max_{k \in d_i^{\text{veh\_list}}} \left\langle \mathbf{v}_i^j, \hat{f}_{\text{text}}(k) \right\rangle, \\ r^{\text{vehicle}}(I_i) &:= -\mathbf{1}\{s_{\text{veh}}(I_i) > \tau\}, \end{aligned} \tag{6}$$

where τ is a fixed similarity threshold, determined by empirical analysis of CLIP similarity scores to approximate the point at which a concept can be considered visually present in the image.

Aesthetic quality. To avoid generating visually unappealing images due to excessive emphasis on semantic and elemental alignment, we introduce an aesthetic quality score as a supplementary measure. This score is based on the LAION aesthetics predictor (Schuhmann et al., 2022), which is trained on 176,000 human-rated images to evaluate the aesthetic quality of the generated images.

To assess the effectiveness of our reward mechanism, we present examples as shown in Figure 4. Images with richer details that accurately reflect the intended semantic nuances are rewarded higher, while literal depictions of the metaphorical vehicle, such as pigs or pigsties, are penalized. These results demonstrate that our reward formulation effectively assesses both the semantic alignment and the appropriateness of visual elements in the generated images.

5 Experiments

5.1 Experiment settings

Dataset. In this work, we focus on metaphor (including personification) and simile, as they often describe concrete visual relationships between entities. For data preparation, we adopt the FLUTE dataset (Chakrabarty et al., 2022b) and apply a filtering step to select high-quality metaphor and simile samples, ensuring both rhetorical clarity and visual interpretability for fine-tuning.

Diffusion model. We use Stable Diffusion v1.4 (Rombach et al., 2022a) as the backbone generative model. Sampling is performed using the Denoising Diffusion Implicit Models (DDIM) algorithm

(a) "My bedroom is a pig sty."
(b) "The classroom was like a zoo, buzzing with chaotic energy."
(c) "He was looking like a wounded lion."
(d) "Her face is like a red apple."

Figure 3: **Qualitative comparison across models on rhetorical prompts.** Rhet2Pix generates semantically faithful and visually coherent images by correctly distinguishing metaphorical vehicles from intended subjects. Baseline models often fail to capture rhetorical meaning, producing literal or stylistically inconsistent outputs.

(a) Bad (b) Medium (c) Good

Figure 4: **Effectiveness of our reward mechanism in capturing rhetorical semantics.** From left to right, the reward increases with images that better align with the underlying rhetorical meaning, receiving higher scores. All examples are generated from the prompt "My bedroom is a pig sty."

(Song et al., 2020). For efficient parameter adaptation, we apply Low-Rank Adaptation (LoRA) (Hu et al., 2022) to the UNet (Ronneberger et al., 2015) component of the diffusion model, enabling fine-tuning with minimal additional memory overhead.

Experimental resources. Experiments were conducted using 8 NVIDIA A100 GPUs, each equipped with 80GB of HBM memory. Training each main baseline required approximately 8 GPU-days (1 day with 8 GPUs).

5.2 Comparison Evaluation

We compare our proposed method to representative baselines: Stable Diffusion (SD), a diffusion-based text-to-image model; DDPO, an RL-enhanced diffusion method fine-tuned by a VLM reward; MMaDA, Imagen, advanced multimodal diffusion foundation model; GPT-5.1 and Grok 3, advanced multimodal large language models.

Qualitative evaluation. As shown in Figure 3, Rhet2Pix excels at generating images that capture the deep semantic intent of rhetorical language while maintaining precise control over visual content. Our method distinguishes between metaphorical vehicles and intended subjects, ensuring that the image emphasizes the correct visual referents and avoids misleading literal interpretations.

In contrast, baseline models have consistent limitations. SD, DDPO, MMaDA, and Imagen generate visually generic outputs that align only superficially with prompt keywords, lacking the ability to infer or represent metaphorical meaning. GPT-5.1 aligns strongly at the surface level but often merges sub-

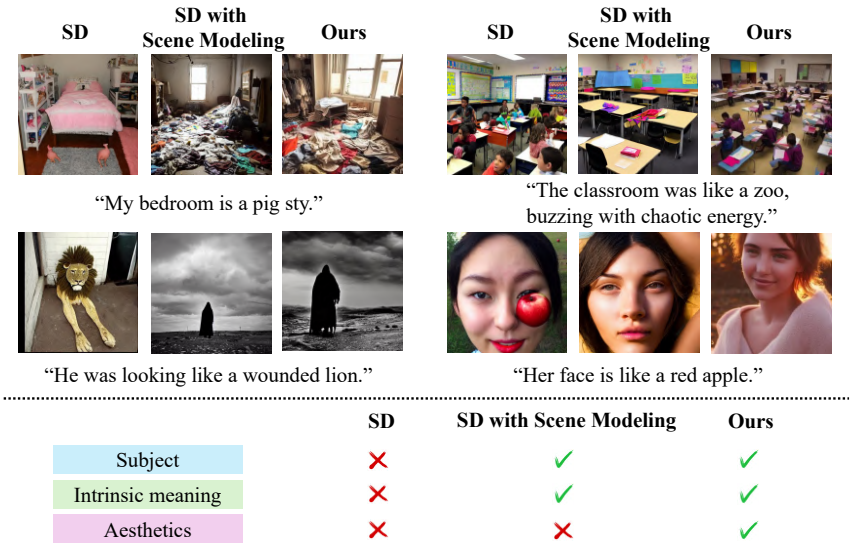


Figure 5: **Ablation study.** Scene modeling enhances semantic alignment by refining interpretation of rhetorical text, while our method further integrates reinforcement learning to improve artistic quality and image detail.

