# FPT UNIVERSITY

## Multi-round Interleaving Reasoning for Text-to-Image Generation

**Dao Nhat Tan**
**Hoang Van Vu**
**Ngo Anh Dung**
**Bui Minh Son**

Supervisor: Dr. Cao Van Mai

Bachelor of Artificial Intelligence
Hoa Lac Campus – FPT University
2025

# Acknowledgements

We would like to express our deepest gratitude to Dr. Cao Van Mai, our mentor, for his continuous support, valuable guidance, and insightful feedback throughout the development of this project. His expertise and encouragement have been instrumental in helping us refine our ideas and overcome challenges during each stage of our research.

Our sincere appreciation also goes to FPT University for providing the resources, facilities, and academic environment that made this project possible. The support from the faculty and the School of Information Technology has inspired us to explore new ideas and pursue excellence in our field.

We are equally thankful to our friends and classmates, whose cooperation, constructive suggestions, and moral support motivated us throughout this journey. Their presence made the process not only productive but also enjoyable.

Lastly, we would like to extend our heartfelt thanks to our families for their unconditional love, patience, and encouragement. Their belief in us has been a constant source of strength and motivation.

This project is the result of collective effort, collaboration, and shared passion for learning. To everyone who contributed to our success, we offer our sincerest thanks.

# Contents

# List of Tables

# List of Figures

# List of Abbreviations and Acronyms

**AI** : Artificial Intelligence

**AI:** Artificial Intelligence

**CFG:** Classifier-Free Guidance

**CLIP:** Contrastive Language–Image Pre-training

**COCO:** Common Objects in Context

**CoT:** Chain-of-Thought

**CPU:** Central Processing Unit

**CUDA:** Compute Unified Device Architecture

**DALL-E:** Model name by OpenAI

**DDPM:** Denoising Diffusion Probabilistic Models

**DETR:** DEtection TRansformer

**FID:** Fréchet Inception Distance

**FP16:** 16-bit Floating Point

**GAN:** Generative Adversarial Network

**GenEval:** Generation Evaluation

**GPU:** Graphics Processing Unit

**Img2Img:** Image-to-Image

**IRG:** Interleaving Reasoning–Generation

**IRGL:** Interleaving Reasoning–Generation Learning

**LAION:** Large-scale Artificial Intelligence Open Network

**LDM:** Latent Diffusion Models

**LLM:** Large Language Model

**LoRA:** Low-Rank Adaptation

**LSTM:** Long Short-Term Memory

**MSE:** Mean Squared Error

**NF4:** NormalFloat 4-bit

**NLP:** Natural Language Processing

**QLoRA:** Quantized Low-Rank Adaptation

**RAM:** Random Access Memory

**RNN:** Recurrent Neural Network

**RoPE:** Rotary Positional Embeddings

**SDXL:** Stable Diffusion XL

**SM:** Streaming Multiprocessor

**SoTA:** State-of-the-Art

**SSD:** Solid State Drive

**SwiGLU:** Swish-Gated Linear Unit

**T2I:** Text-to-Image

**TIFA:** Text-to-Image Faithfulness Evaluation

**Txt2Img:** Text-to-Image

**UNet:** U-shaped Network

**VAE:** Variational Autoencoder

**ViT:** Vision Transformer

**VQA:** Visual Question Answering

**VRAM:** Video Random Access Memory

# Abstract

This thesis investigates a modular adaptation of the Interleaving Reasoning–Generation (IRG) paradigm for text-to-image synthesis. Although diffusion models generate visually realistic images, they consistently struggle with compositional reasoning, and existing IRG approaches that address this issue rely on large, tightly integrated models.

We propose a modular pipeline that separates reasoning, image generation, and visual feedback, enabling iterative refinement using lightweight components. A small language model generates structured reasoning plans to guide and adjust image synthesis across iterations.

Experimental results show that this approach does not reliably overcome the compositional limitations of standard diffusion models. In some cases, reasoning-based refinement improves semantic alignment, but these gains are often offset by reduced visual quality or unstable generation behavior. Overall, the findings suggest that while the modular IRG concept is viable, its practical benefits are limited with mid-scale models and are more likely to emerge when applied to stronger reasoning and generative architectures.

**Keywords**: Text-to-Image Generation, Interleaving Reasoning, Multi-round Reasoning, Visual Refinement, Multimodal Learning

# 1 Introduction

The rapid advancement of text-to-image generation has democratized visual content creation, yet significant challenges remain in achieving compositional accuracy and semantic fidelity. This chapter establishes the motivation for developing resource-efficient reasoning-enhanced generation systems, identifies key limitations of existing approaches, and articulates the specific contributions of this thesis. We begin by examining the evolution of text-to-image technology and its current limitations, then outline our objectives in bridging the gap between cutting-edge reasoning capabilities and educational accessibility.

## 1.1 Motivation and Background

In recent years, text-to-image generation has emerged as one of the fastest-growing areas within modern Artificial Intelligence. Systems such as Stable Diffusion XL [1], DALL-E 3 [2], and Midjourney have enabled users to synthesize high-fidelity, photorealistic images from natural language descriptions alone, making image generation accessible to non-experts [3].

Central to this success is the evolution of diffusion probabilistic models, which have largely superseded earlier Generative Adversarial Networks in terms of training stability and output diversity [4]. Consequently, text-to-image technology has found widespread application across diverse domains, ranging from artistic design and advertising to entertainment and educational visualization.

## 1.2 Limitations of Traditional Text-to-Image Generation

Despite the impressive capabilities of state-of-the-art diffusion models, current systems face significant bottlenecks when tasked with interpreting complex or compositionally dense prompts. These limitations can be categorized into two main challenges.

### 1.2.1 Challenges in Compositional Semantics and Fidelity

Current text-to-image models frequently fail when handling prompts that require precise semantic alignment and structural composition. Specifically, these struggles manifest themselves in three key areas. Models often mix attributes, such as color or texture, between different objects. For instance, a prompt for "a red cat and a blue dog" may result in "a purple cat and a purple dog." This occurs because the system fails to correctly bind specific adjectives to their respective nouns [5]. Models find it difficult to interpret

spatial prepositions like "on top of," "behind," or "to the left of." Consequently, objects often appear in random or physically impossible arrangements rather than the requested layout. When prompted for specific quantities (e.g., "three apples"), the models frequently generate incorrect numbers. This limitation reveals a lack of inherent mathematical or counting ability within the diffusion process.

### 1.2.2 Lack of Reasoning in Single-Shot Paradigms

The root cause of many of the previously discussed failures lies in the single-shot generation paradigm used by most traditional text-to-image systems. In this approach, the model processes the prompt only once and synthesizes the final image in a single forward pass, without any intermediate planning or self-correction.
This stands in contrast to human artists, who typically plan a layout, sketch a draft, and iteratively refine details. Diffusion models, however, lack such a reasoning or reflection phase-they cannot evaluate their own outputs or adjust mistakes mid-generation.
Recent frameworks such as Interleaving Reasoning-Generation [6] attempt to address this limitation by introducing iterative reasoning steps. However, these unified models are computationally expensive, often requiring substantial computational resources, making them impractical for educational or resource-constrained research settings.

## 1.3 Objectives and Contribution

The primary objective of this thesis is to examine whether a modular Interleaving Reasoning–Generation framework can partially alleviate compositional reasoning limitations in text-to-image generation under computational constraints, with an emphasis on analyzing effectiveness and failure cases rather than achieving state-of-the-art performance. In addition, the proposed approach is evaluated using established quantitative metrics, which are treated strictly as analytical tools for measuring compositional accuracy, semantic alignment, and visual quality across refinement iterations, rather than as direct optimization targets. Overall, this thesis aims to provide empirical insight into when and why modular interleaving reasoning succeeds or fails, thereby informing future applications of the approach with more capable reasoning and generative models.

## 1.4 Organization

The remainder of this report is structured as follows. Chapter 2 reviews the theoretical background, covering diffusion models, large language model-guided generation, and the Interleaving Reasoning-Generation framework. Chapter 3 details the theoretical foundations including transformer architectures, diffusion models, and vision-language alignment. Chapter 4 presents the proposed methodology, including the modular architecture,

data synthesis strategy, and iterative refinement pipeline. Chapter 5 describes the experimental setup, including the evaluation protocol, metrics, and implementation details, and presents the empirical results with analysis. Chapter 6 concludes the study and outlines directions for future work.

## 1.5 Chapter Summary

We have established the research context by highlighting the significant limitations of existing text-to-image models, particularly their struggle with complex queries requiring spatial and compositional logic. The primary objective has been defined as constructing a multi-round reasoning architecture capable of enhancing prompt adherence through iterative refinement. To effectively address these challenges, it is crucial to first understand the existing solutions and identify specific technological gaps. Consequently, Chapter 2 will delve into a detailed analysis of related work, positioning our proposed approach within the broader context of current text-to-image advancements.

# 2 Related Work

This chapter provides a comprehensive review of the literature surrounding text-to-image generation, focusing specifically on the paradigm shift towards reasoning-based generation. We analyze the foundational work of Interleaving Reasoning-Generation, the evolution of diffusion models, and existing methods for large language model-guided synthesis. Finally, we identify the research gap regarding the accessibility of these advanced systems in educational environments.

## 2.1 Foundational Work: Interleaving Reasoning-Generation

Interleaving Reasoning–Generation introduces an iterative alternative to conventional single-pass text-to-image generation by explicitly alternating between textual reasoning and visual synthesis. Rather than mapping an input prompt directly to a final image, this paradigm incorporates intermediate reasoning steps that analyze the current visual output and guide subsequent refinement stages. The objective of this process is to improve compositional correctness and maintain semantic consistency across successive generations.
Existing text-to-image approaches generally fall into three categories: single-shot diffusion models without explicit reasoning, pre-generation planning methods that perform reasoning before synthesis, and post-generation editing approaches that modify images only after an initial output is produced. Interleaving reasoning and generation departs from these paradigms by incorporating reasoning within a multi-round loop, where each iteration reflects on the current image and produces a refined version. This design enables visual grounding, allowing the system to identify mismatches such as incorrect object counts or misplaced attributes and to incrementally correct them.
To support this interleaved process, the original framework adopts a two-stage training strategy that strengthens both reasoning and refinement capabilities. The first stage focuses on learning structured reasoning behaviors, while the second stage aligns textual reflections with visual updates in an end-to-end loop, ensuring that identified errors are consistently translated into targeted visual improvements. This foundational framework serves as the conceptual basis for the modular adaptation explored in this thesis.

## 2.2 Taxonomy of Text-to-Image Synthesis Approaches

To contextualize this work, existing text-to-image synthesis methods are categorized into three architectural paradigms based on how language understanding and visual generation

are integrated: diffusion-based models, unified multimodal transformers, and reasoning-augmented systems.

### 2.2.1 Text-to-Image Diffusion Models

Diffusion-based models form the dominant paradigm in modern text-to-image generation. Latent Diffusion Models, exemplified by Stable Diffusion, significantly reduce computational cost by performing the diffusion process in a compressed latent space using a Variational Autoencoder. Building on this foundation, Stable Diffusion XL extends the architecture with a larger UNet and dual text encoders, enabling stronger prompt adherence and native high-resolution image generation, which motivates its selection as the image generator in this thesis.

Commercial diffusion systems further emphasize the importance of strong language understanding in text-to-image generation. Models such as DALL-E 3 and Imagen enhance semantic fidelity through large-scale caption augmentation and powerful pre-trained text encoders, respectively. Despite their strengths in photorealism and aesthetic quality, diffusion-based models continue to struggle with compositional semantics such as counting, spatial relations, and attribute binding due to the absence of explicit reasoning mechanisms during generation.

### 2.2.2 Unified Transformer Models

Recent research has introduced unified multimodal transformers that represent both text and images as token sequences and process them within a single autoregressive backbone. Representative examples include Show-o [7] and Janus-Pro [8], which generate interleaved text–image token streams to tighten semantic coupling across modalities. While unified token representations can improve compositional instruction following and cross-modal consistency, these systems typically require substantially larger model scale and heavier infrastructure than standard diffusion pipelines, which limits their accessibility in resource-constrained educational settings.

### 2.2.3 Reasoning-Augmented Text-to-Image Systems

A growing line of research introduces explicit reasoning processes into text-to-image generation to improve semantic fidelity and compositional accuracy. Interleaving Reasoning-Generation (IRG) integrates multi-round textual reflection and visual refinement. More recent directions explore reinforcement learning to strengthen reasoning behaviors and stabilize multi-step decision making in generative pipelines [9, 10]. These approaches indicate that stronger reasoning supervision can partially mitigate compositional errors such as incorrect counts, spatial misplacement, and attribute leakage, although improvements remain sensitive to model scale and feedback quality.