Methods	Sem. Align.	Elem. Align.	Sum
SD	0.22 ± 0.02	0.03 ± 0.20	0.25 ± 0.22
DDPO	0.22 ± 0.04	0.02 ± 0.21	0.24 ± 0.25
MMaDA	0.22 ± 0.02	0.17 ± 0.28	0.39 ± 0.29
Imagen	0.22 ± 0.03	0.14 ± 0.25	0.36 ± 0.26
GPT-5.1	0.24 ± 0.02	0.16 ± 0.20	0.41 ± 0.21
Grok-3	0.25 ± 0.03	0.17 ± 0.20	0.42 ± 0.21
Rhet2Pix	0.27 ± 0.02	0.40 ± 0.03	0.67 ± 0.05

Table 1: **Quantitative evaluation on alignment.** Sem. Align. and Elem. Align. denote semantic and elemental alignment scores, respectively.

ject and vehicle in unnatural ways (e.g., rendering a human head as an apple) and defaults to cartoon-style outputs for rhetorical prompts. Grok 3 occasionally captures figurative intent but frequently conflates subject and vehicle.

Quantitative evaluation. As reported in Table 1, Rhet2Pix performs consistently well across all alignment evaluation metrics, achieving a clear lead in overall performance. These results reflect its ability to balance high-level semantic understanding with precise control over visual elements when interpreting rhetorical prompts.

By contrast, SD, DDPO, MMaDA, and Imagen fail to reason about the overall intent of the input, producing images that do not reflect its full meaning. GPT-5.1 and Grok-3, while able to capture the general semantics thanks to large-scale multimodal training, struggle to disentangle subject and metaphor. Their focus on literal text often elevates the metaphorical vehicle to the main subject, misrepresenting the intended rhetorical meaning.

5.3 Ablation Studies

As shown in Figure 5, scene modeling (prompt engineering) and reinforcement learning (RL) are crucial for improving the alignment between rhetorical text and image generation. Scene modeling enhances the interpretation of rhetorical text by transforming it into more detailed and semantically enriched prompts, resulting in images that better reflect the intended meaning. However, using scene modeling alone still produces images with an unclear main subject (e.g., the person appears too small in the bottom-left panel) and erroneous details (e.g., the noisy classroom in the top-right panel contains no people). RL refines this process by guiding the model towards higher artistic quality and richer detail. Together, these components enable more precise and expressive image generation.

6 Conclusions

In this work, we identify a critical misalignment in rhetorical text-to-image generation, rooted in the direct use of metaphorical vehicle embeddings that drive literal imagery and obscure intended meanings. To overcome this, we propose Rhet2Pix, a framework that formulates rhetorical image generation as a two-layer, multi-step Markov Decision Process. Experimental results show that Rhet2Pix effectively captures the rhetorical meaning while precisely controlling the elements and their characteristics in the generated images. This framework paves the way for more nuanced multimodal generation grounded in complex linguistic expressions.

7 Limitations

Rhet2Pix leverages reinforcement learning to fine-tune the alignment between rhetorical semantics and visual output. However, fully addressing the challenges of rhetorical text-to-image generation may necessitate a fundamentally new backbone—one that surpasses conventional text-image embedding similarity. Achieving true end-to-end rhetorical generation requires a deep understanding of the rhetorical intent embedded in the input text. This presents a substantial challenge and remains an ambitious avenue for future research.

References

Arjun R Akula, Brendan Driscoll, Pradyumna Narayana, Soravit Changpinyo, Zhiwei Jia, Suyash Damle, Garima Pruthi, Sugato Basu, Leonidas Guibas, William T Freeman, and 1 others. 2023. Metaclue: Towards comprehensive visual metaphors research. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 23201–23211.

Marianne Arriola, Aaron Gokaslan, Justin T Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sekhar Sahoo, and Volodymyr Kuleshov. 2025. Block diffusion: Interpolating between autoregressive and diffusion language models. *arXiv preprint arXiv:2503.09573*.

Georgios Batzolis, Jan Stanczuk, Carola-Bibiane Schönlieb, and Christian Etmann. 2021. Conditional image generation with score-based diffusion models. *arXiv preprint arXiv:2111.13606*.

James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, and 1 others. 2023. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. 2024. Training diffusion models with reinforcement learning. *Preprint*, arXiv:2305.13301.

Stephen Brade, Bryan Wang, Mauricio Sousa, Sageev Oore, and Tovi Grossman. 2023. Promptify: Text-to-image generation through interactive prompt exploration with large language models. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pages 1–14.

Tuhin Chakrabarty, Arkadiy Saakyan, Debanjan Ghosh, and Smaranda Muresan. 2022a. Flute: Figurative language understanding through textual explanations. *arXiv preprint arXiv:2205.12404*.