This thesis contributes to the reasoning-augmented paradigm by demonstrating that iterative reasoning benefits can be obtained through modular architecture combining a lightweight large language model with a diffusion model. Unlike unified reasoning-generation systems, which demand substantial computational resources, the proposed approach is accessible on consumer hardware and therefore more suitable for educational and resource-constrained research environments.

## 2.3 Large Language Model-Guided Image Generation

Prior to the development of interleaving reasoning approaches, researchers primarily explored the use of Large Language Models to guide image synthesis through structural planning. Systems such as LayoutGPT [11] utilize language models to generate spatial layouts (bounding boxes) that condition the diffusion model. While effective for initial spatial positioning, this approach operates as a one-way feed-forward process; it lacks a feedback loop, meaning it cannot correct errors once the image is generated. Similarly, other grounding methods introduce specific grounding tokens to achieve precise spatial control, though they often necessitate manual bounding box annotations or auxiliary models. Fundamentally, these methods differ from interleaving approaches as they restrict reasoning to the pre-generation phase, thereby forgoing the capacity for reflection or self-correction based on the actual visual output.

## 2.4 Iterative Refinement Systems

Recent research has increasingly focused on integrating reasoning capabilities into the generative process. InstructPix2Pix [12] pioneered iterative image editing through natural language instructions, demonstrating the efficacy of progressive refinement; however, its scope is limited to modifying existing images rather than de novo generation. Drawing inspiration from self-correction mechanisms in large language models, where models refine text outputs through introspection, some approaches have attempted to transfer this paradigm to the visual domain. Nevertheless, unlike methods that rely on external Visual Question Answering modules for feedback, integrated reasoning distinguishes itself by incorporating reasoning capabilities directly into the generative backbone, effectively adapting the self-correction concept to a native multimodal context without reliance on auxiliary classifiers.

## 2.5 Accessibility Challenges of Unified Models

While unified multimodal models demonstrate impressive reasoning-generation capabilities, deploying these architectures presents three critical barriers for educational research. The weights for state-of-the-art unified generative models are often proprietary or with-

held from the public domain. A typical multi-billion parameter unified model requires substantial computational resources for inference, far exceeding the capacity of standard educational hardware. Handling mixed text-image token streams-where the model autoregressively generates interleaved sequences of text and visual tokens-requires specialized frameworks that differ significantly from standard open-source libraries.
Consequently, these constraints necessitate the adoption of a modular adaptation strategy for this project, enabling educational engagement with reasoning-augmented generation concepts without requiring access to cutting-edge hardware or proprietary models.

## 2.6 Research Gap and Our Position

Despite the breakthrough of integrated reasoning-generation approaches, a critical gap exists: unified architectures and proprietary training data are inaccessible to students and researchers in resource-constrained environments, preventing educational engagement with interleaving reasoning concepts.
This project addresses this gap through a modular adaptation. We aim to retain the core benefits of interleaving reasoning, iterative refinement, and visual feedback by replacing the unified model with a pipeline of accessible components: a lightweight language model for reasoning, Stable Diffusion XL for generation, and Contrastive Language-Image Pre-training for feedback.
We do not claim to match state-of-the-art performance of unified approaches. Instead, our contribution lies in empirically quantifying how much benefit can be preserved in a modular, education-friendly implementation, specifically answering questions regarding the trade-offs between unified and modular architectures.

## 2.7 Chapter Summary

Through a critical analysis of state-of-the-art text-to-image methodologies, ranging from standard diffusion models to advanced refinement techniques, we identified key shortcomings in current single-shot systems. Specifically, the discussion highlighted the inability of existing models to self-correct and the high hardware barriers preventing widespread accessibility. These identified limitations underscore the necessity for a robust framework that supports interleaving reasoning processes. Chapter 3 establishes the theoretical foundations and core concepts that serve as the basis for the system architecture proposed in this study.

# 3 Theoretical Background

Having reviewed the landscape of reasoning-enhanced text-to-image generation in Chapter 2, this chapter establishes the theoretical foundations necessary to understand the technical mechanisms underlying our proposed modular framework. We provide detailed exposition of three critical components: Large Language Models and their reasoning capabilities through transformer architectures and fine-tuning methodologies, text-to-image diffusion models including the mathematical formulation of denoising processes and latent diffusion frameworks, and vision-language alignment mechanisms, particularly contrastive learning-based multimodal embeddings that enable iterative feedback.

## 3.1 Large Language Models and Reasoning

### 3.1.1 Transformer Architecture

The introduction of the Transformer architecture by Vaswani et al [13] marked a major shift in Natural Language Processing, replacing recurrent approaches such as Recurrent Neural Networks and Long Short-Term Memory networks. Unlike sequential architectures that process tokens one step at a time, the Transformer processes the entire sequence in parallel, fully exploiting graphics processing unit parallelism.

At its core is the self-attention mechanism, which quantifies the contextual relevance between all tokens in a sequence, independent of positional distance. This mechanism operates by projecting the input into three learned matrices: Query, Key, and Value. The attention output is computed as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

where $d_k$ is the dimensionality of the key vectors, acting as a scaling factor to stabilize gradients.

In this project, we employ a decoder-only Transformer, similar to architectures used in unified multimodal models. A text sequence $x$ is modeled autoregressively as:

$$P(x) = \prod_{t=1}^{T} P\left(x_t \mid x_{1:t-1}; \theta\right) \tag{2}$$

where $\theta$ denotes model parameters. This ability to capture long-range dependencies provides the foundation for multi-step reasoning layers used later in our pipeline.

Figure 1 illustrates the Transformer model architecture as introduced by Vaswani et al[13].

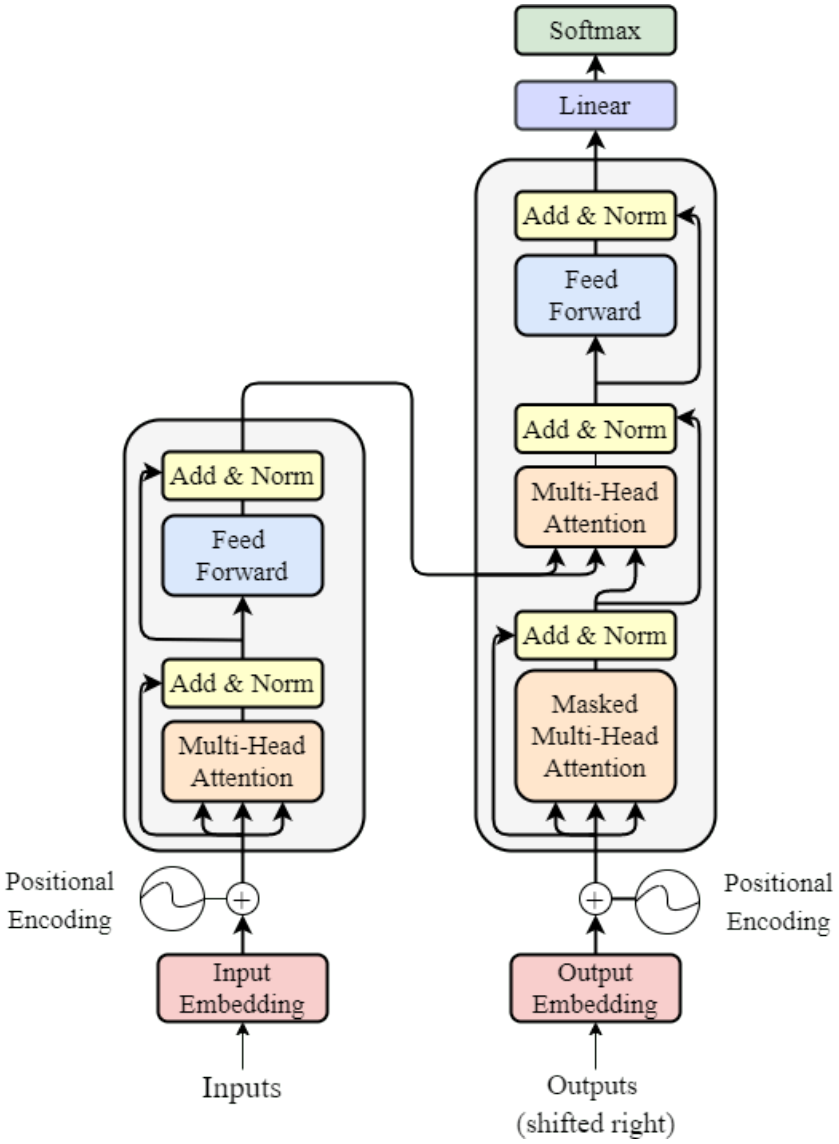


Figure 1: The Transformer model architecture.

### 3.1.2 Chain-of-Thought Reasoning

Although Large Language Models possess strong semantic knowledge, their ability to solve complex reasoning tasks is limited when producing outputs directly in a single step. Chain-of-Thought prompting improves reasoning by encouraging the model to generate intermediate reasoning steps before producing the final answer [14]. Formally, instead of directly modeling $P(Y \mid X)$, Chain-of-Thought introduces a latent variable $Z$ representing the reasoning chain:

$$P(Y \mid X) = \sum_{Z} P(Y \mid X, Z)\, P(Z \mid X) \tag{3}$$

This structure allows the large language model in our system to explicitly reason about spatial relations, object counts, and attribute bindings before generating directives for the

diffusion model. This behavior aligns closely with the Interleaving Reasoning-Generation framework [6] and reinforcement-driven reasoning improvements explored in related work.

### 3.1.3 Qwen Model Family

Qwen is a family of large language models developed by Alibaba Cloud, designed for deep semantic understanding and instruction following. For this work, we adopt Qwen 2.5-3B-Instruct, a model chosen for its balance between reasoning strength and computational efficiency.

This model integrates modern architectural components such as Rotary Positional Embeddings for improved positional encoding and Swish-Gated Linear Unit activation for enhanced non-linearity. With 3 billion parameters, this variant offers strong reasoning capabilities while maintaining a low memory footprint. Compared to larger variants, it enables simultaneous deployment of the language model and diffusion model on constrained hardware-crucial for educational and resource-constrained environments-without compromising the depth of prompt analysis required in our iterative refinement pipeline.

## 3.2 Text-to-Image Diffusion Models

### 3.2.1 Fundamentals of Denoising Diffusion Probabilistic Models

Diffusion Models [15] form a class of generative models rooted in non-equilibrium thermodynamics. The framework consists of two complementary stochastic processes.

The forward process is a fixed Markov chain $q(x_{1:T} \mid x_0)$ that gradually adds Gaussian noise to the original data $x_0$ over $T$ timesteps:

$$x_t = \sqrt{\alpha_t} x_0 + \sqrt{1 - \alpha_t}\epsilon, \quad \epsilon \sim \mathcal{N}(0, I) \tag{4}$$

Eventually, $x_T$ converges to an isotropic Gaussian distribution $\mathcal{N}(0, I)$.

The reverse process is a learned distribution $p_\theta(x_{t-1} \mid x_t)$ that attempts to invert the noising process and reconstruct the data from pure noise. During training, the neural network $\theta$ is optimized to predict the noise added at each timestep. The widely used simplified loss is the Mean Squared Error between true noise $\epsilon$ and predicted noise $\epsilon_\theta$:

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{x_0, \epsilon \sim \mathcal{N}(0,I), t} \left[ \|\epsilon - \epsilon_\theta(x_t, t, c)\|_2^2 \right] \tag{5}$$

where $c$ denotes the conditioning signal (e.g., text prompt). Recent approaches refine this process at inference time to enhance image fidelity.

Figure 2 illustrates the directed graphical model of the diffusion process, showing the forward noising process and reverse denoising process.

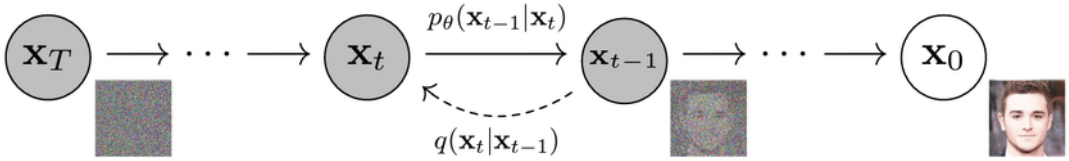


Figure 2: The directed graphical model of the diffusion process.

### 3.2.2 Latent Diffusion Models

A major drawback of traditional Denoising Diffusion Probabilistic Models is their computational cost when operating directly in pixel space. Latent Diffusion Models address this by performing diffusion in a compressed latent representation.
A pretrained Variational Autoencoder encodes the image into a semantic latent:

$$z = E(x) \tag{6}$$

The diffusion process is applied entirely in this latent space, substantially reducing dimensionality. After denoising, the final output is reconstructed by the Variational Autoencoder decoder:

$$x = D(z) \tag{7}$$

This strategy dramatically improves memory and compute efficiency while maintaining high perceptual quality, making Latent Diffusion Models the standard approach for modern text-to-image systems.

### 3.2.3 Stable Diffusion XL Architecture

Stable Diffusion XL represents a major evolution of the Latent Diffusion Model paradigm, designed to improve resolution, semantic fidelity, and prompt adherence. Its key architectural innovations include an expanded UNet backbone that increases the capacity of the UNet by incorporating additional Transformer blocks at lower-resolution stages, enhancing the model's ability to capture complex high-level semantics. The architecture integrates two independent language encoders whose embeddings are concatenated, enabling the system to capture both conceptual semantics and detailed syntactic nuances. Additionally, Stable Diffusion XL introduces additional conditioning inputs such as original image size and crop coordinates. This allows the model to better distinguish between full images and cropped inputs during training, improving its ability to generate coherent, high-resolution compositions.
Figure 3 shows the Stable Diffusion XL pipeline architecture with base and refiner models. The figure is adapted from Podell et al.

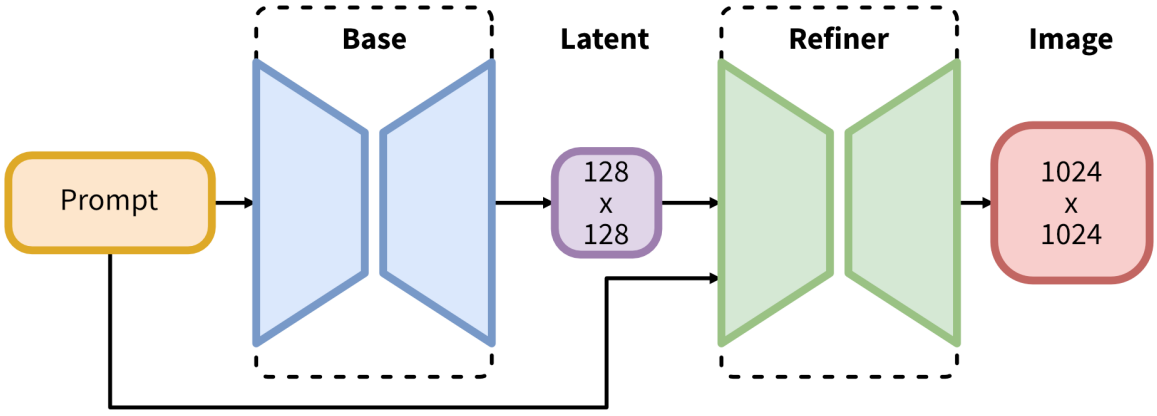


Figure 3: The Stable Diffusion XL pipeline with Base and Refiner models.

## 3.3 Vision-Language Alignment

### 3.3.1 Contrastive Learning Framework

Contrastive Learning is a self-supervised training paradigm designed to learn representation spaces in which semantically similar data pairs (positive pairs) are mapped closer together, while dissimilar pairs (negative pairs) are pushed further apart.

For multimodal text-image data, the standard objective is the contrastive loss, which operates on a batch of $N$ paired samples. The model maximizes the cosine similarity between the correct text-image pair $(v_I, v_T)$ located on the diagonal of the $N \times N$ similarity matrix, while minimizing similarity with off-diagonal non-matching samples. This formulation allows the model to learn a shared latent space where image and text embeddings are directly comparable.

Figure 4 illustrates the contrastive pre-training mechanism, which learns to match image and text pairs through large-scale contrastive learning. The figure is adapted from Radford et al [16].

### 3.3.2 CLIP Architecture

Contrastive Language-Image Pre-training [16] is the seminal model that scales the contrastive learning paradigm to hundreds of millions of image-text pairs. The architecture consists of two separate Transformer-based encoders: an image encoder that is a Vision Transformer which tokenizes images into visual patches, and a text encoder that is a Transformer language model which processes tokenized text sequences.

In our system, Contrastive Language-Image Pre-training functions as the feedback mechanism. Given an image embedding $v_I$ and a text embedding $v_T$, the model computes their cosine similarity:

$$\text{Score} = \frac{v_I \cdot v_T}{\|v_I\| \|v_T\|} \tag{8}$$

This score provides an objective measure of semantic alignment between the generated

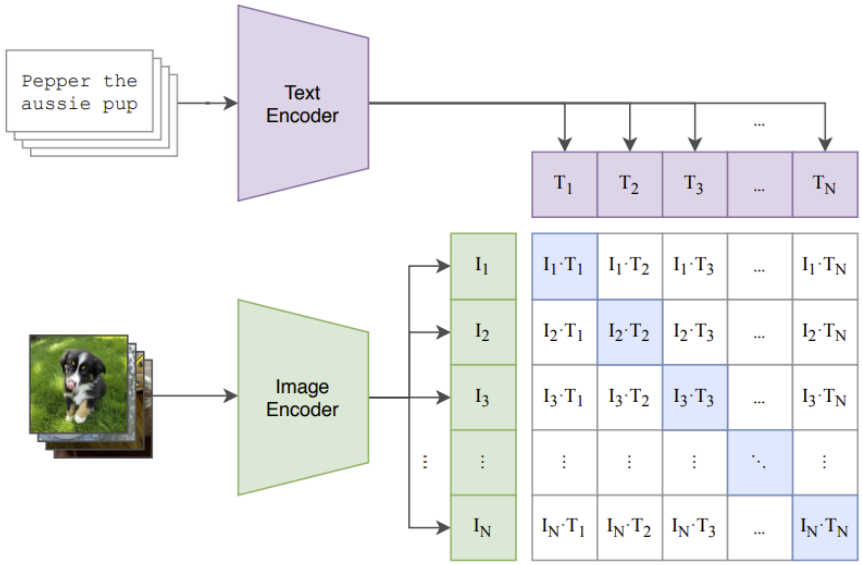


Figure 4: The Contrastive Pre-training mechanism of CLIP, learning to match image and text pairs.

image and the input prompt. Because the modular system lacks native multimodal attention (as found in unified models), Contrastive Language-Image Pre-training serves as an external evaluator that guides refinement by telling the reasoning module how well the current generation matches the intended semantics.

The importance of reliable alignment metrics is underscored by modern evaluation benchmarks such as GenEval [17] and related benchmarks, which assess text-image models for faithfulness, compositional consistency, and semantic correctness.

## 3.4 Chapter Summary

The preceding sections have systematized the fundamental principles of Large Language Models acting as reasoning agents and the mechanics of Latent Diffusion Models. Furthermore, we clarified the multimodal interaction mechanisms required to facilitate the reasoning-generation-feedback loop essential for our proposed solution. With the theoretical groundwork now established, the challenge lies in translating these principles into a tangible, functioning system. Chapter 4 presents the complete design and implementation of the Modular Interleaving Reasoning-Generation framework, detailing how components are integrated into a unified architecture.

# 4 Proposed Framework

Building upon the theoretical foundations established in Chapter 3, this chapter presents the complete design and implementation of our Modular Interleaving Reasoning-Generation framework. We synthesize the components discussed previously-Qwen 2.5-3B-Instruct for reasoning, Stable Diffusion XL for image synthesis, and Contrastive Language-Image Pre-training for visual feedback-into a cohesive architecture optimized for consumer-grade hardware constraints.

The chapter is structured to provide both architectural overview and implementation-specific details. This comprehensive methodology provides the blueprint for replicating our framework in educational and resource-constrained environments, directly addressing the accessibility gap identified in Chapters 1 and 2.

## 4.1 Overview of the Modular Framework

The process operates as follows. In the initial generation phase, the user sends a prompt to the language model module, which processes and generates initial reasoning about how to create the image. The language model sends the prompt with initial reasoning to the image generation module, which creates the first image. This image is sent to the vision-language encoder for encoding, which generates image features and returns them to the language model. In subsequent refinement iterations, the language model receives the features, reflects, and generates refinement feedback for improvement. The language model sends the previous image with refinement instructions to the image generation module, which uses image-to-image techniques to refine the output, creating an improved image. This improved image is again sent to the vision-language encoder for encoding, generating new features that return to the language model. This iteration process repeats for multiple cycles, with each cycle following the pattern of reasoning to image to feedback to refined reasoning. Finally, in the evaluation phase, the final image is sent to the evaluation module, which performs multiple measurements to assess quality and compositional accuracy. The final result with metrics is returned to the user.

The system iteratively refines images through multiple cycles rather than generating once. Each iteration allows the language model to learn from vision-language feedback to provide better guidance. The process typically completes after two to four iterations and provides comprehensive evaluation.

### 4.1.1 Architectural Motivation and Problem Formulation

Although recent advances in text-to-image generation have been driven largely by scaling diffusion models, a fundamental limitation persists: the single-shot generation paradigm. Models such as Stable Diffusion XL attempt to map a natural-language prompt directly to an image in a single forward pass. While this works for simple descriptions, the absence of intermediate reasoning leads to failures on compositionally demanding prompts. For example, in a query such as "a red cat next to a blue dog," single-shot models often exhibit attribute leakage-colors blend, swap, or diffuse inconsistently-reflecting the absence of explicit object-attribute binding and structural planning.

The Interleaving Reasoning-Generation paradigm proposed by Huang et al. [6] addresses this issue by integrating chain-of-thought reasoning directly into the generation loop. However, existing implementations rely on unified multimodal models that jointly process text and image tokens within a shared embedding space. These models typically contain over 14 billion parameters and require high-end computational infrastructure for both training and inference. Such requirements make them impractical for educational environments or consumer-grade deployments.

To address this accessibility barrier, we propose a modular adaptation of the framework. Our objective is to preserve the semantic and compositional benefits of iterative reasoning while operating under strict hardware constraints.

### 4.1.2 High-level Architecture: The Decoupled Approach

Instead of employing a monolithic unified model, we adopt a decoupled modular architecture consisting of three independent components. Models are dynamically loaded into memory only when needed, enabling efficient execution under limited resources.

The system comprises a reasoning agent responsible for decomposing abstract user prompts into explicit visual plans, correcting logic, and generating reflection feedback; a generator responsible for translating the language model's text instructions into pixel data; and an evaluator that assesses the alignment between the generated image and the text prompt, closing the loop by providing objective quality signals.

### 4.1.3 Design Philosophy: Multi-round Iterative Refinement

Our framework operates on the principle of iterative refinement. We conceptualize image generation not as a linear path, but as a recursive cycle of think, generate, reflect, and refine.

In this cycle, the generated image at step $t$ serves as visual grounding for step $t+1$. This allows the system to perform self-correction. For example, if the user requests "three apples" but the model generates two, a single-shot model has no way to fix this. In our framework, the feedback module detects the discrepancy, the reasoning module formulates

a correction ("I see only two apples; I need to add one more"), and the generation module executes this specific update in the next round.

## 4.2 Data Synthesis Pipeline

### 4.2.1 The Need for Synthetic Reasoning Data

A critical challenge in adapting interleaving reasoning is the lack of public training data. The original dataset, which contains high-quality reasoning traces, is proprietary. Standard image-caption datasets lack the intermediate reasoning logic-they only provide the final description, not the thought process required to create it. To overcome this, we developed a data synthesis pipeline. We curated a diverse set of prompts stratified across multiple difficulty levels, ranging from simple objects to complex multi-constraint instructions.

### 4.2.2 Chain-of-Thought Generation and Filtering

Since the original proprietary dataset contains detailed reasoning traces paired with real visual feedback signals and no public alternative exists, we construct a fully synthetic reasoning corpus tailored for our modular framework.

We design a rule-based generation engine that decomposes each prompt into a structured Chain-of-Thought. The templates cover object decomposition, spatial reasoning, attribute binding, lighting and scene layout, and style, color, and technical details. These templates ensure that the reasoning format matches the expected analysis-plan-reflection style used in our training.

Unlike datasets built from real images, our framework does not extract embeddings from vision models. Instead, we simulate vision-language-inspired quality descriptors using statistical profiles (mean, standard deviation, and range values) that reflect common trends found in real alignment scores. These synthetic descriptors do not attempt to replicate real image embeddings; rather, they provide structured signals (e.g., "underexposed", "overexposed", "low contrast"), quantitative cues that allow the language model to practice interpreting feedback, and consistent inputs enabling multi-round refinement during supervised training. This approach preserves the functional role of visual feedback while avoiding dependency on external image datasets.

To ensure data quality, we apply three filters. Reasoning traces shorter than 50 tokens or longer than 200 tokens are discarded to maintain clarity and consistency. A keyword-matching module verifies that every entity from the original prompt appears explicitly in the reasoning chain. Outputs are checked against a strict reasoning layout to prevent malformed or inconsistent traces.

The final corpus consists of approximately 4,000 high-quality synthetic reasoning and

descriptor pairs, suitable for fine-tuning the reasoning module without requiring access to real images or proprietary multimodal datasets.