Tuhin Chakrabarty, Arkadiy Saakyan, Debanjan Ghosh, and Smaranda Muresan. 2022b. Flute: Figurative language understanding through textual explanations. *Preprint*, arXiv:2205.12404.

Tuhin Chakrabarty, Arkadiy Saakyan, Olivia Winn, Artemis Panagopoulou, Yue Yang, Marianna Apidianaki, and Smaranda Muresan. 2023. I spy a metaphor: Large language models and diffusion models co-create visual metaphors. *arXiv preprint arXiv:2305.14724*.

Tuhin Chakrabarty, Xurui Zhang, Smaranda Muresan, and Nanyun Peng. 2021. Mermaid: Metaphor generation with symbolism and discriminative decoding. *arXiv preprint arXiv:2103.06779*.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2021. Knowledge neurons in pretrained transformers. *arXiv preprint arXiv:2104.08696*.

Zhirui Dai, Arash Asgharivaskasi, Thai Duong, Shusen Lin, Maria-Elizabeth Tzes, George Pappas, and Nikolay Atanasov. 2024. Optimal scene graph planning with large language model guidance. *Preprint*, arXiv:2309.09182.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. 2023. Raft: Reward ranked finetuning for generative foundation model alignment. *arXiv preprint arXiv:2304.06767*.

Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. 2023. Dpok: Reinforcement learning for fine-tuning text-to-image diffusion models. *Preprint*, arXiv:2305.16381.

Yingchaojie Feng, Xingbo Wang, Kam Kwai Wong, Sijia Wang, Yuhong Lu, Minfeng Zhu, Baicheng Wang, and Wei Chen. 2023. Promptmagician: Interactive prompt engineering for text-to-image creation. *IEEE Transactions on Visualization and Computer Graphics*, 30(1):295–305.

Jialu Gao, Kaizhe Hu, Guowei Xu, and Huazhe Xu. 2023. Can pre-trained text-to-image models generate visual goals for reinforcement learning? *Advances in Neural Information Processing Systems*, 36:38297–38310.

Hongyu Gong, Kshitij Gupta, Akriti Jain, and Suma Bhat. 2020. Illinimet: Illinois system for metaphor detection with contextual and linguistic information. In *Proceedings of the second workshop on figurative language processing*, pages 146–153.

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

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

Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. 2022. Cascaded diffusion models for high fidelity image generation. *Journal of Machine Learning Research*, 23(47):1–33.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Zijing Hu, Fengda Zhang, Long Chen, Kun Kuang, Jiahui Li, Kaifeng Gao, Jun Xiao, Xin Wang, and Wenwu Zhu. 2025. Towards better alignment: Training diffusion models with reinforcement learning against sparse rewards. *arXiv preprint arXiv:2503.11240*.

Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. 2023. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. *Advances in Neural Information Processing Systems*, 36:78723–78747.

Dongzhi Jiang, Ziyu Guo, Renrui Zhang, Zhuofan Zong, Hao Li, Le Zhuo, Shilin Yan, Pheng-Ann Heng, and Hongsheng Li. 2025. T2i-r1: Reinforcing image generation with collaborative semantic-level and token-level cot. *Preprint*, arXiv:2505.00703.

L. P. Kaelbling, M. L. Littman, and A. W. Moore. 1996. Reinforcement learning: A survey. *Preprint*, arXiv:cs/9605103.

Kimin Lee, Hao Liu, Moonkyung Ryu, Olivia Watkins, Yuqing Du, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, and Shixiang Shane Gu. 2023a. Aligning text-to-image models using human feedback. *Preprint*, arXiv:2302.12192.

Kimin Lee, Hao Liu, Moonkyung Ryu, Olivia Watkins, Yuqing Du, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, and Shixiang Shane Gu. 2023b. Aligning text-to-image models using human feedback. *arXiv preprint arXiv:2302.12192*.

Chee Wee Leong, Beata Beigman Klebanov, and Ekaterina Shutova. 2018. A report on the 2018 vua metaphor detection shared task. In *Proceedings of the workshop on figurative language processing*, pages 56–66.

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

Wenyi Mo, Tianyu Zhang, Yalong Bai, Bing Su, Ji-Rong Wen, and Qing Yang. 2024. Dynamic prompt optimizing for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26627–26636.

OpenAI, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, and et al. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2019. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*.

Mihir Prabhudesai, Anirudh Goyal, Deepak Pathak, and Katerina Fragkiadaki. 2024. Aligning text-to-image diffusion models with reward backpropagation. *Preprint*, arXiv:2310.03739.

Archiki Prasad, Alexander Koller, Mareike Hartmann, Peter Clark, Ashish Sabharwal, Mohit Bansal, and Tushar Khot. 2024. Adapt: As-needed decomposition and planning with language models. *Preprint*, arXiv:2311.05772.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr.

Krishan Rana, Jesse Haviland, Sourav Garg, Jad Abou-Chakra, Ian Reid, and Niko Suenderhauf. 2023. Sayplan: Grounding large language models using 3d scene graphs for scalable robot task planning. *Preprint*, arXiv:2307.06135.