## 4.3 System Component Implementation

### 4.3.1 Reasoning Module: Qwen 2.5-3B-Instruct

For the reasoning engine, we selected Qwen 2.5-3B-Instruct [18], a state-of-the-art small language model. This specific model choice is a strategic optimization for our hardware constraints. While larger models (7 billion or 14 billion parameters) offer broader world knowledge, they consume significant resources, leaving no room for the image generation model. The 3 billion parameter version provides an effective trade-off: it retains robust instruction-following capabilities sufficient for visual planning while consuming substantially less memory.

To specialize the model for visual reasoning without the cost of full-parameter training, we utilize Low-Rank Adaptation [19]. We freeze the pre-trained weights and inject low-rank decomposition matrices into the attention and feed-forward layers. This allows us to train the model on our synthetic corpus in approximately 5 hours on dual graphics processing units.

During fine-tuning, we load the model using 4-bit quantization (NormalFloat 4-bit format with double quantization) via the bitsandbytes library [20]. This technique compresses the model weights to 4-bit precision, reducing the memory footprint significantly. For inference, we load the Low-Rank Adaptation-merged model in 8-bit precision, which offers more stable generation quality compared to 4-bit while remaining memory-efficient.

### 4.3.2 Generation Module: Stable Diffusion XL

The image synthesis backbone is Stable Diffusion XL Base 1.0 [1], chosen for its superior UNet architecture and dual text encoders. The model is loaded in float16 precision to balance quality and memory usage.

A key technical decision in this framework is to disable attention slicing. Attention slicing is a technique often used to save memory by processing the attention matrix in chunks, which significantly slows down inference. Because we opted for the smaller Qwen 2.5-3B model (instead of a 7 billion or 13 billion parameter model), we successfully freed up enough memory headroom to keep the full Stable Diffusion XL attention mechanism in memory. By disabling slicing, we achieve a reduction in generation latency, making the multi-round process computationally feasible.

The module switches dynamically between Text-to-Image for the initial generation and Image-to-Image for subsequent refinement steps.

### 4.3.3 Feedback Module: Vision-Language Model

We distinguish between feedback (information guiding iterative refinement) and evaluation (metrics assessing final outputs). CLIP serves both roles: feature statistics provide feedback during generation, while alignment scores measure final quality.

To enable iterative refinement, we employ CLIP (ViT-B/32) [16] to encode generated images into a 512-dimensional feature space. For each image $I_t$ at iteration $t$, we extract three summary statistics from the CLIP embedding: mean $\mu$, standard deviation $\sigma$, and maximum $f_{\max}$. These statistics are formatted as structured text (e.g., "mean=0.215, std=0.095") and provided to the language model, which has been fine-tuned to interpret them as diagnostic feedback (e.g., "underexposed, increase exposure"). This approach avoids using CLIP alignment scores directly as feedback, which would create circular reasoning by optimizing the same metric used for evaluation.

## 4.4 Multi-round Iterative Pipeline Design

Figure 5 shows our Interleaving Reasoning-Generation system architecture. The diagram illustrates the complete workflow from user prompt input through multiple reasoning-generation-feedback cycles to final evaluation.

### 4.4.1 The Iterative Workflow

The inference pipeline is designed as a fixed cycle of 4 iterations, ensuring a consistent evaluation protocol. The workflow follows this precise sequence.

In iteration 1 (initial generation), the reasoning module analyzes the user prompt and generates an initial reasoning plan (chain-of-thought decomposition). The generation module executes Text-to-Image generation using the original prompt with the reasoning plan. The feedback module computes similarity score between generated image and prompt, and converts to textual feedback. The system stores the image, score, and feedback.

In iteration 2 (major correction), the reasoning module receives the original prompt with previous reasoning, feedback, and score, then generates refinement instructions focusing on structural corrections. The generation module executes Image-to-Image refinement using the previous image as input. The feedback module computes new similarity score and textual feedback. The system stores the new image, score, and feedback.

In iterations 3–4 (progressive refinement), the reasoning-generation-feedback cycle repeats. Denoising strength decreases to preserve established composition. Guidance scale increases to enforce stricter prompt adherence. Focus shifts from structural corrections to detail enhancement and aesthetic polishing.

Importantly, feedback from iteration $i$ guides the refinement instructions for iteration $i+1$. The final image is from iteration 4, but all intermediate images and scores are saved for analysis.

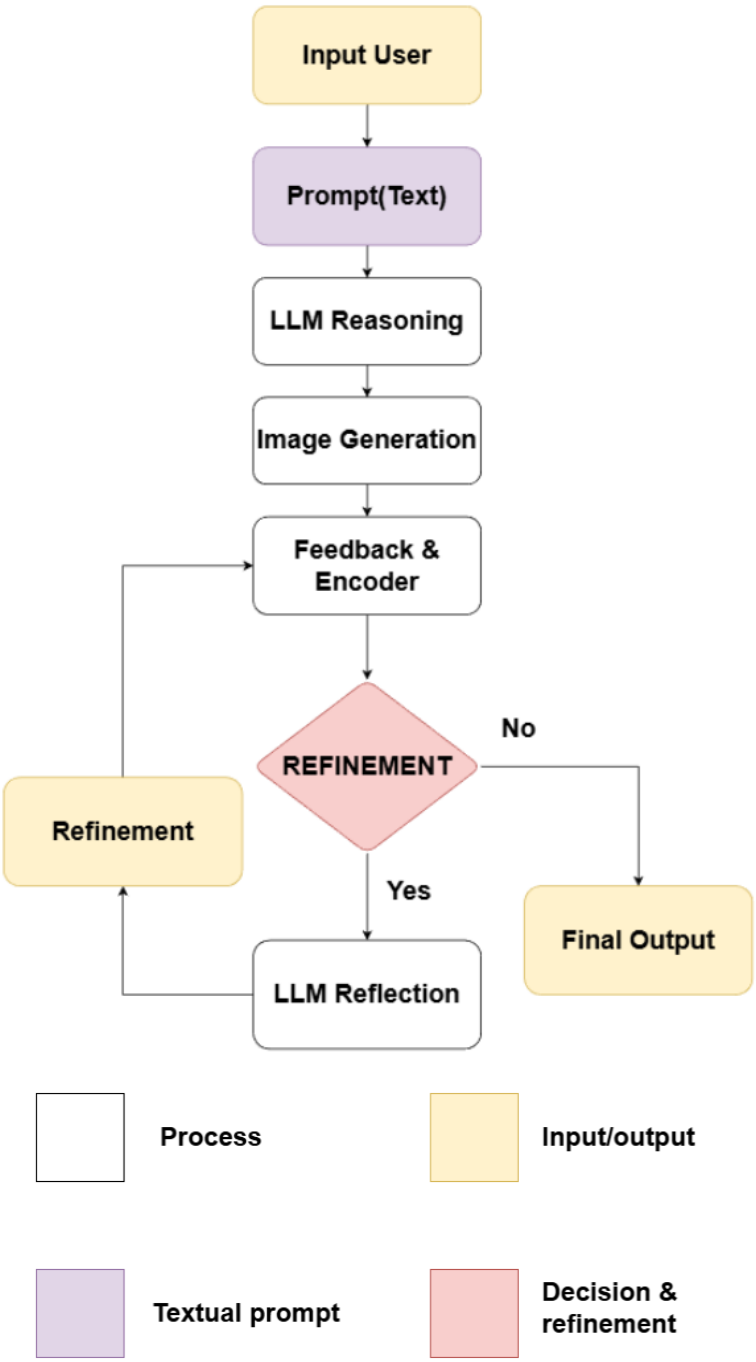


Figure 5: Our Interleaving Reasoning-Generation system architecture illustrating the iterative workflow.

### 4.4.2 Adaptive Parameter Scheduling

A critical design choice in our pipeline is the use of adaptive parameters for image-to-image refinement. Iterative generation risks over-refinement, where repeated processing degrades correct elements or introduces artifacts. To mitigate this, we dynamically adjust two key hyperparameters: denoising strength $S$ and guidance scale $G$.

Denoising strength controls the degree of modification applied to the input image, where $S = 1.0$ corresponds to complete re-generation from noise and $S = 0.0$ preserves the input unchanged. We employ a decay schedule:

$$S_i = \max\left(0.3, \min\left(0.7, 0.7 - 0.2(i-2)\right)\right) \tag{9}$$

where $i$ denotes the iteration index. The strength begins at $S_2 = 0.7$ to permit substantial structural modifications (e.g., object repositioning), then decays to $S_4 = 0.3$ to restrict changes to fine-grained adjustments while preserving the established composition.

Classifier-free guidance controls the trade-off between prompt adherence and output diversity. We use an increasing schedule:

$$G_i = 7.5 + 0.75(i-2) \tag{10}$$

The guidance scale increases from $G_2 = 7.5$ to $G_4 = 9.0$. As the image converges toward the target, higher guidance enforces stricter adherence to refinement instructions, reducing stochastic variance and sharpening details.

## 4.5 Chapter Summary

The architectural design and execution flow of the Modular Interleaving Reasoning-Generation system have been detailed herein. We described the synthesis of Qwen 2.5-3B-Instruct, Stable Diffusion XL, and a vision-language model into a cohesive loop, specifically optimized to operate efficiently on consumer-grade hardware, thereby addressing the accessibility constraints discussed earlier in the report. Having finalized the system design and implementation, the next logical step is to validate its efficacy through rigorous testing. Chapter 5 provides a comprehensive experimental evaluation, comparing the performance of our proposed model against established baselines to demonstrate its practical utility.

# 5 Experimental Setup & Results

Having established the architecture and implementation details of the proposed Modular Interleaving Reasoning-Generation framework in Chapter 4, the focus now shifts to its empirical validation. This chapter provides a comprehensive description of the experimental framework established to evaluate the proposed system. We begin by detailing the hardware specifications, component models, and critical hyperparameter configurations used to ensure reproducibility. Subsequently, we present a rigorous analysis of the system's performance through both quantitative evaluation-utilizing metrics such as alignment scores, perceptual distance, and compositional accuracy-and qualitative analysis of visual outputs. The empirical evidence presented herein confirms the analytical nature of this investigation, demonstrating when and why the multi-round Interleaving Reasoning-Generation architecture succeeds or fails.

## 5.1 Environment and Hardware

To demonstrate the accessibility of the proposed Modular framework for educational and resource-constrained research, all experiments were conducted on the Kaggle Notebooks free tier platform. This choice imposes strict hardware limitations, necessitating efficient resource management.

The evaluation dataset consists of 150 text prompts. These prompts were manually curated to cover a range of difficulty levels, including single-object descriptions (e.g., "a photo of a cat"), simple pairwise compositions (e.g., "a cat and a dog"), and more complex multi-constraint scenes. This design ensures that the benchmark systematically probes counting ability, spatial relationships, and attribute binding, aligning with the core failure modes discussed in Chapter 1.

The hardware specifications include dual NVIDIA Tesla T4 graphics processing units based on the Turing architecture, paired with an Intel Xeon central processing unit (4 cores) and system random access memory. The software environment utilized Ubuntu 20.04.5 LTS, Python 3.10, and PyTorch 2.1.0 with CUDA 12.1.

Table 1 summarizes the hardware and environment specifications used for all experiments. To fit the pipeline within the memory budget, we employed specific optimization strategies. The reasoning module (Qwen 2.5-3B-Instruct) is loaded in 8-bit quantization, consuming approximately 4.0–4.5 GB. The generation module (Stable Diffusion XL Base) is loaded in FP16 precision, consuming approximately 7.8 GB. The feedback module consumes approximately 1.9 GB. The remaining capacity handles buffers and temporary

Table 1: Hardware and Environment Specifications

| Component | Specification | Details |
|---|---|---|
| Graphics Processing Unit | 2× NVIDIA Tesla T4 | 8GB memory each (16GB total) |
| Architecture | Turing (SM 7.5) | Tensor Cores available |
| CUDA Cores | 2×2560 = 5120 | Parallel processing |
| Memory Bandwidth | 2×320 GB/s | Data transfer speed |
| Central Processing Unit | Intel Xeon (4 cores) | Background processing |
| System Memory | 30GB DDR4 | System memory |
| Storage | 73GB SSD | Temporary session storage |
| Session Limit | 12 hours | Auto-timeout constraint |

tensors.

Given the strict 12-hour session timeout imposed by the Kaggle platform and the estimated cumulative evaluation time of approximately 5 hours for 150 prompts, a robust session management strategy was implemented to ensure data integrity. Prompts were executed in discrete batches to manage memory overhead effectively. To mitigate the risk of data loss from potential interruptions, we established an automated checkpointing mechanism that serialized intermediate results to persistent storage periodically, enabling seamless resumption from the last stable state. Furthermore, a priority queue approach was utilized to process computationally intensive complex prompts first, ensuring that the most critical evaluation data was secured prior to any potential resource exhaustion or timeout.

## 5.2 Models Used

### 5.2.1 Baseline System

To establish a rigorous performance floor for comparison, we utilize Stable Diffusion XL Base 1.0 operating in a standard single-shot configuration without any reasoning capabilities. This baseline functions exclusively in Text-to-Image mode, employing the same parameter settings as the first iteration of our proposed method (15 inference steps, guidance scale of 7.5). As an industry-standard diffusion model, this setup represents the typical behavior of current text-to-image systems, characterized by direct prompt-to-image generation and a lack of intermediate compositional reasoning.

### 5.2.2 Proposed System: Multi-round Interleaving Reasoning

Our proposed framework implements the interleaving reasoning paradigm through a modular architecture comprising three distinct, interactively coupled components.

The cognitive core is powered by Qwen 2.5-3B-Instruct, optimized via 4-bit quantization for memory efficiency. This model was fine-tuned using Low-Rank Adaptation on a spe-

cialized corpus of 4,000 reasoning samples. The training configuration employed rank 8, alpha 16, and a dropout rate of 0.1, requiring approximately 3–4 hours over 3 epochs with an effective batch size of 8.

We employ Stable Diffusion XL Base 1.0 in FP16 precision as the visual synthesis engine. The generation pipeline operates dynamically: it initiates with a Text-to-Image process for the first iteration, then transitions to Image-to-Image for iterations 2 through 4. Crucially, we implement an adaptive parameter schedule where denoising strength decreases [0.7 $\rightarrow$ 0.6 $\rightarrow$ 0.5] to preserve structural integrity, while guidance scale increases [7.5 $\rightarrow$ 8.25 $\rightarrow$ 9.0] to enforce stricter prompt adherence in later stages.

To close the reasoning loop, we utilize a vision-language model. This module serves a dual purpose: quantifying text-image alignment through scores in the range [0,1] and providing descriptive features that guide the reasoning module's reflection process.

Table 2 presents the iteration configurations for the proposed system.

Table 2: Iteration Configuration Parameters

| Iteration | Type | Steps | Strength | Guidance | Focus |
|---|---|---|---|---|---|
| 1 (Base) | Txt2Img | 15 | N/A | 7.5 | Initial generation |
| 2 | Img2Img | 20 | 0.7 | 7.5 | Major error correction |
| 3 | Img2Img | 20 | 0.6 | 8.25 | Detail refinement |
| 4 | Img2Img | 20 | 0.5 | 9.0 | Quality polish |

## 5.3 Evaluation Procedure

To systematically assess the efficacy of the Modular framework, we implemented a rigorous evaluation pipeline. This procedure ensures a fair comparison across different iteration depths and guarantees the reproducibility of results.

### 5.3.1 Pipeline Execution

For each of the 150 test prompts, the evaluation protocol is designed to be exhaustive, executing a multi-stage generation and assessment sequence to capture the full trajectory of model performance. The process is divided into four distinct phases.

Initially, a single-shot image is synthesized using the standard Stable Diffusion XL configuration (identical to iteration 1 settings). This step serves as a critical control sample, establishing a performance floor that allows us to quantify exactly how much value the reasoning module adds compared to a traditional generation approach.

To understand the dynamics of iterative refinement, the full modular pipeline is executed at three specific depths: 2 iterations focuses on assessing the impact of initial error correction, specifically targeting gross compositional failures like object counting and spatial

positioning; 3 iterations evaluates intermediate refinement, testing the stability of the model as it attempts to enhance details without losing the original semantic structure; 4 iterations analyzes the effects of maximum polish, aiming to identify the point of diminishing returns or potential over-refinement, where excessive processing might degrade image fidelity.

Unlike standard procedures that might only evaluate the final output, our pipeline computes the key metrics for every intermediate and final image immediately upon generation. This granular data collection allows for step-by-step performance tracking.

### 5.3.2 Detailed Evaluation Protocol

To ensure reproducibility and transparency, we provide complete specifications of our evaluation protocol.

The evaluation dataset consists of 150 prompts sourced from a custom-curated collection based on common object classes with compositional complexity. The categories include single-object (20 prompts), multi-object (20 prompts), counting tasks (20 prompts), spatial relations (20 prompts), color and multi-attribute (20 prompts), and instruction-based scenarios (50 prompts). The prompt length has a mean of 8.4 words (range: 4–15 words). Prompt complexity is ensured through stratified sampling to ensure balanced distribution across difficulty levels.

The inference configuration for initial generation (text-to-image, iteration 1) uses a resolution of 512×512 pixels, a specific sampler, 15 inference steps, guidance scale of 7.5, a negative prompt to avoid common artifacts, and fixed seed per prompt (Seed = 42 + prompt index).

For refinement rounds (image-to-image, iterations 2–4), the configuration maintains a resolution of 512×512 pixels (preserved), the same sampler, 20 inference steps (increased for refinement quality), adaptive guidance scale (iteration 2: 7.5, iteration 3: 8.25, iteration 4: 9.0), adaptive strength (iteration 2: 0.7 (high modification), iteration 3: 0.6 (moderate modification), iteration 4: 0.5 (gentle refinement)), and the same fixed seed maintained across all iterations for a given prompt.

The seed strategy uses a fixed seed per prompt (deterministic) with the formula seed = 42 + prompt index (where prompt index is in the range [0, 149]). This ensures exact reproducibility while providing variation across different prompts. The system performs one run per prompt (deterministic seeding eliminates need for multiple runs).

The language model reasoning configuration uses Qwen 2.5-3B-Instruct (fine-tuned with Low-Rank Adaptation), 8-bit quantization, maximum new tokens of 150, temperature of 0.7 (balanced creativity and consistency), top-p of 0.9, and nucleus sampling enabled.

The evaluation metrics configuration includes alignment score computed using cosine similarity between image and text embeddings, aesthetic score based on a custom heuristic computed from color diversity, brightness, and contrast, a simplified faithfulness metric

based on caption generation and word overlap scoring, perceptual distance using deep features computed against a validation subset, and compositional accuracy using object detection with confidence threshold of 0.7.

### 5.3.3 Computational Cost and Efficiency

The average processing latency is approximately 210 seconds per prompt for a full 4-iteration cycle. Consequently, the total execution time for the 150-prompt dataset is estimated at 31500 seconds (approximately 8.75 hours). With an additional 0.5 hours allocated for metric computation overhead, the total experiment duration is approximately 10 hours. This efficiency is critical, as it allows the complete evaluation to fit comfortably within the Kaggle platform's 12-hour session limit.

### 5.3.4 Reproducibility Protocols

To ensure the highest standards of scientific rigor and facilitate the independent verification of our findings, we enforced a strict set of reproducibility protocols throughout the experimental phase. First, to eliminate stochastic variability and guarantee deterministic behavior, a fixed global random seed (Seed=42) was universally applied across all computational modules, including PyTorch, NumPy, and Diffusers; this ensures that identical input prompts consistently yield the exact same reasoning trajectories and visual outputs across different runs. Second, to mitigate the risks associated with the prolonged execution time on the Kaggle platform, we implemented a robust checkpointing strategy wherein intermediate results were serialized to persistent storage periodically, thereby safeguarding against potential data loss in the event of unexpected session timeouts.

## 5.4 Results and Analysis

In this section, we evaluate the system's performance using a suite of quantitative metrics-including alignment scores, aesthetic scores, faithfulness scores, perceptual distance, and compositional accuracy-and validate these results through qualitative visual analysis. The primary focus of this analysis is to quantify the impact of iterative reasoning on compositional accuracy, semantic alignment, and visual fidelity, as well as to identify optimal stopping points for the refinement process.

### 5.4.1 Quantitative Evaluation

**Alignment Score (Semantic Alignment)** To assess the semantic alignment between the generated images and their corresponding text prompts, we utilized an alignment score based on vision-language embeddings. As detailed in Table 3, the results indicate a complex, non-linear progression across iterations.

Table 3: Alignment Score Performance across Iterations

| Variant | Mean Score | vs. Base | Interpretation |
|---|---|---|---|
| Base SDXL | 0.2652 | — | Baseline |
| IRG 2-iter | 0.2638 | -0.53% | Slight degradation |
| IRG 3-iter | 0.2591 | -2.30% | Moderate degradation |
| IRG 4-iter | 0.2528 | -4.68% | Significant degradation |
| SDXL 2-round (no reasoning) | 0.2605 | -1.77% | Baseline degradation |
| SDXL 3-round (no reasoning) | 0.2539 | -4.26% | Stronger degradation |
| SDXL 4-round (no reasoning) | 0.2458 | -7.31% | Severe degradation |

Contrary to initial expectations, alignment scores exhibit monotonic degradation across all refinement iterations for both interleaving reasoning and baseline variants. The multi-round image-to-image process introduces semantic drift, where repeated denoising at high strength progressively blurs the alignment between generated images and original prompts. Critically, interleaving reasoning consistently outperforms the no-reasoning baseline at each iteration count (2-iter: 0.2638 vs 0.2605, 3-iter: 0.2591 vs 0.2539, 4-iter: 0.2528 vs 0.2458), demonstrating that language model reasoning provides value in maintaining semantic coherence despite the image-to-image degradation. This finding suggests that while multi-round refinement is inherently harmful to alignment, reasoning helps mitigate-but cannot fully prevent-this decline.

Figure 6 illustrates the monotonic degradation of semantic alignment (alignment score) across refinement iterations. Both interleaving reasoning and no-reasoning variants show progressive decline, with interleaving reasoning consistently outperforming the no-reasoning baseline at each iteration count. The best alignment performance is achieved by the single-pass Base Stable Diffusion XL.

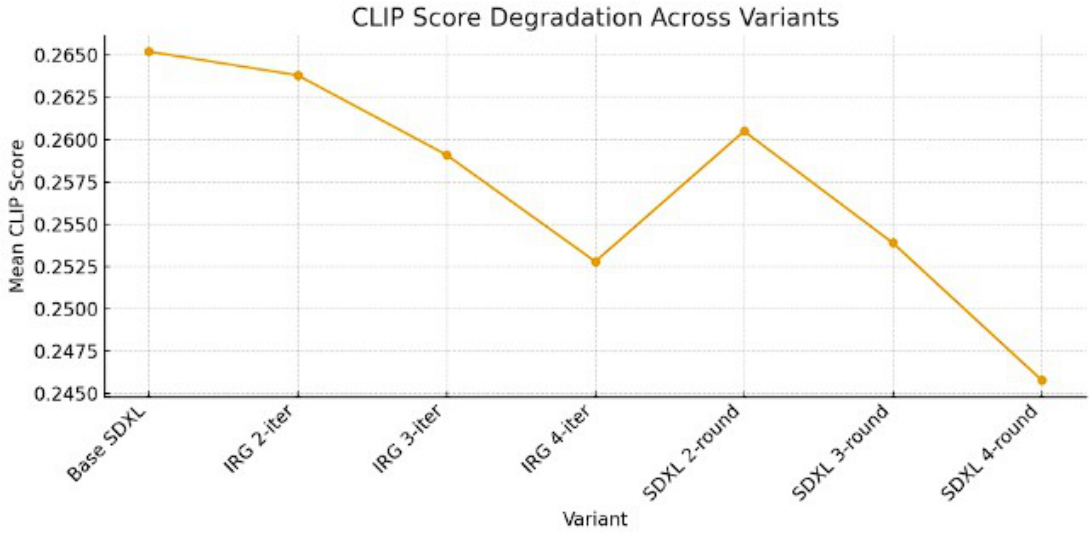


Figure 6: Monotonic degradation of semantic alignment across refinement iterations.

**Aesthetic Score (Visual Quality)** In contrast to semantic alignment, visual quality metrics exhibited a consistent monotonic improvement throughout the refinement process. Table 4 presents the aesthetic score performance.

Table 4: Aesthetic Score Performance

| Variant | Mean Aesthetic | vs. Base | Interpretation |
|---|---|---|---|
| Base SDXL | 9.336 | — | Baseline |
| IRG 2-iter | 9.410 | +0.79% | Slight improvement |
| IRG 3-iter | 9.525 | +2.02% | Moderate improvement |
| IRG 4-iter | 9.624 | +3.08% | Strong improvement |
| SDXL 2-round (no reasoning) | 9.457 | +1.30% | Baseline improvement |
| SDXL 3-round (no reasoning) | 9.622 | +3.06% | Stronger improvement |
| SDXL 4-round (no reasoning) | 9.715 | +4.06% | Best aesthetic |

The aesthetic score exhibits monotonic improvement across all refinement iterations for both interleaving reasoning and baseline variants, with the strongest gains at 4 rounds (+3.08% for interleaving reasoning, +4.06% for no-reasoning baseline). Interestingly, the no-reasoning baseline slightly outperforms interleaving reasoning at each iteration count (2-round: 9.457 vs 9.410, 3-round: 9.622 vs 9.525, 4-round: 9.715 vs 9.624). This suggests that language model reasoning introduces minor perturbations that slightly reduce visual polish, though the difference is marginal ($< 0.5\%$).