Allen Z Ren, Justin Lidard, Lars L Ankile, Anthony Simeonov, Pulkit Agrawal, Anirudha Majumdar, Benjamin Burchfiel, Hongkai Dai, and Max Simchowitz. 2024. Diffusion policy policy optimization. *arXiv preprint arXiv:2409.00588*.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022a. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022b. High-resolution image synthesis with latent diffusion models. *Preprint*, arXiv:2112.10752.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. *Preprint*, arXiv:1505.04597.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Preprint*, arXiv:2205.11487.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski,

Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. *Preprint*, arXiv:2210.08402.

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

Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. 2024. Autoregressive model beats diffusion: Llama for scalable image generation. *Preprint*, arXiv:2406.06525.

Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. 2024. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *Preprint*, arXiv:2404.02905.

Yulia Tsvetkov, Leonid Boytsov, Anatole Gershman, Eric Nyberg, and Chris Dyer. 2014. Metaphor detection with cross-lingual model transfer. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 248–258.

Tony Veale. 2016. Round up the usual suspects: Knowledge-based metaphor generation. In *Proceedings of the Fourth Workshop on Metaphor in NLP*, pages 34–41.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. 2023. Diffusion model alignment using direct preference optimization. *Preprint*, arXiv:2311.12908.

Zhendong Wang, Yifan Jiang, Huangjie Zheng, Peihao Wang, Pengcheng He, Zhangyang Wang, Weizhu Chen, and Mingyuan Zhou. 2023. Patch diffusion: Faster and more data-efficient training of diffusion models. *Preprint*, arXiv:2304.12526.

Yutong Xie, Zhaoying Pan, Jinge Ma, Luo Jie, and Qiaozhu Mei. 2023. A prompt log analysis of text-to-image generation systems. In *Proceedings of the ACM Web Conference 2023*, pages 3892–3902.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. 2023. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Preprint*, arXiv:2304.05977.

Ling Yang, Ye Tian, Bowen Li, Xinchen Zhang, Ke Shen, Yunhai Tong, and Mengdi Wang. 2025. Mmada: Multimodal large diffusion language models. *Preprint*, arXiv:2505.15809.

Ling Yang, Zhaochen Yu, Chenlin Meng, Minkai Xu, Stefano Ermon, and Bin Cui. 2024. Mastering text-to-image diffusion: Recaptioning, planning, and generating with multimodal llms. *Preprint*, arXiv:2401.11708.

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

Shu Zhang, Xinyi Yang, Yihao Feng, Can Qin, Chia-Chih Chen, Ning Yu, Zeyuan Chen, Huan Wang, Silvio Savarese, Stefano Ermon, and 1 others. 2024. Hive: Harnessing human feedback for instructional visual editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9026–9036.

A Discussion on Multi-stage Image Generation

(1) Why do we adopt multi-stage image generation?

Existing text-to-image pipelines adopt the input text as prompts and convert it into word embeddings. The lack of genuine semantic understanding causes generated images to merely depict isolated visual elements of individual words, missing the deeper meaning conveyed by the text. Even approaches that use advanced multimodal models to parse the text into high-level semantic prompts before embedding still fall short: by feeding the literal subject, metaphorical vehicle, and environmental context simultaneously and without distinction, the model's generation capacity is spread across competing objectives. Consequently, the intended focal subject loses prominence, often appearing blurred or de-emphasized, while background elements or metaphorical vehicles become the dominant factors shaping the generated image.

Taking these into consideration, We propose a multi-stage framework that structures both prompt generation and RL fine-tuning into sequential phases, producing images that faithfully convey the text's meaning and exhibit rich visual detail. Figure 7 provides an illustrative walkthrough of this process.

- **Multi-Stage Prompting for Semantic Alignment.** We recursively derive a sequence of scene modeling prompts: starting with a prompt focused solely on the core subject, we then introduce prompts that gradually incorporate contextual, lighting, and compositional details—preserving subject clarity while progressively enhancing the scene's richness. During training, a stage-wise semantic-alignment reward preserves subject priority by assigning a higher weight to early, subject-focused prompts.

- **Multi-Stage Dependency Optimization for Detail Enrichment.** As the multi-stage prompts drive the generation of a sequence of interdependent images, each image incrementally enriches the previous one by introducing additional semantic details. We model this as a MDP where each image is a state, and the semantic difference between prompts is the action driving state transitions. Using Generalized Advantage Estimation (GAE), we compute an advantage for each intermediate state, which corresponds to the generated images, and propagate these values through the diffusion denoising steps to update the policy. By emphasizing advantages from later stages, the model learns to focus on progressively enriching details, leading to final images with enhanced vividness and clarity.

(2) How do we construct a multi-stage image generation process?

In Rhet2Pix, we implement multi-stage generation as a two-layer MDP that decouples semantic planning from pixel-level synthesis. For each rhetorical text input T_i, we first construct a sequence of C scene instructions $\{P_i^1, P_i^2, \ldots, P_i^C\}$, where each prompt P_i^j is generated by conditioning on both the validated semantic factors and the immediately preceding prompt P_i^{j-1}. This design ensures that each stage builds upon the last: P_i^1 establishes a precise description of the core subject, and each subsequent P_i^j introduces additional details- environmental context, lighting cues, compositional elements, and emotional tone-while preserving earlier content.