Figure 7 shows the monotonic improvement in visual quality (aesthetic score). Unlike compositional metrics, aesthetic quality consistently improves with refinement iterations for both interleaving reasoning (+3.08% at 4-iter) and no-reasoning baseline (+4.06% at 4-round). The no-reasoning baseline achieves slightly higher aesthetic scores.


Figure 7: Monotonic improvement in visual quality (Aesthetic Score).

**Faithfulness Score (Prompt Faithfulness)** We employ a simplified caption-based surrogate for prompt faithfulness evaluation, which we term faithfulness-lite. Unlike the original faithfulness metric, which uses Visual Question Answering to verify attribute grounding, our implementation measures lexical overlap between generated captions and the original prompt. Due to computational constraints (Visual Question Answering models require significant additional memory), we use this caption-based proxy. This simplified approach does not fully capture the original protocol's attribute-level faithfulness verification.

The faithfulness metric showed a counter-intuitive decline across all iterations. Table 5 presents the results.

Table 5: Faithfulness Score Performance

| Variant | Mean Score | vs. Base | Interpretation |
|---|---|---|---|
| Base SDXL | 0.6485 | — | Baseline |
| IRG 2-iter | 0.6490 | +0.08% | Minimal change |
| IRG 3-iter | 0.6226 | -3.99% | Moderate degradation |
| IRG 4-iter | 0.5750 | -11.33% | Significant degradation |
| SDXL 2-round (no reasoning) | 0.6535 | +0.77% | Best faithfulness |
| SDXL 3-round (no reasoning) | 0.6059 | -6.57% | Moderate degradation |
| SDXL 4-round (no reasoning) | 0.5732 | -11.61% | Severe degradation |

Faithfulness scores follow a similar inverted U-curve pattern as compositional accuracy, with only the 2-iteration variants maintaining baseline performance (interleaving reasoning: +0.08%, no-reasoning: +0.77%). The degradation at higher iterations (-11.33% for interleaving reasoning 4-iter, -11.61% for no-reasoning 4-round) appears to be caused by vocabulary drift combined with object blending artifacts. As the reasoning module adds descriptive details and the image-to-image process blurs object boundaries, the generated captions diverge lexically from the original concise prompts, resulting in lower word-overlap scores.

Figure 8 shows the trend of faithfulness score (prompt faithfulness) across iterations. The consistent decline is hypothesized to be an artifact of vocabulary drift, where iterative reasoning adds descriptive details that reduce lexical overlap with the original prompt, despite preserving semantic meaning. Note that faithfulness-lite is a caption-based surrogate and does not reflect the full attribute-grounding capabilities of the original Visual Question Answering-based metric.

**Perceptual Distance (Distribution Quality)** The Fréchet Inception Distance results reveal a critical insight regarding image realism and distributional alignment with real-world data. Table 6 presents the perceptual distance analysis.

The perceptual distance scores (computed against Base Stable Diffusion XL distribution)

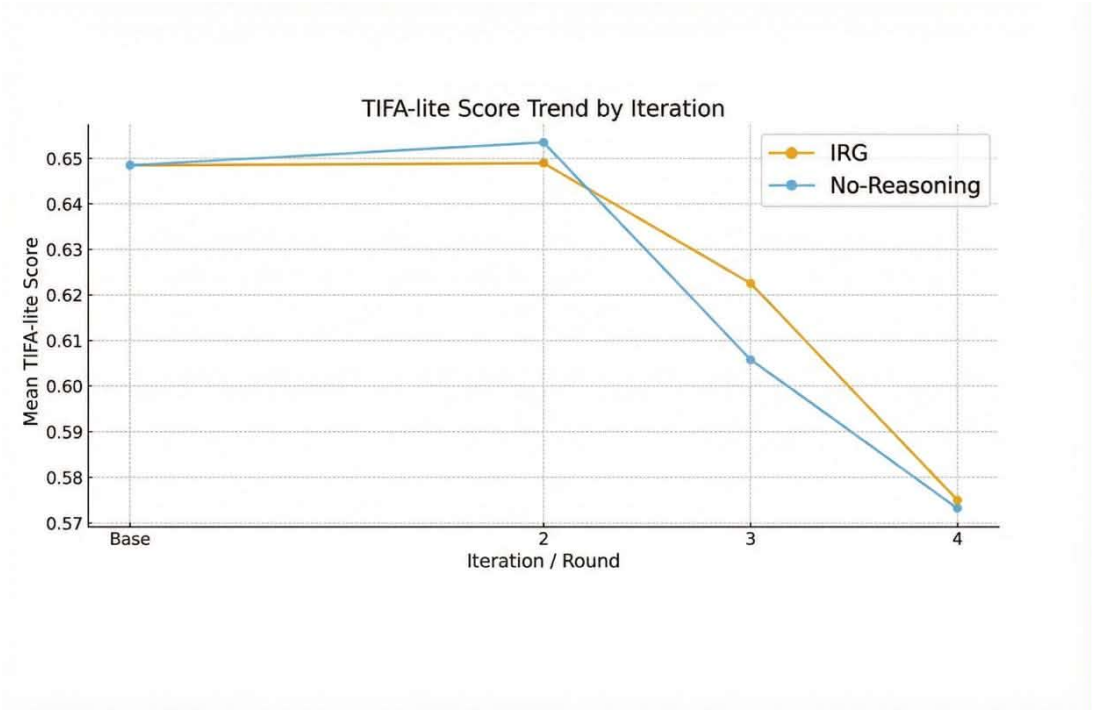


Figure 8: Trend of faithfulness score across iterations.

Table 6: Perceptual Distance Analysis

| Variant | FID Score (vs Base SDXL) | Interpretation |
|---|---|---|
| Base SDXL | N/A (reference) | Reference distribution |
| IRG 2-iter | 155.88 | Best IRG variant |
| IRG 3-iter | 171.80 | Moderate divergence |
| IRG 4-iter | 185.61 | Significant divergence |
| SDXL 2-round (no reasoning) | 148.19 | Best overall FID |
| SDXL 3-round (no reasoning) | 169.84 | Moderate divergence |
| SDXL 4-round (no reasoning) | 193.16 | Severe divergence |

reveal that all multi-round refinement variants diverge significantly from the baseline distribution (FID 148–193), with the no-reasoning baseline slightly outperforming interleaving reasoning at each iteration count (2-round: 148.19 vs 155.88, 3-round: 169.84 vs 171.80, 4-round: 193.16 vs 185.61). The consistently high perceptual distance scores indicate substantial distribution shift from the baseline model. This supports the over-refinement hypothesis: excessive image-to-image iterations create progressively more stylized images with boosted contrast and saturation that diverge from the baseline distribution. The advantage of the no-reasoning baseline in perceptual distance scores suggests that language model reasoning introduces additional perturbations that slightly increase distribution shift.

Figure 9 shows perceptual distance scores computed against Base Stable Diffusion XL distribution. All multi-round refinement variants show significant distribution shift (FID 148–193), with the no-reasoning baseline achieving slightly lower FID than interleaving reasoning at each iteration count. The consistently high FID values indicate progressive stylization away from the baseline distribution.

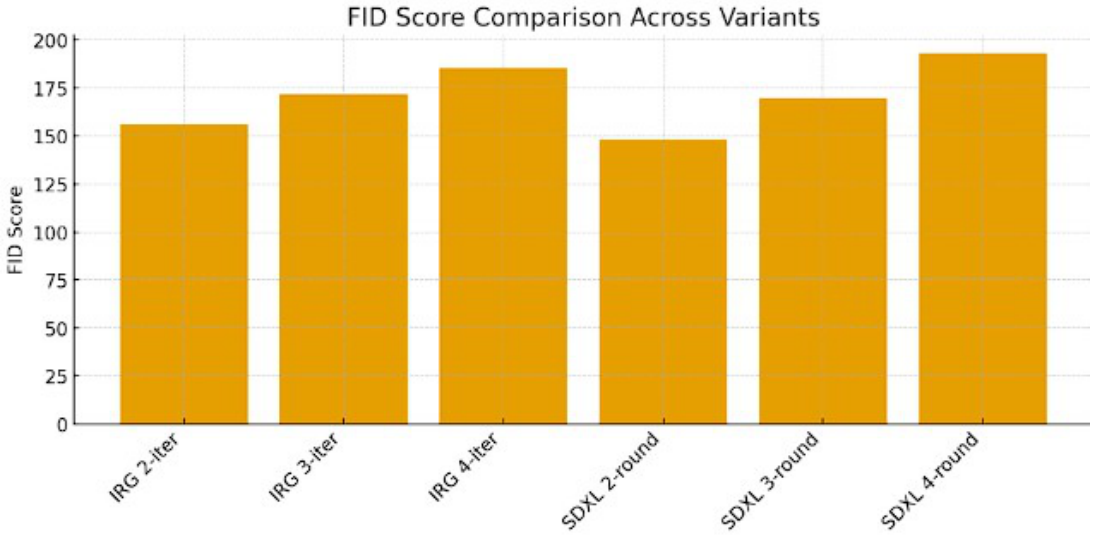


Figure 9: Perceptual distance scores computed against Base SDXL distribution.

**Compositional Accuracy** The most significant finding of this study is captured by the compositional accuracy metric, an object detection-based composite score measuring counting, confidence, and diversity.

We observed a pronounced inverted U-curve with only interleaving reasoning 2-iteration showing improvement over baseline (+7.74%, from 0.3497 to 0.3768). All other variants-including the no-reasoning baseline-degrade performance. Critically, interleaving reasoning consistently outperforms the no-reasoning baseline at each iteration count (2-iter: 0.3768 vs 0.3232, 3-iter: 0.3472 vs 0.2638, 4-iter: 0.2876 vs 0.1708), with gaps widening dramatically at higher iterations. This demonstrates that language model reasoning

provides substantial value in maintaining compositional structure during multi-round refinement, even as both variants degrade. The severe degradation of no-reasoning variants (-7.59% at 2-round, -51.17% at 4-round) confirms that multi-round image-to-image refinement without reasoning is highly detrimental to compositional accuracy.

Figure 10 shows that compositional accuracy follows an inverted U-curve, with only interleaving reasoning 2-iteration improving over baseline. Interleaving reasoning consistently outperforms the no-reasoning variants, whose accuracy collapses sharply at higher iteration counts.

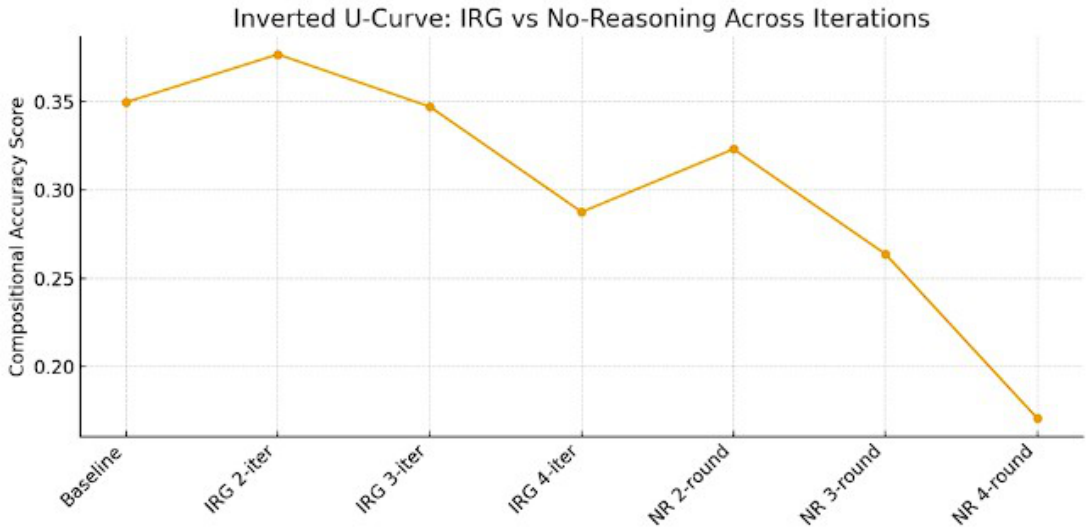


Figure 10: Compositional accuracy follows an inverted U-curve, with only IRG 2-iteration improving over baseline.

### 5.4.2 Aggregate Statistics Summary

The results present a clear conflict where different metrics optimize at different iteration counts, indicating that a single "best" setting does not exist for all objectives.

Table 7 presents a comprehensive summary of all metrics across all variants.