Formally, we view this as an outer-layer MDP: the state at stage j is the image I_i^j synthesized from P_i^j, and the action is the semantic increment Δ_i^j that transforms P_i^j into P_i^{j+1}. Through Generalized Advantage Estimation (GAE), we compute an advantage $\hat{A}_i^j$ for each state (i.e., each generated image) that quantifies its marginal benefit toward fulfilling the accumulated semantic objectives.

Simultaneously, the inner-layer MDP governs the diffusion denoising trajectory for any given prompt P_i^j: at timestep t, the state $\mathbf{s}_t = (\mathbf{c}, t, \mathbf{x}_t)$ transitions via action $\mathbf{x}_{t-1}$ under policy $\pi_\theta(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{c})$, where $\mathbf{c}$ is encoded from P_i^j. By backpropagating outer-layer advantages through this inner trajectory, we align low-level denoising updates with high-level semantic objectives.

At inference, as shown in Figure 6, only the final prompt P_i^C is passed to the fine-tuned diffusion generator to produce the output image. All intermediate prompts and images serve exclusively for advantage computation and policy learning, rather than as visible outputs. Unlike video or comic generation-which present a sequence of frames or panels by issuing a lot of prompts-our method iteratively refines few images(e.g. 3 images) through successive semantic prompts and outputs only the final, richly detailed result.

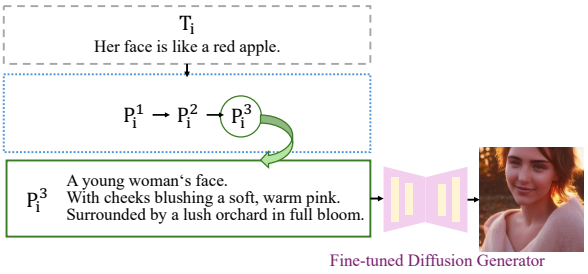


Figure 6: Illustrative example of the inference process of Rhet2Pix.

B Discussion on sparse rewards

(1) What is the sparse reward problem in diffusion-based text-to-image generation?

When we use reinforcement learning to fine-tune a pre-trained Stable Diffusion model, the only reward signal originates from the final denoised image $\mathbf{x}_0$, leaving all intermediate denoising actions without direct feedback. Methods like DDPO address this by assigning the same terminal reward $R(\mathbf{x}_0)$ to every timestep.

$$r_{T-1} = r_{T-2} = \cdots = r_0 = R(\mathbf{x}_0)$$

However, the denoising actions a_t at different timesteps have different effects on alignment: early iterations establish layout and overall style, whereas later iterations refine texture and fine detail (Hu et al., 2025). Treating each step identically disperses the reward across tasks of unequal importance, resulting in noisy updates, slower convergence, and reduced capacity to capture subtle visual features.

(2) How does our method effectively address sparse reward propagation?

As noted above, denoising actions at different timesteps contribute unevenly to the final image—early iterations set global layout, middle iterations refine style, and late iterations enhance object-level detail. Given our RL objective of semantic and visual alignment, we solve the reward sparisty along the trajectory by propagating the terminal advantage back through all diffusion steps via

$$\hat{A}_t^{(i,j)} = \gamma_{\text{denoise}}^t \hat{A}_i^j.$$

By propagating the terminal advantage backwards along the diffusion trajectory with a decay factor, diffusion steps closer to the final image receive proportionally greater advantage, reflecting their critical role in semantic and visual refinement. This mechanism overcomes the sparse reward problem in intermediate denoising steps and guides the model more efficiently toward our alignment objectives.

C Notation

The list of important symbols and their corresponding definition in this paper goes as Table 2.

D Pseudo-code

The pseudo-code of Rhet2Pix for one training round goes as Algorithm 1 and 2. Algorithm 1 illustrates the multi-stage scene modeling and sampling procedure, while Algorithm 2 presents the policy optimization process.

E Hyperparameters

We list model configuration and training hyperparameters of our method in Table 3.

F More experimental results

F.1 Quantitative evaluation

We apply our framework to fine-tune the diffusion model. Figure 8 displays the mean reward curve when fine-tuning the Stable Diffusion model using our method and DDPO as environmental steps (number of training images) increases. To highlight its underlying progression, both lines are smoothed by the exponential moving average (EMA) with the same parameter.

Under the same reward function and identical evaluation prompts from the rhetorical dataset, our method attains a higher reward than DDPO. Moreover, it recovers more quickly from the pronounced oscillations and initial declines during the warm-up phase, entering the correct performance regime sooner.

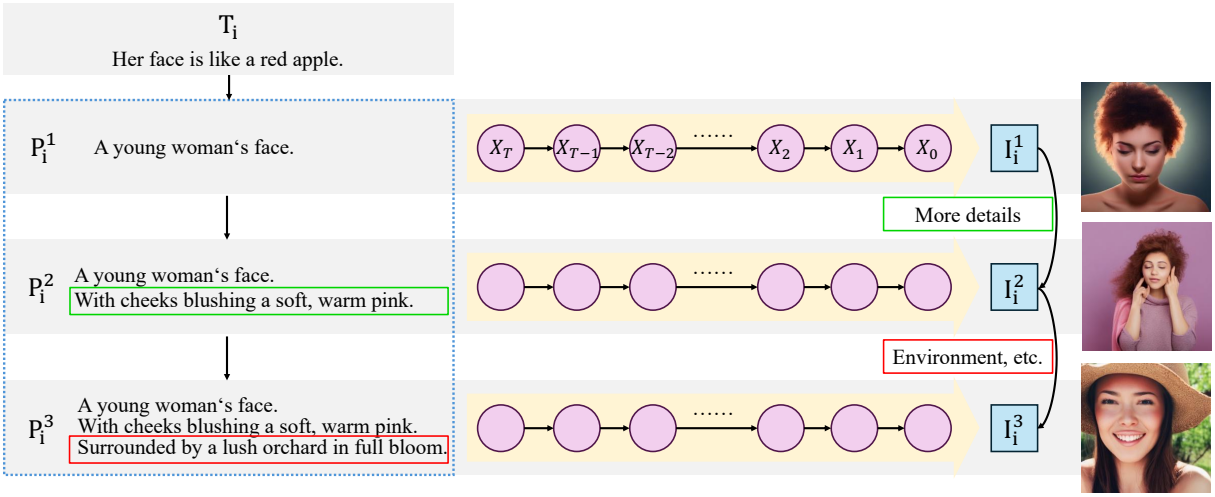


Figure 7: Illustrative example of the multi-stage image generation pipeline in Rhet2Pix.

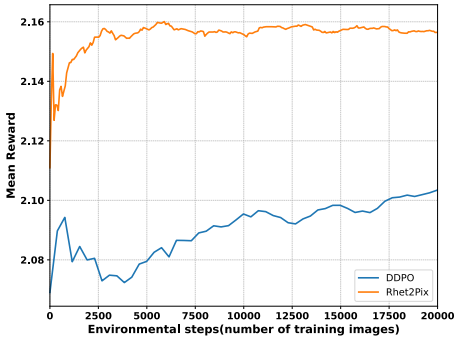


Figure 8: Reward curve of our method and DDPO when fine-tuning the diffusion model.

F.2 More samples

In this section, we present additional samples generated by Rhet2Pix, alongside outputs from several strong baselines, including Stable Diffusion (SD), DDPO, GPT-5.1, and Grok-3. All samples of Rhet2Pix are generated from the 50-th epoch from the same network weights in the same GPU.

As shown in Figure 9 and Figure 10, Rhet2Pix produces images with notable visual richness and strong semantic alignment, effectively capturing the rhetorical intent while ensuring that the visuals highlight the correct referents rather than relying on literal or misleading interpretations.

In contrast, baseline models exhibit limitations. Stable Diffusion and DDPO tend to produce generic outputs that match surface-level prompt cues but lack metaphorical understanding. Multimodal LLMs such as GPT-5.1 and Grok-3 occasionally capture figurative meaning but often struggle to differentiate rhetorical components, resulting in semantically ambiguous depictions.