Table 7: Comprehensive Results Summary

| Metric | Base | IRG 2 | IRG 3 | IRG 4 | NoR 2 | NoR 3 | NoR 4 | Optimal |
|---|---|---|---|---|---|---|---|---|
| Alignment ↑ | 0.2652 | 0.2638 | 0.2591 | 0.2528 | 0.2605 | 0.2539 | 0.2458 | No refinement |
| Aesthetic ↑ | 9.336 | 9.410 | 9.525 | 9.624 | 9.457 | 9.622 | 9.715 | Max iter (NoR) |
| Faithfulness ↑ | 0.6485 | 0.6490 | 0.6226 | 0.5750 | 0.6535 | 0.6059 | 0.5732 | NoR 2-round |
| FID ↓ | N/A | 155.88 | 171.80 | 185.61 | 148.19 | 169.84 | 193.16 | NoR 2-round |
| Compositional ↑ | 0.3497 | 0.3768 | 0.3472 | 0.2876 | 0.3232 | 0.2638 | 0.1708 | IRG 2-iter |

Based on these findings, we identify the need for an adaptive iteration routing strategy. For tasks requiring compositional accuracy (counting, spatial), the system should stop

at iteration 2. Conversely, for tasks prioritizing visual polish or semantic elaboration (artistic, style), continuing to iteration 4 yields better results.

### 5.4.3 Qualitative Evaluation

Visual inspection of the generated samples across 150 prompts corroborates the quantitative data, highlighting specific behavioral patterns across different prompt categories. For counting tasks, Prompt 42, "a photo of four horses", demonstrates interleaving reasoning's capability to partially correct counting errors. While the baseline generated 5 horses, iteration 2 successfully moved closer to the target count. Figure 11 shows this visual comparison across all 7 variants.


Figure 11: Visual comparison of counting task success.

For spatial relationships, Prompt 68, "a cat lying below a window", showcases interleaving reasoning's effectiveness in correcting spatial arrangements. The baseline often misplaced objects or failed to establish the vertical relationship. Iteration 2 successfully positioned the cat in the correct spatial relation to the window. Among top alignment improvements, Prompt 77 ("a cat on top of a stack of books") achieved +29.44%, further validating spatial reasoning benefits. Figure 12 shows this comparison.

For color attribution, Prompt 85, "a photo of a pink teddy bear", illustrates the reasoning module's success in fixing attribute binding. The baseline occasionally generated brown or beige teddy bears, while iteration 2 correctly applied the pink color. The highest alignment improvement observed was Prompt 141 ("a white dog lying beside a blue sofa") at +56.58%, demonstrating strong gains in color-spatial compositional tasks. Figure 13 shows this comparison.

Analysis of alignment score improvements reveals that interleaving reasoning excels at complex compositional prompts combining spatial and color attributes. The top 5 improvements ranged from +27.67% to +56.58%, with prompts like "put a green backpack

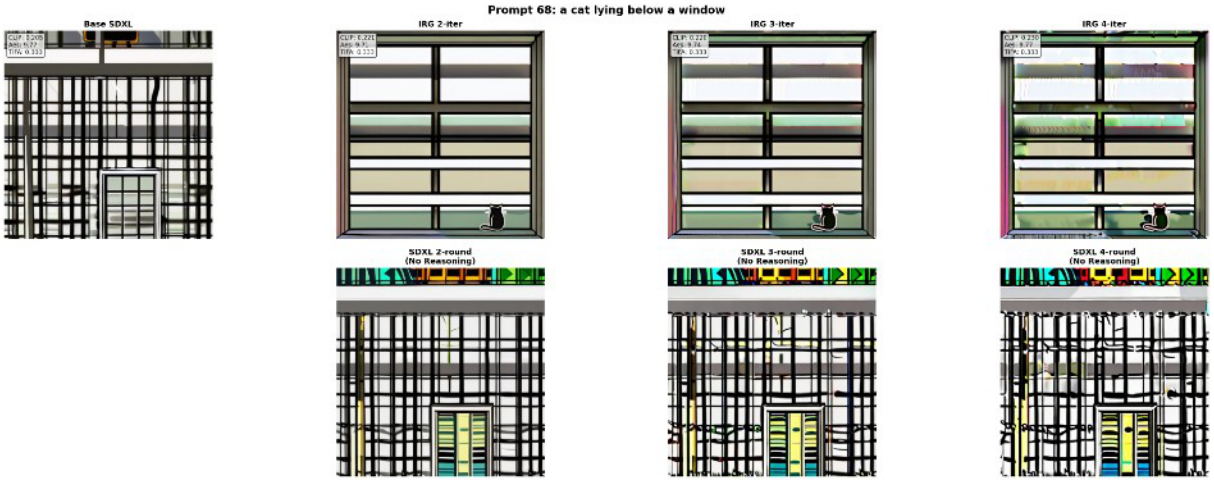


Figure 12: Visual comparison of spatial reasoning success.

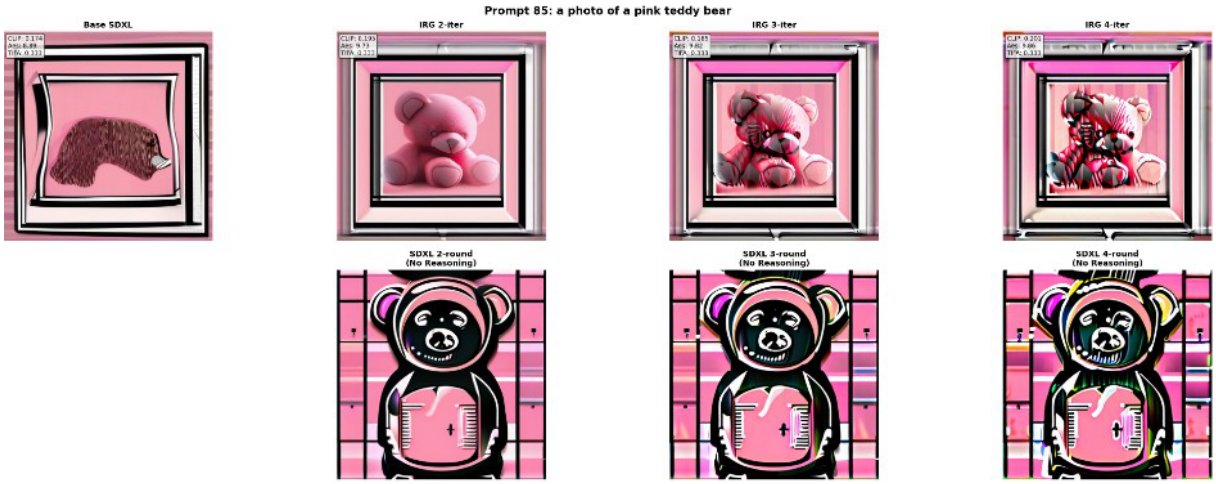


Figure 13: Visual comparison of color attribution success.

under a chair in the room" (Prompt 140, +34.03%) and "a smartphone next to a laptop on a shelf" (Prompt 113, +27.67%) showcasing successful multi-object spatial reasoning.

### 5.4.4 Failure Case Analysis

To provide deeper insight into system limitations, we analyze specific failure patterns observed across difficult prompt categories.

For object hallucination in counting and multi-object tasks, Prompt 29 ("a cat and a laptop") demonstrates failure. The baseline generated cat correctly with laptop, while interleaving reasoning 2-iteration generated just the cat. Interleaving reasoning 4-iteration showed over-refinement following the wrong case. The failure mode is that the language model suffers from over-refinement and semantic drift, causing loss of critical prompt elements (the laptop) in later interleaving reasoning iterations. Figure 14 shows this comparison.

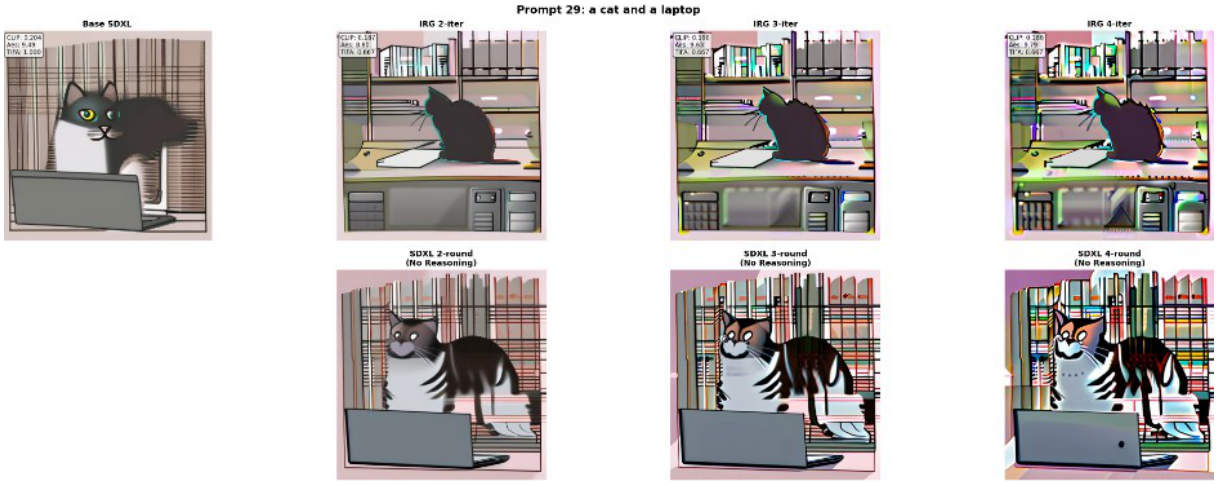


Figure 14: Visual comparison of Object Hallucination in Counting and Multi-Object Tasks.

For spatial relation confusion, Prompt 61 ("a dog under a table") shows another failure mode. The baseline positioned the dog correctly under the table, while interleaving reasoning 2-iteration attempted improvement but resulted in the dog being omitted. Interleaving reasoning 4-iteration followed an incorrect refinement path. The failure mode is that over-refinement during iterative prompt refinement causes object omission and spatial relationship loss, leading to semantic drift across iterations. Figure 15 illustrates this.

For over-refinement and detail loss, Prompt 70 ("a bottle on top of a wooden table") demonstrates progressive degradation. The baseline showed clear bottle outline with distinct wooden texture. Interleaving reasoning 2-iteration enhanced lighting and texture detail (successful refinement). Interleaving reasoning 4-iteration showed bottle edges becoming soft and blurred, with wooden texture over-smoothed. The failure mode is

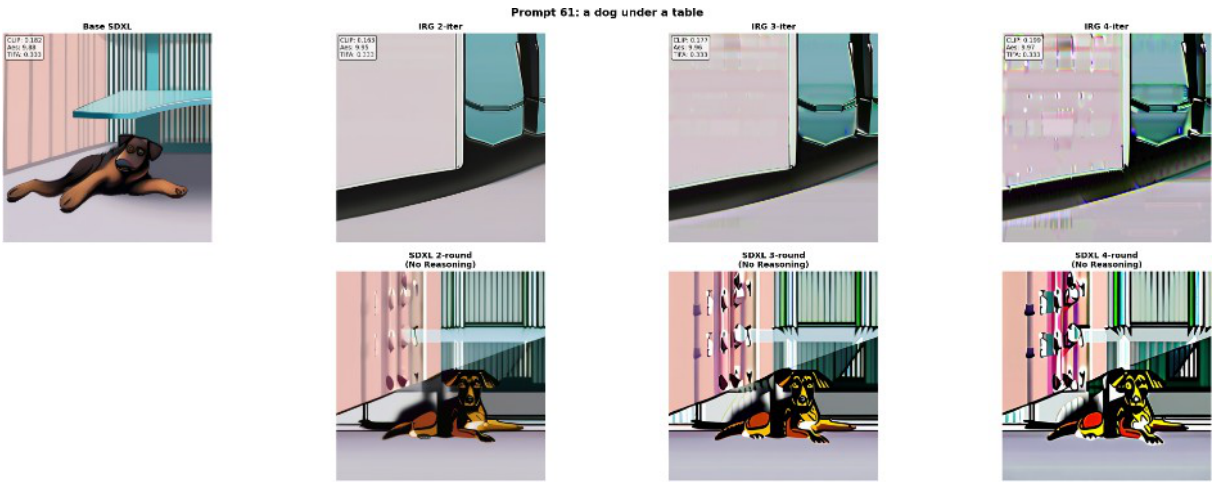


Figure 15: Visual comparison of Spatial Relation Confusion.

that excessive iteration causes over-smoothing where crisp object boundaries degrade into painterly blurs. Figure 16 shows this progression.


Figure 16: Visual comparison of over-refinement.

These failure patterns highlight the fundamental limitations of using vision-language-only feedback for spatial and compositional reasoning, motivating future integration of multimodal vision-language models as discussed in Chapter 6.