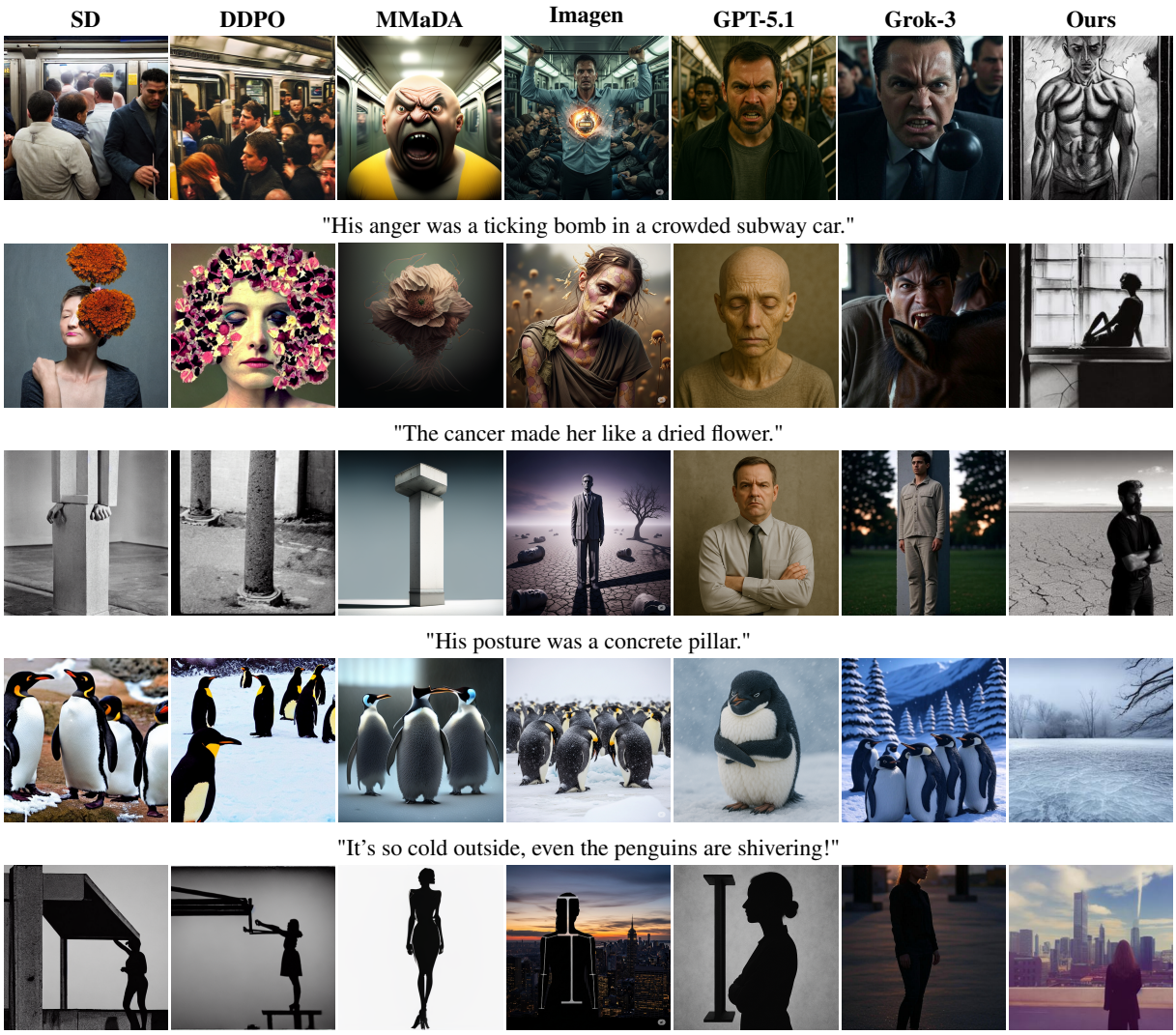
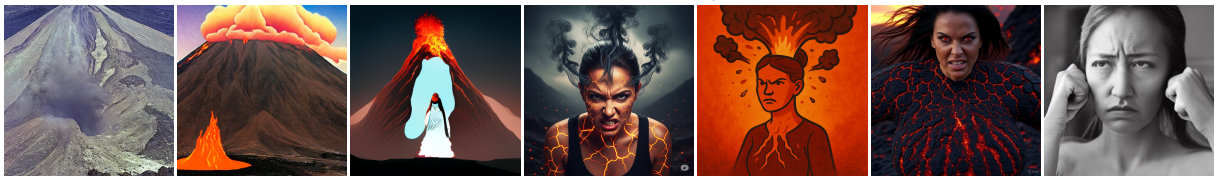


Figure 9: More samples generated by our method compared with other baselines

"It is so hot he is melting."



"She is a volcano ready to erupt."

"He is over the moon."

"Her teeth were like polished marble tiles."

"He is an ice cube in this room."

Figure 10: More samples generated by our method compared with other baselines

Table 2: Notation table

Symbol	Description
T_i	Input rhetorical texts
F_i	Key factors decomposed from T_i for scene modeling and reward evaluation
P_i^j	Multi-stage prompts for input text T_i
I_i^j	Image generated from prompt P_i^j
r_i^j	Reward of image I_i^j
$\mathcal{D}$	Replay buffer storing denoising trajectories of generated images
$\mathbf{c}$	Embedding vector produced from prompt P_i^j
$\mathbf{x}_t$	Diffusion model's latent tensor at timestep t
$\mathbf{s}_t = (\mathbf{c},\, t,\, \mathbf{x}_t)$	Tuple of prompt embedding, timestep index, and latent tensor
$\mathbf{a}_t = \mathbf{x}_{t-1}$	Updated latent tensor after one diffusion step
$V_\phi(I_i^j)$	State value of image I_i^j estimated by the critic network with parameters ϕ
δ_i^j	TD residual used in GAE computation
$\hat{A}_i^j$	Advantage of image I_i^j computed via GAE
γ_{denoise}	Discount factor applied across diffusion timesteps
$\hat{A}_t^{(i,j)}$	Advantage at diffusion step t for prompt P_i^j
w_t	Likelihood ratio used in PPO update
ϵ	Clipping threshold in PPO, bounding w_t to ensure stable updates

Algorithm 1 Pseudo-code for Rhet2Pix (Part I): Scene Modeling and Sampling

Input: Rhetorical texts $\{T_i\}_{i=1}^{N}$, number of stages C, diffusion steps T, pretrained diffusion model, LLM, reward function R.

Output: Intermediate rollout buffer $\mathcal{D}$.

Multi-stage scene modeling with self-validation
for $i = 1$ **to** N **do**
  $k \leftarrow 1$
  repeat
    $F_i^{(k)} \leftarrow \mathrm{LLM}(T_i)$
    $k \leftarrow k + 1$
  until $\mathrm{verify}(F_i^{(k)}, T_i) = 1$
  $F_i \leftarrow F_i^{(k)}$ {Generate validated factors}
  $P_i^1 \leftarrow \mathrm{LLM}(T_i, F_i)$
  for $j = 2$ **to** C **do**
    $P_i^j \leftarrow \mathrm{LLM}(P_i^{j-1}, \text{next factor})$ {Generate multi-stage prompts}
  end for
end for
Sampling (Two-layer MDP rollout)
Initialize $\mathcal{D} \leftarrow \emptyset$
for $i = 1$ **to** N **do**
  for $j = 1$ **to** C **do**
    $\mathbf{c} \leftarrow \mathrm{Embed}(P_i^j)$
    Sample $\mathbf{x}_T \sim \mathcal{N}(0, I)$
    for $t = T$ **to** 1 **step** -1 **do**
      $\mathbf{s}_t \leftarrow (\mathbf{c}, t, \mathbf{x}_t)$
      $\mathbf{a}_t \sim \pi_\theta(\cdot \mid \mathbf{s}_t)$
      $\mathbf{x}_{t-1} \leftarrow \mathbf{a}_t$
      if $t = 1$ **then**
        $r_i^j \leftarrow R(\mathbf{x}_0, P_i^j, F_i)$ {Reward only for the final step}
      else
        $r_i^j \leftarrow 0$
      end if
      store $(\mathbf{s}_t, \mathbf{a}_t, t, r_i^j)$ in $\mathcal{D}$
    end for
  end for
end for

Algorithm 2 Pseudo-code for Rhet2Pix (Part II): Policy Optimization

Input: Rollout buffer $\mathcal{D}$, discount factors γ, γ_{denoise}, GAE parameter λ.
Output: Optimized policy parameters θ, value parameters ϕ.

Training
for $i = 1$ **to** N **do**
  for $j = 1$ **to** C **do**
    $\delta_i^j \leftarrow r_i^j + \gamma\, V_\phi(I_i^{j+1}) - V_\phi(I_i^j)$
    $\hat{A}_i^j \leftarrow \sum_{l=0}^{C-j} (\gamma\lambda)^l\, \delta_i^{j+l}$ {Outer-layer advantage via GAE}
  end for
end for
for all $(\mathbf{s}_t, \mathbf{a}_t, t, r_i^j) \in \mathcal{D}$ **do**
  $\hat{A}_t^{(i,j)} \leftarrow \gamma_{\text{denoise}}^t\, \hat{A}_i^j$ {Inner-layer advantage propagation}
  update transition to $(\mathbf{s}_t, \mathbf{a}_t, t, r_i^j, \hat{A}_t^{(i,j)})$ in $\mathcal{D}$
end for
Shuffle and partition $\mathcal{D}$ into minibatches
for all minibatch $\mathcal{B}$ **do**
  for all $(\mathbf{s}_t, \mathbf{a}_t, t, r_i^j, \hat{A}_t^{(i,j)}) \in \mathcal{B}$ **do**
    $w_t \leftarrow \dfrac{\pi_\theta(\mathbf{a}_t \mid \mathbf{s}_t)}{\pi_{\theta_{\text{old}}}(\mathbf{a}_t \mid \mathbf{s}_t)}$
    $\mathcal{L}_{\text{PPO}} \leftarrow \mathbb{E}\big[\min(w_t\, \hat{A}_t^{(i,j)},\ \text{clip}(w_t, 1-\epsilon, 1+\epsilon)\, \hat{A}_t^{(i,j)})\big]$ {PPO update}
    $\mathcal{L}_V \leftarrow \mathbb{E}\big[(V_\phi(\mathbf{s}_t) - \hat{A}_t^{(i,j)})^2\big]$
  end for
  update θ by minimizing $\mathcal{L}_{\text{PPO}}$
  update ϕ by minimizing $\mathcal{L}_V$
end for

Table 3: Model configuration and training hyperparameters.

Module	**Hyperparameter**	**Value**
ViT Encoder	Patch Size	8
	Transformer Depth	1
	Embedding Dimension	128
	Attention Heads	4
	Activation Function	GELU
Patch Embedding	Embedding Normalization	Disabled
	Conv1 (Kernel / Stride)	8 / 4
	Conv2 (Kernel / Stride)	3 / 2
	GroupNorm	Disabled
	Patch Embed Output Dim	128
Transformer Block	MLP Expansion Ratio	4×
	Dropout Rate	0.0
	Attention Type	Flash Attention
Spatial Embedding	Projection Dimension	128
	Spatial Embedding Dropout	0.0
	Weight Initialization	$\mathcal{N}(0, 0.02^2)$
RGB Critic	Input Feature	ViT + Spatial Emb
	MLP Hidden Dimensions	[256, 256, 256]
	Activation Function	Mish
	Residual Style	Enabled
	LayerNorm	Disabled
	Output Dimension	1
Training	Train Batch Size	3
	Gradient Accumulation Steps	1
	Image Conditioning Steps	1
DDIM Sampling	Denoising Steps	50
	Guidance Scale	5.0
	Noise Weight (η)	1.0
	Scheduler Type	DDIM
UNet Optimizer	Optimizer	AdamW
	Learning Rate	3×10^{-4}
	Betas	(0.9, 0.999)
	Weight Decay	1×10^{-4}
	ϵ	1×10^{-8}
Critic Optimizer	Optimizer	AdamW
	Learning Rate	1×10^{-3}
	Betas	(0.9, 0.999)
	Weight Decay	1×10^{-4}
	ϵ	1×10^{-8}