## 5.5 Chapter Summary

The quantitative and qualitative analyses presented above characterize the behavior of the interleaving reasoning architecture relative to baseline single-shot models. With a demonstrated peak improvement of +7.74% on the compositional accuracy benchmark and effective logic correction across iterations, the system has successfully validated the initial research hypothesis that modular interleaving reasoning can provide measurable benefits under resource constraints. However, the inverted U-curve phenomenon and the

degradation at higher iteration counts reveal the limitations of the approach. Based on these empirical findings, Chapter 6 concludes the research by summarizing the key contributions, discussing the implications of the results, and suggesting potential directions for future development.

# 6 Discussion and Conclusion

This final chapter synthesizes the empirical findings presented in Chapter 5 into broader theoretical and practical implications for reasoning-enhanced text-to-image generation. Having demonstrated that modular Interleaving Reasoning-Generation achieves +7.74% compositional accuracy improvement under constrained conditions while revealing the inverted U-curve phenomenon, we now contextualize these results within the larger landscape of generative artificial intelligence research.

## 6.1 Discussion of Empirical Findings

### 6.1.1 The Divergence of Metrics: Aesthetics versus Accuracy

Our analysis uncovers a critical trade-off between visual aesthetics and compositional accuracy. While aesthetic score exhibits monotonic improvement across all iterations (peaking at +4.06% for no-reasoning 4-round), compositional accuracy only improves with interleaving reasoning 2-iteration (+7.74%), degrading significantly at higher iterations. Notably, alignment score consistently degrades across all variants (-0.53% to -7.31%), contrary to expectations. This divergence indicates that iterative refinement increasingly prioritizes perceptual aesthetics over semantic fidelity. While additional iterations enhance low-level visual coherence, they simultaneously weaken the model's ability to preserve compositional constraints such as object relations and spatial structure. The consistent decline in alignment score further suggests that improved visual quality does not imply better text-image alignment, but rather reflects a growing bias toward learned aesthetic priors. The fact that compositional accuracy peaks at interleaving reasoning 2-iteration highlights the presence of an optimal refinement depth. At this stage, structured guidance improves compositional reasoning without excessive aesthetic over-optimization. Beyond this point, further iterations yield diminishing-and often negative-returns in compositional accuracy, underscoring the need for balanced or adaptive refinement strategies.

## 6.2 Comparative Analysis: Modular versus Unified Architectures

This study provides the first direct feasibility analysis of adapting the Interleaving Reasoning-Generation paradigm [6] from a unified multimodal model to a modular architecture under resource constraints.

### 6.2.1 Performance Retention

Despite operating with significantly reduced computational resources, our modular framework demonstrates measurable improvements in compositional understanding over the baseline Stable Diffusion XL model, with +7.74% compositional accuracy improvement at 2 iterations. The observed improvements in spatial reasoning and object counting suggest that substantial benefits from iterative reasoning can be achieved through modular architectures. We did not perform direct head-to-head comparison with unified models due to access constraints and differing evaluation protocols, so quantitative performance retention estimates relative to these systems cannot be reliably determined.

## 6.3 Practical Implications: Adaptive Iteration Routing

Based on the distinct peaks of different metrics, we propose that a one-size-fits-all approach to iteration count is suboptimal. Instead, we advocate for an adaptive iteration routing strategy for production deployment. For composition-critical tasks involving counting, spatial layouts, or multiple distinct entities, the system should strictly terminate at iteration 2 to maximize compositional accuracy scores and prevent object blending. For aesthetic-critical tasks such as abstract, artistic, or landscape prompts where visual polish is paramount, the system should proceed to iteration 4 to maximize aesthetic scores. For simple queries involving single-object prompts, the baseline generation (iteration 1) is often sufficient, and bypassing further refinement saves computational costs.
Implementing this routing logic could reduce average computational costs by up to 50% while maintaining optimal quality for each specific use case.

## 6.4 Limitations

We acknowledge several limitations inherent to our resource-constrained design.
The use of a vision-language model as a feedback mechanism relies on global semantic similarity scores. This approach lacks spatial granularity; the system knows the image is "wrong" but cannot pinpoint where (e.g., "the cat is 50 pixels too far left"). This coarseness limits the precision of the language model's reflection.
The necessity of loading and unloading large models (language model, diffusion model) sequentially to fit within constrained memory creates a bottleneck that prevents real-time application.
Due to hardware constraints, we employed simplified versions of certain metrics (e.g., faithfulness-lite based on word overlap rather than Visual Question Answering). While this provides valid relative comparisons, absolute scores may differ from benchmarks using full-scale evaluation models.
Our baseline is a single-shot Stable Diffusion XL generation without reasoning. A stronger

experimental design would include additional baselines such as: Stable Diffusion XL multi-round image-to-image with fixed or random prompts (no language model reasoning), Stable Diffusion XL with refiner, and Stable Diffusion XL with naive prompt engineering. Without these, we cannot definitively isolate whether improvements come from the reasoning component or simply from multiple image-to-image passes. Future work should address this ablation to strengthen causal claims about the value of language model-guided reflection.

### 6.4.1 Threats to Validity

We identify the following threats to the validity of our findings, in the interest of scientific transparency.

Regarding construct validity, the faithfulness-lite metric (caption-based word overlap) is a simplified proxy for attribute grounding and does not capture the full semantic faithfulness measured by Visual Question Answering-based approaches. Our heuristic aesthetic metric (color diversity plus brightness plus contrast) captures technical quality but may not align with human aesthetic preferences. Our object detection-based compositional accuracy approximation may differ from the original benchmark's object detection methodology.

Our empirical evaluation is subject to computational limitations, as all experiments were conducted exclusively on specific hardware. This restriction constrained model scale, batch size, and the number of refinement iterations that could be explored, potentially influencing the observed performance trends. In particular, phenomena such as the degradation in compositional accuracy at higher iteration counts may partially stem from hardware-specific bottlenecks rather than intrinsic limitations of the proposed method. Consequently, while our findings are valid within the evaluated setting, their generalizability to more powerful computational environments remains uncertain.

The rule-based template system generates reasoning traces that may lack the diversity and naturalness of human-written or larger language model-generated data, potentially limiting the fine-tuned model's generalization.

These limitations are acknowledged to enable appropriate interpretation of our results and to guide future research toward more rigorous experimental designs.

## 6.5 Future Research Directions

To address these limitations, future work should focus on three key areas.

Replacing the text-only language model and vision-language feedback loop with a unified Vision-Language Model would allow the reasoning module to "see" the image directly, enabling pixel-level feedback and reducing the hallucination rate during reflection.

Investigating flow matching models to replace the diffusion component could potentially reduce the generation steps significantly, substantially mitigating the latency issues of the

iterative pipeline.
Applying the interleaving reasoning paradigm to video synthesis could enable a language model to act as a director, planning the storyboard and maintaining temporal consistency between frames through iterative reflection, addressing the flickering issues common in current video models.

## 6.6 Conclusion

This thesis presents a pioneering effort to democratize advanced Interleaving Reasoning-Generation capabilities. By architecting a modular framework that operates within the constraints of consumer-grade hardware, we have successfully demonstrated that core benefits of reasoning-enhanced generation-specifically in error correction and semantic alignment-can be partially preserved in accessible implementations.
Our empirical discovery of the inverted U-curve in compositional accuracy challenges the prevailing "more is better" philosophy in iterative refinement, suggesting a need for more nuanced, adaptive control strategies. Ultimately, this work provides a foundational framework and a high-quality reasoning corpus that empowers students and researchers to explore the frontiers of generative artificial intelligence without the barrier of prohibitive computational costs. We believe that such accessible, transparent approaches are essential for the sustainable and inclusive advancement of Artificial Intelligence.

# Appendix

## Appendix A: Data Synthesis Pipeline Details

### A.1. Rule-Based Template System for Visual Reasoning Generation

To generate the synthetic training corpus, we developed a rule-based template synthesis system. This approach was chosen over large language model-based generation for several reasons: cost efficiency (no API costs for large-scale data generation), reproducibility (deterministic outputs given the same random seed), and control (fine-grained control over data distribution and quality profiles).

The generator consists of two main components: a feature generator that generates synthetic vision-language feature statistics (mean, standard deviation, maximum) based on quality profiles, and an improved corpus generator that generates training examples using template-based composition.

### A.2. Data Generation Examples

Example 1 shows initial reasoning (Iteration 1). Input prompt: "cat, in a forest, photorealistic". Generated reasoning: "Composition: Use rule of thirds to create visual balance and interest. Position the main subject according to this principle. Lighting: Apply golden hour to establish mood and depth. Ensure consistent light direction and appropriate shadows. Color palette: Choose colors that support the subject and mood. Apply proper color temperature and saturation. Technical: Maintain sharp focus on the subject with appropriate depth of field. Ensure high detail quality and proper exposure." Type: initial reasoning. Iteration: 1.

Example 2 shows feature-aware refinement starting from Iteration 2. The input prompt is *"motorcycle, underwater"*. The extracted synthetic features have a mean value of 0.433, a standard deviation of 0.181, and a maximum value of 0.965, corresponding to an *excellent* quality profile.

The generated output is as follows. *Image Analysis (Vision–Language Features):* the feature state reports a mean of 0.433, a standard deviation of 0.181, and a maximum of 0.965. *Feature Interpretation:* brightness is adequate (mean = 0.433), contrast is good (standard deviation = 0.181), and highlights exhibit a good dynamic range (maximum = 0.965). *Corrective Actions:* image quality is already good, therefore only subtle enhancements are applied, including fine-tuning color balance and saturation, as well as sharpening key details without over-processing.

This example is classified as a *feature refinement* case.

Example 3 shows problem-solving example. Scenario: Fixing Underexposure. Synthetic features: mean=0.22, std=0.11, max=0.58. Diagnosis: "Brightness: severely underex-

posed (mean=0.220), Contrast: low contrast (std=0.110), Highlights: lacks highlights (max=0.580)". Corrective actions: "Increase overall exposure by 22%. Boost midtones by 11%. Increase contrast by 22%. Add shadow depth by 11%. Add specular highlights to key surfaces."

### A.3. Critique and Refinement Prompt (Reflection Stage)

For iterations 2–4, the language model uses a different prompt template to generate refinement instructions. The system role states: "You are a visual critic analyzing generated images. Based on the current image state (represented by vision-language features and alignment score), identify specific improvements." The user provides: original prompt, previous reasoning, current alignment score, and feedback. The model generates a refinement plan focusing on what needs correction or enhancement.

Example refinement output: "The current image shows the cat and dog, but their interaction is weak-they appear to be merely standing near each other rather than 'playing together'. REFINEMENT: Add dynamic poses-cat should have paw extended toward dog, dog should be in play-bow position. Increase spatial overlap by 20% to emphasize 'together'. Enhance lighting on interaction point to draw visual focus."

# References


[1] 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. *arXiv preprint arXiv:2307.01952*, 2023.

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

[3] Susung Hong, Donghoon Ahn, and Seungryong Kim. Debiasing scores and prompts of 2d diffusion for view-consistent text-to-3d generation. *Advances in Neural Information Processing Systems*, 36:11970–11987, 2023.

[4] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

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

[6] Wenxuan Huang, Shuang Chen, Zheyong Xie, Shaosheng Cao, Shixiang Tang, Yufan Shen, Qingyu Yin, Wenbo Hu, Xiaoman Wang, Yuntian Tang, et al. Interleaving reasoning for better text-to-image generation. *arXiv preprint arXiv:2509.06945*, 2025.

[7] 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. *arXiv preprint arXiv:2408.12528*, 2024.

[8] 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*, 2025.

[9] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[10] Dongzhi Jiang, Ziyu Guo, Renrui Zhang, Zhuofan Zong, Hao Li, Le Zhuo, Shilin Yan, Pheng-Ann Heng, and Hongsheng Li. T2i-r1: Reinforcing image generation with

collaborative semantic-level and token-level cot. *arXiv preprint arXiv:2505.00703*, 2025.

[11] 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. *Advances in Neural Information Processing Systems*, 36:18225–18250, 2023.

[12] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18392–18402, 2023.

[13] Vaswani Ashish. Attention is all you need. *Advances in neural information processing systems*, 30:I, 2017.

[14] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

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

[16] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[17] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36:52132–52152, 2023.

[18] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

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

[20] Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. Gpt3. int8 (): 8-bit matrix multiplication for transformers at scale. *Advances in neural information processing systems*, 35:30318–30332, 2022.

[21] Akito Watanabe, Yoshinobu Nakatani, Hiroyuki Awano, and Kenji Tanabe. Machine learning-based determination of magnetic parameters from magnetic images with different imaging scales. *arXiv preprint arXiv:2408.12181*, 2024.